# Table of Contents

How do I create a body? .................................................................................................................. 5
  Brief Introduction to bodies: ....................................................................................................... 5
  Steps to create a ground body ................................................................................................. 5
  Steps to create a rigid body ..................................................................................................... 7
  Steps to create a flex body ...................................................................................................... 7
  Example model ....................................................................................................................... 8
  Comments: ............................................................................................................................. 9
  Additional info: ....................................................................................................................... 9

How do I create a beam? .............................................................................................................. 10
  Brief introduction to beams: .................................................................................................. 10
  How to create beams using points ......................................................................................... 10
  Example Model ..................................................................................................................... 14
  Comments: ............................................................................................................................. 15
  Additional info: ....................................................................................................................... 15

How do I create a Joint? ............................................................................................................. 16
  A brief introduction to joints: ............................................................................................... 16
  How to create a joint ............................................................................................................. 17
    Ball ......................................................................................................................................... 17
    Fixed ....................................................................................................................................... 18
    Revolute ................................................................................................................................ 19
    Translational ....................................................................................................................... 20
    Cylindrical ............................................................................................................................. 21
    Universal ............................................................................................................................... 22
    Constant Velocity ............................................................................................................... 23
    Planar ...................................................................................................................................... 24
    Inline ..................................................................................................................................... 25
    Perpendicular ...................................................................................................................... 26
    Parallel axes ......................................................................................................................... 27
    Inplane ................................................................................................................................... 28
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>How do I create a Bushing?</td>
<td>41</td>
</tr>
<tr>
<td>Brief introduction to bushings</td>
<td>41</td>
</tr>
<tr>
<td>How to create a bushing</td>
<td>41</td>
</tr>
<tr>
<td>Example Model</td>
<td>42</td>
</tr>
<tr>
<td>Comments</td>
<td>42</td>
</tr>
<tr>
<td>Additional Info</td>
<td>42</td>
</tr>
<tr>
<td>Brief introduction to Analysis Methods</td>
<td>43</td>
</tr>
<tr>
<td>How to specify more than one Method of Analysis</td>
<td>43</td>
</tr>
</tbody>
</table>
Example Model ........................................................................................................................................45
Comments ..............................................................................................................................................45
Additional info.......................................................................................................................................46
How do I define Units? ..........................................................................................................................46
Brief introduction to Units .....................................................................................................................46
How to define Units ................................................................................................................................46
Example model ........................................................................................................................................47
Comments ..............................................................................................................................................47
Additional info.......................................................................................................................................47
How do I create a Spring Damper? ........................................................................................................48
Brief introduction to Spring Dampers ....................................................................................................48
How to create a Spring Damper ...............................................................................................................48
Other Methods ........................................................................................................................................48
Example Model ........................................................................................................................................50
Comments ..............................................................................................................................................51
Additional info.......................................................................................................................................51
How do I create a coupler? .....................................................................................................................52
Brief introduction to couplers ................................................................................................................52
How to create a coupler..........................................................................................................................52
Example model ........................................................................................................................................53
Comments ..............................................................................................................................................53
Additional info.......................................................................................................................................53
How do I create a Marker? .....................................................................................................................54
Brief introduction to Markers ................................................................................................................54
How to create a Marker ..........................................................................................................................54
Example Model ........................................................................................................................................55
Comments ..............................................................................................................................................56
Additional info.......................................................................................................................................56
How do I create a curve? .........................................................................................................................57
How do I create a Force or Moment? .....................................................................................................62
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brief introduction to Forces and Moments</td>
<td>62</td>
</tr>
<tr>
<td>How to set up a load collector:</td>
<td>63</td>
</tr>
<tr>
<td>How to add grid point Force (MBFRC)</td>
<td>64</td>
</tr>
<tr>
<td>How to add grid point Force (MBFRCC)</td>
<td>67</td>
</tr>
<tr>
<td>How to add grid point Force (MBFRCE)</td>
<td>69</td>
</tr>
<tr>
<td>How to add grid point Moment (MBMNT)</td>
<td>71</td>
</tr>
<tr>
<td>How to add grid point Moment (MBMNTC)</td>
<td>73</td>
</tr>
<tr>
<td>How do I create an Initial Condition?</td>
<td>81</td>
</tr>
<tr>
<td>Brief introduction to Initial Conditions</td>
<td>81</td>
</tr>
<tr>
<td>How to create an Initial Condition</td>
<td>81</td>
</tr>
<tr>
<td>Example model</td>
<td>82</td>
</tr>
<tr>
<td>Comments</td>
<td>82</td>
</tr>
<tr>
<td>Additional info</td>
<td>82</td>
</tr>
<tr>
<td>Bulk data input format</td>
<td>83</td>
</tr>
</tbody>
</table>
How do I create a body?

Brief Introduction to bodies:
Bodies are the model elements that have mass and inertia. Bodies can be of types: rigid or flexible. Rigid bodies are ideal representations of solid bodies or parts of fixed shape and size. They are highly useful in simulations where deformation of a part is negligible. A rigid body has 6 DOF, and each rigid body added to a system adds an additional 6 DOF. Flexible bodies, or flex bodies, are used to model elastic deformation of bodies in a system. The flex body connects to its neighboring elements and bodies through interface nodes. The flex body consists of reduced stiffness and mass matrices, which can be obtained in various ways. The two methods used in HyperMesh are the Craig-Bampton and Craig-Chang methods. Properties of a flex body are defined by the user in HyperMesh, and control the interaction of the flex body with other bodies.

Steps to create a ground body

a. Click on the analysis radio button in the project browser, then click on the “bodies” button.

b. Make sure that the radio buttons on the left are set to “create”. In the window next to “body =”, Give a name to the ground body.
c. Click on the "type =" button, and select the option "GROUND".

d. At least one property, element, or node must be selected to create a body. To select a property, element, or node, click the corresponding yellow button, and a blue box will appear on the button. In the model display window, click on the entity you want to assign the body to.

e. Click “Create”.
Steps to create a rigid body

f. Follow steps a. and b. from creating a ground body.
g. From the “type=” options, choose “PRBODY”.
h. Choose the properties, elements, and nodes the same way as in part d. of creating the ground body.
i. Click “Create”.

Steps to create a flex body

j. Follow steps a. and b. from creating a ground body.
k. From the “type=” options, choose “PFBODY”.
a. Choose the properties, elements, and nodes the same way as in part d. of creating the ground body.
b. You may choose the Component Mode Synthesis method and frequency upper bound, or leave them as default. You must have values for at least the frequency upper bound or the number of modes.
c. Click “Create”.
Example model

This model of a MacPherson suspension uses rigid and flex bodies. Each component in the system is labeled in order to identify their properties, which can be found in the table. The ground body describes the reference environment, and does not add any degrees of freedom to the system and a flex body is used on the component of interest. Because this model is only focused on the LCA characteristics under simulation, the other bodies in the mechanism are rigid.
Comments:

1. A maximum of 56 characters may be given for the body name.
2. Flex H3D file name will be <BODY_NAME>.h3d or OUTFILE_body_<BID>.h3d in the outfile directory.
3. Any number of property definitions; CELAS2, CONM2, PLOT, RBAR, RBE2, RBE3 or ROD elements or grid points can be given.
4. At least one property definition, element, or grid point must be given.
5. A property definition; CELAS2, CONM2, PLOT, RBE2, RBE3, RBAR or ROD element or grid point can only belong to one body (flexible or rigid).
6. All property definitions, elements and grid points defined on a PFBODY or PRBODY bulk data entry forms one flexible or rigid body respectively.
7. CMS definition defines the component mode synthesis method to reduce the flexible body for the multi-body analysis. Exactly one must be defined for each PFBODY.
8. Two methods are available for Component Mode Synthesis: CB – Craig-Bampton and CC – Craig-Chang.
9. UB_FREQ and NMODES cannot both be blank. When UB_FREQ = 0.0 and NMODES = 0, this is a special case where no Eigen modes will be included in CMS mode generation.
10. If FLXNODE is not defined, a default set of interface nodes and degrees-of-freedom will be generated based on the actual interface nodes and degrees-of-freedom of the flexible body. One FLXNODE line can have up to six interface grid IDs. No continuation lines are allowed. Add multiple FLXNODE lines to add more than six interface nodes.
11. The mass, inertia and center of gravity input is optional if element/property information is provided in the PRBODY definition. If one of MASS, INERTIA or COG continuations is provided, all three continuations must be provided. MASS must be positive non-zero values. If just the principal inertia is specified, Ixx, Iyy, Izz must be positive non-zero values and they must satisfy the condition: the sum of two inertia values must be greater than the third (Ixx + Iyy > Izz, Iyy+Izz > Ixx, Izz+Ixx > Iyy).
12. For a PRBODY, a CID of zero or blank references the basic coordinate system.
13. The grid ID provided in the ground card is considered grounded.
14. These cards are represented as a group in HyperMesh.

Additional info:

- file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/body_point.htm
- file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/body_rigid_model.htm
- file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/body_flexible.htm
- file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/ground.htm
- file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/prbody.htm
- file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/pfbody.htm
How do I create a beam?

Brief introduction to beams:
A beam is an entity defined as a straight, massless beam of uniform cross section acting between two points or markers belonging to two different bodies. The mass of the beam is lumped at each point. Beam stiffness properties are derived using Timoshenko beam theory.

The beam axis is assumed to be along the x-axis of the Body 2 Reference Marker. This marker axis also represents the neutral axis of the un-deformed beam. The beam is assumed to undergo small rotational deflections; large rotations are not supported.

How to create beams using points

a. In the project menu, click on the 1D radio button, then click the “HyperBeam” button.

b. Leave the settings as “Standard Selection”, library: “HYPERBEAM”, and type: “thin walled box”, then click “create”. This will bring you to the HyperBeam workspace environment.

c. In the Parameter Definition window in the lower left corner of the window, change the values of dimensions a and b, as well as the thickness, to the desired values.

d. In the browser menu at the top of the window, click File>Exit.
Create a new property by right clicking in the Model tree, and selecting Create>Properties.

give the property a name, and set the card image to PBEAM.

Assign a material to the property in the material tab, and make sure the card edit box is checked before clicking “Create”.

h. This will bring up the card edit panel for PBEAM. Click the yellow button labeled "beamsec".

![Card edit panel for PBEAM]

i. In the selection panel, choose the newly created beam.

![Selection panel with beam selected]

j. Hit return to exit the card edit panel.

k. In the project menu, select the 1D radio button, then click on the “bars” button.

![Project menu with 1D and bars selected]

l. Set the orientation to “vector”, and switch the value next to it to “x-axis”.

![Elements panel with orientation set to x-axis]

m. Click the “property =" button and select the property that was just created.

![Property selection dialog]

n. Click the “elem types =” button and select the “CMBEAM” card.

![Element types selection dialog]

o. The first of the two yellow boxes will automatically be checked. Click on the first node to be connected in the model window. After this, the second yellow box will already be selected. Click on the second node to connect.
This will create a beam between the two nodes.

Click "return" to go back to the project menu.

To create a beam using markers, a bulk data card must be used. Begin by clicking on the “control cards” button in the analysis panel of the Project Menu.

Click on the “BULK_UNSUPPORTED_CARDS” button to open the Control Card window.

Define the beam in the control card using the following format:
How do I?

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMEEAMM</td>
<td>EID</td>
<td>MID</td>
<td>M1</td>
<td>M2</td>
<td></td>
<td></td>
<td>L</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>I1</td>
<td>I2</td>
<td>J</td>
<td>K1</td>
<td>K2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Where EID is the element ID of the beam, MID is the Material identification number, M1 and M2 are Marker identification numbers, L is the Undeformed Length of the beam, A is the Cross sectional area of the beam, I1 is the area moment of inertia in plane 1 about the neutral axis, I2 is the area moment of inertia in plane 2 about the neutral axis, J is the torsional constant, and K1 and K2 are the Area Factors for Shear. An example is shown here:

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMEEAMM</td>
<td>1</td>
<td>2</td>
<td>123</td>
<td>125</td>
<td>0.0</td>
<td>0.0</td>
<td>1.0</td>
<td>5.0</td>
<td></td>
</tr>
<tr>
<td>100.0</td>
<td>833.3</td>
<td>833.3</td>
<td>1485.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Click “ok” to close the Control Cards window, and the beam will automatically be created. Click “return” to go back to the Project Menu.

Example Model

beam_start.hm

This model consists of two point mass bodies, using predefined components, materials, and properties, in order to simplify the process of adding a beam between two nodes.
Comments:

r. Element identification numbers must be unique with respect to all other element identification numbers.
s. The x-axis of the beam is always along the line connecting G1 and G2. The z-axis of the beam is determined based on the x-axis and the y-axis provided by G3/X1, Y1, Z1.
t. Only MAT1 material definitions may be referenced by the CMBEAM element.
u. This card is represented as a bar2 element in HyperMesh.

Additional info:
file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/cmbeam.htm
file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/cmbeamm.htm
file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/beam_elements_guidelines.htm
How do I create a Joint?

A brief introduction to joints:
A joint or constraint is a mechanical connection which constrains relative motion between bodies. They are also used to restrict the motion of bodies independently. There are a number of joints that can be used to do this and the joint may restrict the absolute motion of a body relative to the ground body or the relative motion between interconnected bodies. In MotionView you can create 17 types of Joints. These 17 joints fall into three separate categories: Lower Pair Constraints, Higher Pair Constraints, and Joint Primitives.

A Lower Pair Constraint is an ideal joint that constrains contact between a point, line or plane in the moving body to a corresponding point, line or plane in a fixed body or another moving body.

In Higher pair constraints, the two bodies are in contact at a point or along a line, as in a ball bearing or disk cam and follower and the relative motions of coincident points are not same.

A joint primitive places a restriction on relative motion, such as restricting one part to always move parallel to another part. The joint primitives do not have physical counterparts as the idealized joints do. You can, however, combine joint primitives to define a complex constraint that cannot be modeled using the idealized joints.

The joints used by HyperMesh, as well as their DOF constraints can be seen in the table below:

<table>
<thead>
<tr>
<th>Constraint Type</th>
<th>Translational Constraints</th>
<th>Rotational Constraints</th>
<th>Total Constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant Velocity</td>
<td>3</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>Cylindrical</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>Fixed</td>
<td>3</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>Planar</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Revolute</td>
<td>3</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>Translational</td>
<td>2</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>Spherical / Ball</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>Orientation</td>
<td>0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Universal</td>
<td>3</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>Inline</td>
<td>2</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Inplane</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Perpendicular Axes</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Parallel Axes</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Point to Deformable Surface</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Point to Curve</td>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>Curve to Curve</td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>Point to Deformable Curve</td>
<td></td>
<td></td>
<td>2</td>
</tr>
</tbody>
</table>
How to create a joint

The following instructions on creating joints are organized in order of their occurrence in HyperMesh.

<table>
<thead>
<tr>
<th>ball</th>
<th>universal</th>
<th>parallel axes</th>
<th>ptdcv</th>
</tr>
</thead>
<tbody>
<tr>
<td>fixed</td>
<td>constant vel</td>
<td>inplane</td>
<td>ptdsrf</td>
</tr>
<tr>
<td>revolute</td>
<td>planar</td>
<td>orient</td>
<td></td>
</tr>
<tr>
<td>translational</td>
<td>inline</td>
<td>ptcv</td>
<td></td>
</tr>
<tr>
<td>cylindrical</td>
<td>perpendicular</td>
<td>cvc</td>
<td></td>
</tr>
</tbody>
</table>

Ball

A Ball Joint (also known as Spherical joint) is a three degree of freedom kinematic pair used in mechanisms. Spherical Joints provide three-axes rotational function used in many places such as ball and socket joints, pivots between wheels and suspension of an automobile, and motion control systems.

i. Begin by opening the Joints panel in the 1D Project Menu.

Figure 15: A ball joint
iii. A ball joint requires two elements from separate bodies to be selected. Use the model window to click on the elements needed, or specify them by id by clicking on the “node” button that is already highlighted.

![Model window with joint creation interface]

iv. After selecting both nodes, click the “create” button on the right, and “return” to go back to the project menu.

(return to Joints)

Fixed

![Fixed joint diagram]

**Figure 4:** A schematic of a FIXED joint

A Fixed Joint is a zero degree of freedom kinematic pair used in mechanisms. Fixed Joints do not provide any degree of freedom, and used in many places such as welded joints or riveted joints.

v. Begin by opening the Joints panel in the 1D Project Menu.

<table>
<thead>
<tr>
<th>nodes</th>
<th>bars</th>
<th>connectors</th>
<th>line mesh</th>
<th>edit element</th>
</tr>
</thead>
<tbody>
<tr>
<td>joint</td>
<td>node</td>
<td>spot weld</td>
<td>linear 1d</td>
<td>gait</td>
</tr>
<tr>
<td>markers</td>
<td>ngds</td>
<td>Hyper Stream</td>
<td>replace</td>
<td>attach</td>
</tr>
<tr>
<td>rts</td>
<td>sqins</td>
<td>gaps</td>
<td>vectors</td>
<td>config edit</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>systems</td>
<td>elem types</td>
</tr>
</tbody>
</table>
vii. A fixed joint requires nodes from two separate bodies. Use the model window to click on the elements needed, or specify them by id by clicking on the “node” button that is already highlighted.

viii. After selecting both nodes, click the “create” button on the right, and “return” to go back to the project menu.

(return to Joints)

Revolute

![Figure 13: A revolute joint](image)

A Revolute Joint (also known as a pin joint or a hinge joint) is a one degree of freedom kinematic pair used in mechanisms. Revolute joints provide single-axis rotation function used in many places such as door hinges, folding mechanisms, and other uni-axial rotation devices.

ix. Begin by opening the Joints panel in the 1D Project Menu.

x. For “joint type:” select revolute from the joints list.

xi. A revolute joint attaches two bodies and constrains 5 DOF. Entities from two different bodies are required, as well as an orientation for the axis of rotation. This can be done by choosing either a node or a vector; this can be specified by clicking the arrow next to the yellow “node” button.
xii. After selecting both nodes and the orientation, click the “create” button on the right, and “return” to go back to the project menu.

(return to Joints)

Translational

Figure 16: A translational joint

A Translational joint (also known as Prismatic Joint) provides a linear sliding movement between two bodies. It allows one degree of freedom in the joint. The relative position of two bodies connected by a prismatic joint is defined by the amount of linear slide of one relative to the other one. Translational joints provide single-axis translation function used in many places such as piston-cylinder arrangements, and slider-crank linkage.

xiii. Begin by opening the Joints panel in the 1D Project Menu.

xiv. For “joint type:” select translational from the joints list.

xv. Entities from two different bodies are required, as well as an orientation for the axis of translation. This can be done by choosing either a node or a vector; this can be specified by clicking the arrow next to the yellow “node” button.
xvi. After selecting both nodes and the orientation, click the “create” button on the right, and “return” to go back to the project menu.

(return to Joints)

Cylindrical

Figure 3: A cylindrical joint

A Cylindrical Joint is a two degree of freedom kinematic pair used in mechanisms. Cylindrical Joints provide single-axis sliding function as well as single-axis rotation, providing a way for the rigid bodies to translate and rotate freely. It is used to model constraints in systems like hydraulic cylinders, and slider-crank pin-slot links.

xvii. Begin by opening the Joints panel in the 1D Project Menu.

xviii. For “joint type:” select cylindrical from the joints list.

xix. A cylindrical joint allows two degrees of freedom; one in a translational and one in a rotational direction. A single axis of direction is needed, as the translational and rotational directions of the joint are codependent. Select two nodes, and specify an orientation node or vector, then click the “Create” button.

xx. After selecting both nodes and the orientation, click the “create” button on the right, and “return” to go back to the project menu.

(return to Joints)
Universal

A Universal joint (also known as universal coupling, U-joint, Cardan joint, Hardy-Spicer joint, or Hooke’s joint) is a joint or coupling between two rigid shafts that allows the shafts to orient in any direction, and is commonly used in shafts that transmit rotary motion. It consists of a pair of hinges, oriented at 90° to each other, connected by a cross pins. Universal joints are most commonly used in drive shafts to transfer rotation.

xxi. Begin by opening the Joints panel in the 1D Project Menu.

xxii. For “joint type:” select universal from the joints list.

xxiii. Universal joints allow two rotational degrees of freedom, and require two orientations. Click on the nodes defining the joint location in the model window, or enter their id’s in order to create the joint.

xxiv. After selecting both nodes and orientations, click the “create” button on the right, and “return” to go back to the project menu.

(return to Joints)
How do I?

Constant Velocity

The constant velocity joint is a fairly complex subsystem. It ensures that the angular speed of the output shaft is the same as the angular speed of the input shaft, regardless of their relative orientation. The construction of a constant velocity joint is shown above in Figure 2. The detail in the constant velocity joint is abstracted away; all intermediate parts are ignored and the input-output relationship between the shafts is captured via a simple set of algebraic constraints. Energy loss is not modeled.

xxv. Begin by opening the Joints panel in the 1D Project Menu.

xxvi. For “joint type:” select constant vel from the joints list.

xxvii. Like with universal joints, constant velocity joints constrain all but two degrees of rotational freedom. Creating a constant velocity joint requires two nodes from different bodies, and two orientation directions. Select all four from the Model Window, or by specifying their id number.

xxviii. After selecting both nodes and orientations, click the “create” button on the right, and “return” to go back to the project menu.
A Planar joint (also known as Gliding Joint) constrains two bodies so as to restrict relative motion between the two bodies to the xy plane and allow relative rotations about the z-axis only. Planar Joints provide single-axis rotation and two axes translation function used in many places.

xxi. Begin by opening the Joints panel in the 1D Project Menu.

xxx. For “joint type:” select planar from the joints list.

xxxi. Select “planar” for the joint type, then select two nodes by clicking in the Model Window, or specifying their id’s, to define the joint location. Choose either a node or vector to define the joint orientation.

xxi. After selecting both nodes and the orientation, click the “create” button on the right, and “return” to go back to the project menu.

(return to Joints)
The inline joint is a joint primitive that requires that a point on Body 1 (Origin 1) translate along the axis represented by the line connecting Origin 1 and Origin 2. All rotations are allowed

xxxiii. Begin by opening the Joints panel in the 1D Project Menu.

xxxiv. For “joint type:” select inline from the joints list.

xxxv. Select “inline” for the joint type, and choose two nodes to define the joint location. Set the orientation to either a node or vector.

xxxvi. After selecting both nodes and the orientation, click the “create” button on the right, and “return” to go back to the project menu.

(return to Joints)
Perpendicular

Figure 1: A Schematic of the Perpendicular Axes Joint Primitive

This joint primitive constrains Body 1 such that the z-axis of a Reference Marker (I) in Body 1 is always PERPENDICULAR to the z-axis of a Reference Marker (J) on Body 2.

xxxvii. Begin by opening the Joints panel in the 1D Project Menu.

xxxviii. For “joint type:” select perpendicular from the joints list.

xxxix. Perpendicular joints only constrain one rotational degree of freedom. Select the two nodes to define the joint location. Next, choose the two orientations by specifying a directional node or vector for each orientation.

xl. After selecting both nodes and orientations, click the “create” button on the right, and “return” to go back to the project menu.

(return to Joints)
Parallel axes

Figure 1: Parallel Axes Joint Schematic

The Parallel Axes Joint Primitive constrains Body 1 such that the z-axis of a Reference Marker (I) is always parallel to the z-axis of a Reference Marker on Body 2 (J). The Parallel Axis Joint Primitive removes two rotational degrees of freedom. Rotation of Body 1 is only allowed about the z-axis of the I Reference Marker. All three translations are allowed.

xli. Begin by opening the Joints panel in the 1D Project Menu.

xlii. For “joint type:” select parallel axes from the joints list.

xliii. Parallel axes joints constrain two rotational degrees of freedom. Define the joint location at two coincident nodes, and specify the orientation of the joint. It is important to keep both orientations the same, as they could cause solution failure.

xliv. After selecting both nodes and orientations, click the “create” button on the right, and “return” to go back to the project menu.

(return to Joints)
Inplane

Figure 1: A Schematic of the Inplane Joint

The inplane joint is a joint primitive that requires that the origin of the joint that belongs to Body 1 to stay in the XY plane defined by Joint, the Plane belongs to Body 2.

xlv. Begin by opening the Joints panel in the 1D Project Menu.

xlvi. For “joint type:” select inplane from the joints list.

xlvii. An inplane joint constrains one translational degree of freedom, and no rotational degrees of freedom. The joint requires two translation orientations, in addition to two nodes, in order to be created.

xlviii. After selecting both nodes and orientations, click the “create” button on the right, and “return” to go back to the project menu.

*(return to Joints)*
Orient

An Orientation Joint requires that the orientation of Body 1 always be the same as the orientation of Body 2. An orient joint constrains all three relative rotations between the two bodies. The relative translations, however, are unconstrained.

i. Begin by opening the Joints panel in the 1D Project Menu.

   1. For “joint type:” select orient from the joints list.
   2. Orient joints constrain 3 degrees of freedom, all of which are rotational. No orientations need to be specified in addition to the two nodes defining the joint location. Select the nodes from the Model Window, or specify the id number of the nodes.

   iii. After selecting both nodes and orientations, click the “create” button on the right, and “return” to go back to the project menu.

PTCV

A Point to Curve Joint is a higher pair constraint. A fixed point on one body slides on a curve that is fixed on a second body. The point is not allowed to lift off the curve. The curve on the second body can be 2-D or 3-D and needs to be specified using a HyperMesh Curve which is
How do I?

A reference entity.

iii. Begin by opening the Joints panel in the 1D Project Menu.

- Begin by opening the Joints panel in the 1D Project Menu.
- For “joint type:” select PTCV from the joints list.
- Point to Curve joints use a node to define the point, and a node to define the location on the curve of the joint. A curve also needs to be defined. Select both nodes from the Model Window, then click on the “set” button.
- Select the curveset to be applied, and the panel will automatically return to the joint creation window.
- Click the “create” button on the right, and “return” to go back to the project menu.

CVCV

The Curve to Curve Joint defines higher pair constraint. The constraint consists of a 3D curve fixed on one body rolling and sliding on a 3D curve fixed on a second body. The curves are required to have a unique point of contact and a common tangent at that point of contact. The curve-to-curve constraint is useful for modeling cams where the point of contact between two parts changes during the motion of the system. The curves always maintain contact, however, even when the physics of the model might dictate that one curve lift off the other.

The curve-curve constraint removes three degrees of freedom from the system. One constraint enforces that the tangents to the two curves at the point of contact always
How do I? 31

remain parallel. This constraint prevents the bodies from spinning relative to each other about any axis normal to the common tangent at the point of contact. The other two constraints prohibit translational motion in the plane normal to the common tangent at the point of contact.

Iviii. Begin by opening the Joints panel in the 1D Project Menu.

lix. For “joint type:” select CVCV from the joints list.

lx. Curve to Curve joints use nodes from both curves for the location of the joint. Two curves also need to be defined. Select both nodes from the Model Window, then click on the “set” button for each curve.

lxii. Click the “create” button on the right, and “return” to go back to the project menu.

(return to Joints)

PTDCV

The Point to Deformable Curve Joint constrains a fixed point on a body to slide along a curve that passes through the origins of a specified set of markers. These markers may belong to different bodies. As the markers move in space, the curve is calculated at every time step using CUBIC spline interpolation through the marker origins. Hence, the curve deforms as the markers move about. This constraint is useful for simulating connection between a point on a body and a slender, flexible element such as a cable, or a ski chair-lift.

lxii. Begin by opening the Joints panel in the 1D Project Menu.
lxiii. For “joint type:” select PTDCV from the joints list. Select a node from the Model Window, and then click the “set” button to select the curveset.

lxiv. Click the “create” button on the right, and “return” to go back to the project menu.

(return to Joints)

PTDSRF

The Point to Deformable Surface Joint constrains a fixed point on a body to slide along a surface that passes through the origins of a specified set of markers. These markers may belong to different bodies. As the markers move in space, the surface is calculated at every time step using CUBIC spline interpolation through the marker origins. Hence, the surface deforms as the markers move about.

lxv. Begin by opening the Joints panel in the 1D Project Menu.

lxvi. For “joint type:” select PTDSRF from the joints list. Select a node from the Model Window, then click the “set” button to select the surface.

lxvii. Click the “create” button on the right, and “return” to go back to the project menu.

(return to Joints)

How to review a joint:

1. In the joint panel, switch the radio option from “create” to “update”.
2. Click the “review” button, and then select the joint element you wish to review in the Model Window.
3. Repeat process to review more joints.

Example Model

`suspension.fem`
How do I?

This model uses several joint entities described above. The ones used are translational (TRANS), ball (BALL), and fixed (FIX).

Comments:

b. Joints are only valid in a multi-body solution sequence.

c. Nodes1 and 2 identify the bodies being joined and must, therefore, belong to different bodies.

d. Joints define a set of constraints in the model. These constraints only allow the two connected bodies to have relative motion in certain specific directions. Motion in all the other directions is prohibited. The system is "constrained" not to move in those directions.

e. In their most general form, constraints can involve displacements, velocities, and time. Joint elements are distinguished by the fact that they only involve displacements. The constraint relationships do not involve velocities or time explicitly. Therefore, they do not add or remove energy from the system.

f. This card is represented as a joint element in HyperMesh.

Additional info:

file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/joint_bulk_data.htm

file:///C:/Program%20Files/Altair/11.0/help/hwd/joint.htm
How do I create a Motion?

Brief introduction to motion:

4. The MOTION card in HyperMesh is represented as a load collector. It is a combination of motion sets, which may be defined by MOTNJ, MOTNJC, MOTNJE, MOTNG, MOTNGC, and MOTNGE.

The motions supported by HyperMesh can be divided into two categories. MOTNJ, MOTNJC, and MOTNJE are all motions applied to joints. MOTNG, MOTNGC, and MOTNGE are all motions applied at points on a grid. These cards are all represented as constraint loads.

The cards specifying motion on joints can only define a motion at a degree-of-freedom in a joint. This input is only allowed in revolute, translational, and cylindrical joints. Translational motion may be specified on translational and cylindrical joints, and rotational motion may be specified on rotational and cylindrical joints. MOTNJ applies a specified constant motion on one of the three aforementioned joints. MOTNJC applies a motion as a function of time, dependant on a given curve. MOTNJE applies a motion as a function of a defined expression.

The cards specifying motion acting on grid points define motion inputs between two reference markers in the model. The motion input may be either translational or rotational.

Motion for both grid and joint based cards can be defined as displacement, velocity, or acceleration.

How to set up a MOTION load collector:

1. The MOTION card in HyperMesh is a load collector that takes all the motion sets from MOTNG, MOTNGC, MOTNGE, MOTNJ, MOTNJC, MOTNJE cards. First, go to the load collector panel by clicking the button above the Project Menu.

2. In the load collector creation panel, set the radio button on the left to “create”, and name the load collector by typing in the “loadcol name =” box.

3. Set the card image to MOTION by clicking on the “card image =” button.
4. Click “create/edit” to edit your load collector.

5. Click on the yellow “M(1)” button, and choose the name of your load collector.

6. After clicking the appropriate load collector, you will return to the “create/edit” panel. The number underneath the yellow button is the load collector id, or SID. This number is used when defining motion cards.

How to add grid point motion:

MOTNG

7. Adding a MOTNG card to a HyperMesh model involves the use of control cards. In the analysis panel of the Project Menu, click on “control cards”.
8. On the first page of options, click on the button labeled “BULK_UNSUPPORTED_CARDS”.

9. This will bring up a control card edit window. This window is where you define many different cards, and has a specific format. To view this format, follow this link.

10. Begin the first field with the name of the card, MOTNG, followed by the Load Set Identification Number (SID), Grid Point Identification Number (G1), Component Number (C1), Grid Point ID to define relative motion (G2), Scale factor (D), Initial displacement (if velocity or acceleration) (D0), and Initial velocity (if acceleration) (V0). Note: You only need to enter values in fields that are applicable to your model.

11. Click “ok”. You will notice that the “BULK_UNSUPPORTED_CARDS” button is now green. Additionally, there will be a new card in the Model Tree. Click “return” to go back to the main Project Menu.

12. By defining the MOTNG card with the SID number of the MOTION load collector, it will automatically be included with it in the simulation.

MOTNGC

1. The process for creating a MOTNGC card is almost the same as for a MOTNG card. Follow the steps for creating a MOTNG card, replacing the values with the following:

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOTNGC</td>
<td>SID</td>
<td>G1</td>
<td>C1</td>
<td>G2</td>
<td>CVID</td>
<td>INT</td>
<td>EID</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>D0</td>
<td>V0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. CVID is the set identification number of the MBCRV entry that gives the motion vs. time. INT is the interpolation type (LINEAR, CUBIC, AKIMA), where AKIMA is the default. EID is the set identification number of the MBVAR for the independent variable expression. An example entry would look like:
How do I?

MOTNGE

1. The process for creating a MOTNGE card is almost the same as for a MOTNG card. Follow the steps for creating a MOTNG card, replacing the values with the following:

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOTNGE</td>
<td>SID</td>
<td>G1</td>
<td>C1</td>
<td>G2</td>
<td>EID</td>
<td>D0</td>
<td>V0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. EID is the expression identification number of the MBVAR entry that gives the motion vs. time. An example entry would look like:

MOTNGE, 4, 75, 3, , 1, AKIMA, 2

How to add joint motion:

MOTNJ

1. The process for creating a MOTNJ card is almost the same as for a MOTNG card. Follow the steps for creating a MOTNG card, replacing the values with the following:

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOTNJ</td>
<td>SID</td>
<td>JID</td>
<td>MTYPE</td>
<td>D</td>
<td>DTYPE</td>
<td>D0</td>
<td>V0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. JID is the element id of the joint to which the motion is being applied. MTYPE is TRANS or ROT to define the motion for a cylindrical joint, but is ignored for translational and revolute joints. D is the value of motion. DTYPE is the motion data type (DIS, VEL, or ACC), where blank defaults to displacement motion. An example entry would look like:
How do I?

MOTNJC

1. The process for creating a MOTNJC card is almost the same as for a MOTNG card. Follow the steps for creating a MOTNG card, replacing the values with the following:

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOTNJC</td>
<td>SD</td>
<td>JID</td>
<td>MTYPE</td>
<td>CVID</td>
<td>INT</td>
<td>EID</td>
<td>DTYPE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>D0</td>
<td>V0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. JID is the element identification number for the joint to which motion is being applied. MTYPE is TRANS or ROT to define the motion for a cylindrical joint, but is ignored for translational and revolute joints. CVID is the set identification number of the MBCRV entry that gives the motion vs. time. INT is the interpolation type (LINEAR, CUBIC, AKIMA), where AKIMA is the default. EID is the set identification number of the MBVAR for the independent variable expression. DTYPE is the motion data type (DIS, VEL, or ACC), where blank defaults to displacement motion. An example entry would look like:

MOTNJ, 4, 2771675, TRANS, 0.8, DIS

MOTNJE

1. The process for creating a MOTNJE card is almost the same as for a MOTNG card. Follow the steps for creating a MOTNG card, replacing the values with the following:

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOTNJE</td>
<td>SID</td>
<td>JID</td>
<td>MTYPE</td>
<td>EID</td>
<td>DTYPE</td>
<td>D0</td>
<td>V0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
2. JID is the element identification number for the joint to which motion is being applied. MTYPE is TRANS or ROT to define the motion for a cylindrical joint, but is ignored for translational and revolute joints. EID is the set identification number of the MBVAR for the independent variable expression. DTYPE is the motion data type (DIS, VEL, or ACC), where blank defaults to displacement motion. An example entry would look like:

```
H0TNJE, 4, 675, TRANS, 1
```

Example Model:

Comments:

1. Joint motion can only be applied to cylindrical joints (MTYPE = TRANS or ROT), revolute joints (MTYPE = ROT), and translational joints (MTYPE = TRANS).
2. A CID of zero or blank references the basic coordinate system.
3. The component numbers 1, 2, 3 belong to the three displacement directions; 4, 5, 6 to the three rotations; 7, 8, 9 to the three translational velocities; and 10, 11, 12 to the three angular velocities; 13, 14, 15 to the three translational accelerations; and 16, 17, 18 to the three angular acceleration.
4. If G1 and G2 are defined, the motion is a relative motion between the two grid points; if G2 is blank, absolute motion of G1 is defined.
5. This card is represented as a constraint load in HyperMesh.

Additional Info:

file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/motion_bulk_data.htm

file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/motiong.htm

file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/motngc.htm

file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/motnge.htm

file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/motnj.htm

file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/motnjc.htm

file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/motnje.htm
How do I create a Bushing?

Brief introduction to bushings
A Bushing entity defines a force and torque acting between two Bodies. The force and torque consist of three major effects: a spring force, a damping force, and a pre-load vector.

The spring force is defined by the product of the stiffness matrix and the relative displacement between the two bodies - Body 1 and Body 2. The damping force is defined by the product of the damping matrix and the relative velocity between the two bodies - Body 1 and Body 2. A preload vector can also be added to the spring and damping forces. The six components (three forces and three moments) are defined in the coordinate system of the Body 2 Reference Marker.

The Bushing elements are used as compliant connectors in mechanical systems. Any Joint (Lower and Higher Pairs) built in MotionView can be made Compliant and made so they in turn are represented as Bushings.

Bushings are typically used to reduce vibration and noise, absorb shock, and accommodate misalignments

How to create a bushing
A bushing entity in HyperMesh is represented as a spring element. From the 1D Project Menu panel, click on the “springs” button.

Click on “element types =”, and select CMBUSH from the table.

Click on “property =”, and select a property from the list.
Specify the first node by selecting it from the Model Window, then specify the second node from the Model Window. This will automatically create a bushing. Click the “return” button to go back to the Project Menu.

To define bushings using a curve, expression, table, or marker, follow the formats included in the help sheets linked below.

**Example Model**

![Slider_crank.fem](file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/cmbushc.htm)

This slider crank model uses a bushing between the crank and connecting rod, in order to introduce additional degrees of freedom into the system.

**Comments**

1. Element identification numbers must be unique with respect to all other element identification numbers.
2. This card is represented as a spring element in HyperMesh.

**Additional Info**

[file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/cmbushe.htm](file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/cmbushe.htm)

[file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/cmbusht.htm](file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/cmbusht.htm)

[file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/cmbushm.htm](file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/cmbushm.htm)
How do I specify more than one Method of Analysis?

Brief introduction to Analysis Methods
Multi body systems are analyzed through different methods, depending on the load case. A car starting from rest might be analyzed using static analysis to identify static equilibrium followed by a transient analysis to study vehicle dynamics. In HyperMesh, there are three methods of simulation analysis supported; transient, static, and quasi static.

How to specify more than one Method of Analysis
The MBSIM card can be used to specify analysis types. In any given system, when more than one type of analysis is required, multiple MBSIM cards are used (each with a specific analysis type) which are referenced in sequence by a single MBSEQ card.

1. Begin by creating a load collector. Click on the load collector icon above the Project Menu.
2. Ensure that the radio option is set to “create”, and name the load collector.
3. Set the card image to MBSIM by clicking on the “card image =” button, and selecting MBSIM from the table.
4. Click the “create/edit” button on the right, and set the simulation type to transient.
5. Set the TIME to 5, and the DELTA to 0.01. Notice the ID number of the card in the upper left corner of the window. Click the “return” button to go back to the load collector panel.
6. Create the next load collector the same way as the first.
7. Click the “create/edit” button.
8. Set the Simulation Type to “Static”, and leave the values as default. Note the ID of the load collector. Hit the “return” button to go back to the load collector menu.

9. Define a third load collector in the same method as the first two. Click the “create/edit” button, and set the simulation to “Transient”.

10. Change the TIME value to 5, and the DELTA value to 0.01, then click “return”.

11. Create a fourth load collector, but instead of MBSIM, set the card image to MBSEQ.

12. Click the “create/edit” button, and you will see a new window. In the bottom of the window, change the MBSEQ_NUM_SID value to 3. This will create 2 new yellow boxes at the top of the window.

13. Each yellow button will hold the id of a load collector previously defined. Click each button to select the appropriate load collector from the list. It is important to keep the load collector id’s in the correct order, or the simulation will not run correctly.

14. Click the “return” button twice to go back to the Project Menu.
15. Create a new load step by clicking on the “Load Step” button in the analysis panel. Name the load step, and set the type to multi-body dynamics.

16. Check the box next to “MBSIM”, and click on the “=” button that appears next to it.

17. From the menu, choose the load collector with the MBSEQ card.

18. Click the “create” button, and “return” to go back to the Project Menu.

Example Model

Comments

1. The continuation card is used to distinguish between either a TRANS or STAT simulation type. The reader will look for appropriate options based on the type of simulation specified.

2. When the simulation type is static (STAT), the solver will perform a static simulation if the termination type, termination time/duration, step type, delta/nsteps are not provided. A quasi-static simulation will be performed if that information is provided.

3. If the output step type is DELTA, then the next argument is expected to be a positive real value, and it is the output step time during the simulation run. If the step type is NSTEPS, then the next argument is expected to be a positive integer value which will be the number of output steps during the simulation. If the step type is PRINCR, then the next argument is expected to be a positive integer value which will be print increment. Solver will output at every intermediate print increment value. If the PRINCR is set to 1, then the solver will output intermediate results at every integrator step.

4. ITYPE, DTOL, H0, HMAX, HMIN, VTOLFAC, MAXODR, DAEIDX, DCNTOL, DCRMXIT, DCRMNIT, DVCTRL, DJACEVL, and DEVLEXP are only applicable for TRANS simulation types. The 2nd continuation card (DAEIDX, DCNTOL, DCRMXIT, DCRMNIT, DVCTRL, DJACEVL, and DEVLEXP) are available when ITYPE = DSTIFF. See the comments in the Param_Transient in the online help for more details.

5. KETOL, DQTOL, NITER are only applicable for STATIC simulation type.

6. RESTOL, FITOL, NITER, TLIMIT, ALIMIT are applicable for the force imbalance static method used for quasi-static solutions.

7. Static solver type is used to select from the two static solution types currently offered. FIM represents the Force Imbalance Method and MKM represents the Maximum Kinetic Energy Attrition Method.

8. Note that when quasi-static simulation is requested (STAT with termination time), then the STTYPE option is ignored and the quasi-static simulation will be performed using the force imbalance method.

9. Every time an MBSIM card is encountered, a simulation is performed on the multi-body model. The initial condition for the next MBSIM will be the ending condition of the current simulation.

10. This card is represented as a loadcollector in HyperMesh.
How do I define Units?

Brief introduction to Units
Unit systems in HyperMesh are defined with the DTI,UNITS control card. The DTI,UNITS card can specify four different types of units: Mass, Length, Time, and Force. The last is obviously a function of the first three and can be automatically calculated for a consistent set of units. However, many unit systems are not “consistent”, so it may be necessary to specify a force unit also.

How to define Units
To set the units for a system, click the “control cards” button in the “analysis” panel of the Project menu.

Click on “DTI_UNITS”.

On the buttons under “MASS”, “FORCE”, “LENGTH”, and “TIME”, click the buttons and select the desired units from the menu.

Click “return” and you will see that the “DTI_UNITS” button has been highlighted green. This means that the control card for DTI,UNITS has been enabled. Editing a control card automatically enables the card. To enable or disable a card, click the “enable” or “disable” button on the right of the menu, and click on
the button of the control card you wish to adjust. Enabling a card will turn the button green, while disabling it will turn the button red. To delete a card from the system, click the “delete” button, then click on the control card to be deleted. After deletion, a card will return to the default color of the menu, and is no longer included in the system.

Click “return” again to go back to the Project Menu.

### Example model

![Slider crank model](image)

Slider_crank.fem

This slider crank model uses adjusted units of measurement when solving.

### Comments

1. PARAM, WTMASS is ignored for the multi-body and component mode synthesis (flexible-body preparation) solution sequences. Unit data for these solution sequences is supplied on the DTI, UNITS bulk data entry.

2. This card is represented as a control card in HyperMesh.

### Additional info

[file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/dti_units.htm](file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/dti_units.htm)

[file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/radUnits.htm](file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/rad_units.htm)
How do I create a Spring Damper?

Brief introduction to Spring Dampers
A Spring Damper entity defines a spring damper acting between two Bodies. The element can be a Coil Spring - translational (applies a force) or Torsion Spring - rotational (applies a torque). In both cases, the force is characterized by a stiffness coefficient, a damping coefficient, a free-length and a preload. Spring dampers can be found in automobile shock absorbers, anti-vibration mounts, hinge dampers, and door stoppers.

How to create a Spring Damper
To define a spring damper, click on the “springs” button from the 1D panel in the project menu.

Click on “element types =”, and select CMSPDP from the table.

Click on “property =”, and select a property from the list.

Specify the first node by selecting it from the Model Window, then specify the second node from the Model Window. This will automatically create the spring damper. Click the “return” button to go back to the Project Menu.

Other Methods
Spring Dampers can also be defined through curves, expressions, tables, and markers. To define a spring damper one of these ways, bulk data cards must be used. A detailed guide to bulk data cards can be found here.
To define a spring damper through a curve, input the following data into the bulk entry:

<p>| | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>(2)</td>
<td>(3)</td>
<td>(4)</td>
<td>(5)</td>
<td>(6)</td>
<td>(7)</td>
<td>(8)</td>
<td>(9)</td>
<td>(10)</td>
</tr>
<tr>
<td>CMSPDPC</td>
<td>EID</td>
<td>KCID</td>
<td>G1</td>
<td>G2</td>
<td>BCID</td>
<td>L</td>
<td>PF</td>
<td>TYPE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>KNT</td>
<td>KED</td>
<td>BINT</td>
<td>BEID</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Where EID is the Element Identification number, KCID is the Stiffness Curve ID, G1 is a Grid point ID, G2 is a Grid point ID, B is the Damping Curve ID, L is the Unstretched length of the spring damper, PF is the Preload Force, TYPE is TRANS or ROT (default is TRANS), KNT is the Stiffness Interpolation type (LINEAR, CUBIC, AKIMA, default is AKIMA), KED is MBVAR ID for independent variable ID for stiffness, BINT is the Damping Interpolation type (LINEAR, CUBIC, AKIMA, default is AKIMA), and BEID is MBVAR ID for independent variable ID for damping. Here is an example:

<p>| | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>(2)</td>
<td>(3)</td>
<td>(4)</td>
<td>(5)</td>
<td>(6)</td>
<td>(7)</td>
<td>(8)</td>
<td>(9)</td>
<td>(10)</td>
</tr>
<tr>
<td>CMSPDPC</td>
<td>3</td>
<td>34</td>
<td>223</td>
<td>324</td>
<td>0</td>
<td>1.0</td>
<td>TRANS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AKIMA</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

To define a spring damper through an expression, input the following data into the bulk entry:

<p>| | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>(2)</td>
<td>(3)</td>
<td>(4)</td>
<td>(5)</td>
<td>(6)</td>
<td>(7)</td>
<td>(8)</td>
<td>(9)</td>
<td>(10)</td>
</tr>
<tr>
<td>CMSPDPE</td>
<td>EID</td>
<td>KEID</td>
<td>G1</td>
<td>G2</td>
<td>BEID</td>
<td>L</td>
<td>PF</td>
<td>TYPE</td>
<td></td>
</tr>
</tbody>
</table>

Where KEID is the MBVAR ID for stiffness expression. All other values are the same as for CMSPDPC. Here is an example:

<p>| | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>(2)</td>
<td>(3)</td>
<td>(4)</td>
<td>(5)</td>
<td>(6)</td>
<td>(7)</td>
<td>(8)</td>
<td>(9)</td>
<td>(10)</td>
</tr>
<tr>
<td>CMSPDPE</td>
<td>3</td>
<td>3</td>
<td>223</td>
<td>324</td>
<td>4</td>
<td>1.0</td>
<td>TRANS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

To define a spring damper through a table, use the following format:
How do I?

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMSPDPT</td>
<td>ED</td>
<td>KTID</td>
<td>G1</td>
<td>G2</td>
<td>BTID</td>
<td>L</td>
<td>PF</td>
<td>TYPE</td>
<td></td>
</tr>
</tbody>
</table>

Where KTID is the Table ID for stiffness, and BTID is the Table ID for Damping. All other values are the same as for CMSPDPC. Here is an example:

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMSPDPT</td>
<td>3</td>
<td>3</td>
<td>223</td>
<td>324</td>
<td>4</td>
<td>1.0</td>
<td>TRANS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

To define a spring damper using markers, use the following format:

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMSPDPM</td>
<td>ED</td>
<td>K</td>
<td>M1</td>
<td>M2</td>
<td>B</td>
<td>L</td>
<td>PF</td>
<td>TYPE</td>
<td></td>
</tr>
</tbody>
</table>

Where M1 and M2 are Marker Identification numbers and K is the Stiffness Value. All other values are the same as for CMSPDPC. Here is an example:

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMSPDPM</td>
<td>3</td>
<td>34.5</td>
<td>223</td>
<td>324</td>
<td>0.0</td>
<td>1.0</td>
<td>TRANS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example Model

Slider_crank.fem
This slider crank mechanism uses a spring damper attached to the slider opposite the crank mechanism, and resists the movement of the slider.

**Comments**
1. Element identification numbers must be unique with respect to all other element identification numbers.
2. The spring damper force is along the line segment connecting the grids G1 and G2, or markers M1 and M2.
3. The positive preload force is a stretching force.
4. This card is represented as a spring element in HyperMesh.

**Additional info**
file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/cmspd.htm
How do I create a coupler?

Brief introduction to couplers
A Coupler entity defines an algebraic relationship between the degrees of freedom of two or three joints. This constraint element may be used to model idealized spur gears, rack and pinion gears, and differentials as simple constraints that relate the displacements in a set of joints. Couplers can be used to specify relationships between Translational, Rotational and Cylindrical Joints only.

How to create a coupler
A coupler is a constraint that can be defined by a control card. From the “analysis” page in the Project Menu, click the “control cards” button.

<table>
<thead>
<tr>
<th>vectors</th>
<th>load types</th>
<th>interfaces</th>
<th>control cards</th>
</tr>
</thead>
<tbody>
<tr>
<td>systems</td>
<td>constraints</td>
<td>rigid walls</td>
<td>output blocks</td>
</tr>
<tr>
<td>equations</td>
<td>temperatures</td>
<td>entity sets</td>
<td>load steps</td>
</tr>
<tr>
<td>forces</td>
<td>load on gear</td>
<td>contact links</td>
<td>optimization</td>
</tr>
<tr>
<td>moments</td>
<td>bodies</td>
<td>bodies</td>
<td>Padloss</td>
</tr>
<tr>
<td>pressures</td>
<td>num</td>
<td>OptStruct</td>
<td></td>
</tr>
</tbody>
</table>

From the “control cards” menu, click on the “BULK_UNSUPPORTED_CARDS” button.

From the Control Cards window, input the following values in order:

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>COUPLER</td>
<td>COD</td>
<td>JID1</td>
<td>TYPE1</td>
<td>RATIO1</td>
<td>JID2</td>
<td>TYPE2</td>
<td>RATIO2</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>JID3</td>
<td>TYPE3</td>
<td>RATIO3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Where COID is the coupler ID, JIDi is the joint id number, TYPEi is TRA or ROT, and RATIOoi are coefficients of the coupler restraint equation. An example is shown below:

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>COUPLER</td>
<td>3</td>
<td>1</td>
<td>T</td>
<td>2.0</td>
<td>4</td>
<td>R</td>
<td>1.0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Click “ok” to confirm the coupler definition and to close the Control Cards window.
Example model

Comments
1. COUPLERs are only valid in a multi-body solution sequence.
2. At least JID1 and JID2 need to be defined.
3. The type is optional if the Joint is revolute or translation. But if the joint is cylindrical, the type should be set to TRA to denote that the translational motion is coupled or ROT to specify that the rotational motion is coupled.

Additional info
file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/coupler_bulk_data.htm
How do I create a Marker?

Brief introduction to Markers
A Marker entity is an orthonormal, right-handed coordinate system and reference frame in MotionView. A Marker must belong to a body. The body can be any type: rigid, flexible, or point mass. The Ground Body (Newtonian reference frame) is considered to be a special case of a rigid body and thus the default marker that belongs to the Ground Body is called Global Frame.

Markers can be either fixed or floating. A fixed marker gets attached to a body and moves with that body. To attach a marker on a body, you must specify its location and orientation with respect to the body.

Markers have many different uses in Modelling. They can define the graphical boundary of a body, provide points and reference frames for reaction force application and are used for other location and orientation functions. Some forces and constraints in use floating markers to specify points for applying forces on bodies. There is no need to specify a fixed position for a floating marker. This position and orientation can change during the simulation. Its position and orientation time history is determined by the force or joint element it is used to define.

How to create a Marker
Markers can be defined in HyperMesh through the 1D panel in the Project menu.

Click on the “markers” button in the 1D panel

The radio button on the left should be set to “create”. Give a name to your marker by typing it in the text field next to “marker name =”.

The yellow “node” button should be highlighted. Click on the node in your model to define its location, and the “system” button will be highlighted. To define an orientation system, click on the global coordinates system displayed in the bottom left of the Model Window.
Click the “create” button to define your marker, then go back to the main Project menu with the “return” button.

Example Model

Excavator_with_joints(2).fem
Comments

Each marker has a unique ID. The marker gets its location from the grid corresponding to the GiD and the orientation from the coordinate system (CID).

1. Multiple markers can be located at the same grid with the same or with a different CID.

2. This card is represented as a sensor in HyperMesh.

Additional info

file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/marker.htm
How do I create a Curve?

Brief introduction about curve:

Curve entities are used to define and store xy data and are associated with a plot entity. MBCRV defines the data of a curve. It represents a curve in HyperMesh.

Method 1- using BULK_UNSUPPORTED_CARDS:

- In the analysis page of the project menu click on “Control Cards”

- On the first page of options, click on the “BULK_UNSUPPORTED_CARDS” button.

- This will open the control card edit window where bulk data entries can be defined

- Begin the first field of with the name of the card, MBCRV, followed by the curve identification number CVID, Extrapolating the curve using a linear function instead of a parabolic function LINEAR( blank is default), Independent curve data x1, x2, x3, …, xn and dependent curve data y1,y2,y3,…, yn.

- Format of the control card is shown below

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MBCRV</td>
<td>CVID</td>
<td>LINEAR</td>
<td>X1</td>
<td>Y1</td>
<td>X2</td>
<td>Y2</td>
<td>…</td>
<td>…</td>
<td>Xn</td>
</tr>
</tbody>
</table>
Method 2 – Using GUI:

- In the Post page of the project menu, click on “xy plots”

- On the page of options, click on “plots”

- Enter a name in the field “plot =” and click on create to notice a plot created on the top left corner of the graphics area
How do I?

- Click return and then click on “edit curves” and make sure the radio button on the left is set to “create”

- Curves can be created by two options either load an existing csv file or by using the math option

- To create curve using the first method, select the “file” radio button and click on “load” to load an already existing csv file.

  ![Image of software interface](image1)

  - Change the radio button to “y”, click on “comp” and select “column 2” from the given list.

  ![Image of software interface](image2)

  - Click on “create”. You can notice that the curve has been created in the plot. Now click on “return” and “exit”

  ![Image of curve](image3)
Another way to create a curve is using the math option. Activate the “math” radio button.

Enter the range of x and y in step increments in the respective fields. Initial value: final value: step increment.

A curve is created in the plot to the top left corner of the graphics area.

Click on “return” and then “exit”

Now, click on the Card edit icon in the tool bar and change the entity selection to curves.
• Click on the “curves” button and select the appropriate curve from the given list of curves.

• Click on edit and the MBCRV card image appears.

• Change the type to “LINEAR”.

• Click “return” twice.
How do I create a Force or Moment?

Brief introduction to Forces and Moments

A Force entity defines a general force and/or torque acting one body (Action Only) or between two bodies (Action Reaction). It can be used to model a force (Translational), a torque (Rotational), or both (TransRotational). The force and/or torque vectors are defined by their three (or six) components with respect to a third Reference Marker. The components may be defined using a Linear Value, MotionSolve expressions or a user defined subroutine. They may be a function of any system state and time. The Force element can also be used to define as a single component force or torque between two bodies. The single component force or torque can be defined linear scalar value

- There are two kinds of forces: Action only forces which are applied to one point and action-reaction forces which are applied at two points.
- Force components can be a constant value, a curve, an expression, or a user written subroutine.
- Forces are always defined at grid points and can be applied to one grid point (action only) or two (action-reaction).
- The bulk data entry $0$, defines constant force, MBFRCC defines force by a curve and MBFRCE defines a force by equation.
- Moments are always defined at grid points, and can be applied to one grid point (action-only) or two (action-reaction). The bulk data entry MBMNT defines constant moment; the entry MBMNTC defines moment by a curve; and the entry MBMNTE defines a moment by equation.
- The entry MLOAD can be used to derive force moment set combinations.
How to set up a load collector:

- The MLOAD card in HyperMesh is a load collector that takes all the load sets from GRAY, MBFRCC, MBFRCE, MBMNT, MBMNTC, MBMNTE, MBSFRC, MBSFRCC, MBSFRCE, MBSMNT, MBSMNTC, MBSMNTE, and MLOAD bulk data entries.

- Go to the load collector panel by clicking the button above the project menu.

- In the load collector panel, set the radio button on the left to “create” and name the load collector panel by typing in the “loadcol name= ” box.

- Select the “no card image” option by clicking on the down arrow to the left of “card image= ” button.

- Use the ID of this load collector in MBFRCC/MBFRCE/MBFRCE definition.

- Now create another load collector by typing a name in the “loadcol name= ” box and selecting MLOAD as the card image.

- Click on “Create/Edit”

- Click on the yellow L(1) button and choose the name of your load collector from the list shown. Select the previously created load collector without the card image.
After selecting the suitable load collector, you will return to the “create/edit” panel. The number under the yellow button is the load collector id, or SID. This number is used while defining load cards.

Refer the MLOAD load collector while defining the load step.

How to add grid point Force (MBFRC)

Method 1 – Using GUI:

- Click BCs > Create > Forces to open the Forces panel. Or click on forces button in the analysis page.
How do I?

- Make sure nodes are selected from the entity selection switch.
- Set the coordinate system toggle to global system.
- Enter the magnitude of the force in the “magnitude =” box and select a suitable axis from the pop-up menu.
- Change the “load types” to MBFRC by selecting it from the pop-up menu.

![Image](image.png)

- Click create and then click return to go to the main menu

Method 2 – Using BULK_UNSUPPORTED_CARDS:

- Adding MBFRC card to a HyperMesh model is similar to adding MOTNG card. In the analysis panel of the Project menu, click on “control cards”

<table>
<thead>
<tr>
<th>Vectors</th>
<th>Load Types</th>
<th>Interfaces</th>
<th>Control Cards</th>
</tr>
</thead>
<tbody>
<tr>
<td>systems</td>
<td>constants</td>
<td>rigid walls</td>
<td>output block</td>
</tr>
<tr>
<td></td>
<td>equations</td>
<td>entity sets</td>
<td>load types</td>
</tr>
<tr>
<td></td>
<td>forces</td>
<td>blocks</td>
<td></td>
</tr>
<tr>
<td></td>
<td>moments</td>
<td>contact springs</td>
<td>optimization</td>
</tr>
<tr>
<td></td>
<td>pressures</td>
<td>bodies</td>
<td>number of elements</td>
</tr>
</tbody>
</table>

- On the first page of options, click on the “ BULK_UNSUPPORTED_CARDS” button

<table>
<thead>
<tr>
<th>ANALYSIS</th>
<th>DEBUG</th>
<th>DMasename</th>
<th>ECHO</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASSEMBLY</td>
<td>DENSITY</td>
<td>DT_UNITS</td>
<td></td>
</tr>
<tr>
<td>BDDG</td>
<td>DENSES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BULK, UNSUPPORTED, CARDS</td>
<td>DESIGNS</td>
<td>EKNAME</td>
<td></td>
</tr>
<tr>
<td>CASE, UNSUPPORTED, CARDS</td>
<td>DESVARG</td>
<td>ELEMISSUAL</td>
<td></td>
</tr>
<tr>
<td>CHECK</td>
<td>DOGLOBAL</td>
<td>FORMAT</td>
<td></td>
</tr>
</tbody>
</table>

- This will open the control card edit window where different Bulk Data entries can be defined. The format can be viewed by following
• Begin the first field with the name of the card, MBFRC, followed by the load set identification number (SID), Grid point identification number where the action force is applied (G1), Coordinate system identification number (CID) (blank or 0 infer the basic coordinate system), Force magnitude F, Grid point identification number (G3) to optionally supply N1, N2, N3 in conjunction with G1. (N1; N2; N3) to define the direction of the force vector (At least one of the vector components must be non-zero), Grid point identification number (G3) if reaction force is applied (blank or zero when the force is action-only force), Grid point identification (G4) number whose parent body hosts the coordinate system with respect to which the force is defined. The force vector changes direction with the orientation of the body. Note: you only need to enter values in fields that are applicable to your model.

• Format of the control card is as shown below

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MBFRC</td>
<td>SID</td>
<td>G1</td>
<td>CID</td>
<td>F</td>
<td>G3/N1</td>
<td>N2</td>
<td>N3</td>
<td>G2</td>
<td>G4</td>
</tr>
</tbody>
</table>

• Click “ok”. You will notice that the “BULK_UNSUPPORTED_CARDS” button is now green. Additionally, there will be a new card in the Model Tree. Click “return” to go back to the main project menu.

• By defining MBFRC card with the SID number of the MLOAD load collector, it will automatically be included with it in simulation.
How to add grid point Force (MBFRCC)

Force MBFRCC Defines a curve force at a grid point by specifying a vector

Method 1 – Using GUI:

- Click **BCs > Create > Forces** to open the Forces panel. Or click on forces button in the analysis page.

- Click the down arrow button to the left of “magnitude=” and select “curve, vector” from the pop-up menu.

- Make sure **nodes** are selected from the entity selection switch.

- Set the coordinate system toggle to global system.

- Enter the magnitude of the force in the “magnitude =” box and select a suitable axis from the pop-up menu.

- Select an MBD curve by clicking on the “curve” button.

- Change the “load types” to **MBFRCC** by selecting it from the pop-up menu.

- Click **create** and then click **return** to go to the main menu.

Method 2 – Using **BULK_UNSUPPORTED_CARDS:**

- In the Analysis page of project menu click on “control cards”
How do I?

- On the first page of options, click on the “BULK_UNSUPPORTED_CARDS” button.

  ![Image of control card edit window]

  - This will open the control card edit window where different Bulk Data entries can be defined.

  - Begin the first field with the name of the card, `MBFRCC`, followed by the load set identification number (SID), Grid point identification number where the action force is applied (G1), Coordinate system identification number (CID) (blank or 0 infer the basic coordinate system), Set identification number CVID of MBCRV entry that gives the load vs. independent variable measured in the coordinate system defined by CID, Grid point identification number (G3) to optionally supply N1, N2, N3 in conjunction with G1. (N1; N2; N3) to define the direction of the force vector (At least one of the vector components must be non-zero), Grid point identification number G2 where the reaction force is applied (if blank or 0, then the force is an action-only force), Interpolation type INT - Default = AKIMA (Character: LINEAR, CUBIC, AKIMA), Set identification number EID of the MBVAR for the independent variable expression, Grid point identification number G4 whose parent body hosts the coordinate system with respect to which the force is defined. The force vector changes direction with the orientation of the body.

- Format of the control card is as shown below

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MBFRCC</td>
<td>SID</td>
<td>G1</td>
<td>CID</td>
<td>CVID</td>
<td>G3/N1</td>
<td>N2</td>
<td>N3</td>
<td>G2</td>
<td></td>
</tr>
<tr>
<td>INT</td>
<td>EID</td>
<td>G4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
How do I?

Click “ok”. You will notice that the “BULK_UNSUPPORTED_CARDS” button is now green. Additionally, there will be a new card in the Model Tree. Click “return” to go back to the main project menu.

By defining MBFRCC card with the SID number of the MLOAD load collector, it will automatically be included with it in simulation.

How to add grid point Force (MBFRCE)

Expression Force for Multi-body Solution Sequence

Method 1 - Using BULK_UNSUPPORTED_CARDS:

Firstly, create a load collector with no card image by entering the name in the “loadcol name=” field.

In the Analysis page of project menu click on “control cards”

On the first page of options, click on the “BULK_UNSUPPORTED_CARDS” button.
• This will open the control card edit window where different Bulk Data entries can be defined.

• In the control card window, give MBVAR entry by beginning the first field with the name of the card, MBVAR, followed by the Variable identification number VID i.e. the number against the load collector created in the first step and then a character string expression

• Begin the first field with the name of the card, MBFRCE, followed by the load set identification number (SID), Grid point identification number where the action force is applied (G1), Coordinate system identification number (CID) (blank or 0 infer the basic coordinate system), Set identification number EID of the MBVAR entry that gives the load measured in the coordinate system defined by CID, Grid point identification number (G3) to optionally supply N1, N2, N3 in conjunction with G1. (N1; N2; N3) to define the direction of the force vector (At least one of the vector components must be non-zero), Grid point identification number G2 where the reaction force is applied (if blank or 0, then the force is an action-only force), Grid point identification number G4 whose parent body hosts the coordinate system with respect to which the force is defined. The force vector changes direction with the orientation of the body.

• Format of the control card is as shown below

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MBFRCE</td>
<td>SID</td>
<td>G1</td>
<td>CID</td>
<td>EID</td>
<td>G3/N1</td>
<td>N2</td>
<td>N3</td>
<td>G2</td>
<td>G4</td>
</tr>
</tbody>
</table>

• Click “ok”. You will notice that the “BULK_UNSUPPORTED_CARDS” button is now green. Additionally, there will be a new card in the Model Tree. Click “return” to go back to the main project menu.

• By defining MBFRCE card with the SID number of the MLOAD load collector, it will automatically be included with it in simulation.
How do I?

How to add grid point Moment (MBMNT)

Method 1 – Using GUI:

- Click **BCs > Create > Moments** to open the moments panel. Or click on forces button in the analysis page.

  - Make sure **nodes** are selected from the entity selection switch.
  - Set the coordinate system toggle to global system.
  - Enter the magnitude of the moment in the “magnitude =” box and select a suitable axis of rotation from the pop-up menu.
  - Choose the axis about which the moment has to be applied using either the N1,N2,N3 or the Vector Option
  - Change the “load types” to **MBMNT** by selecting it from the pop-up menu.

- Click **create** and then click **return** to go to the main menu

Method 2 – Using BULK_UNSUPPORTED_CARDS:

- Adding MBMNT card to a HyperMesh model is similar to adding MOTNG card. In the analysis panel of the Project menu, click on “control cards”
On the first page of options, click on the “BULK_UNSUPPORTED_CARDS” button. This will open the control card edit window where different Bulk Data entries can be defined.

Begin the first field with the name of the card, MBFRC, followed by the load set identification number (SID), Grid point identification number where the action moment is applied (G1), Coordinate system identification number (CID) (blank or 0 infer the basic coordinate system), Moment magnitude M, Grid point identification number (G3) to optionally supply N1, N2, N3 in conjunction with G1. (N1; N2; N3) to define the direction of the axis of the moment (At least one of the vector components must be non-zero), Grid point identification number (G3) if reaction moment is applied (bank or zero when the moment is action-only moment).

Format of the control card is as shown below:

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MBMNT</td>
<td>SID</td>
<td>G1</td>
<td>CID</td>
<td>M</td>
<td>G3/N1</td>
<td>N2</td>
<td>N3</td>
<td>G2</td>
<td></td>
</tr>
</tbody>
</table>

Control Card

```
MBMNT, 3, 345, , 100, 0.0, 1.0, 0.0, , ,
```
How do I?

- Click “ok”. You will notice that the “BULK_UNSUPPORTED_CARDS” button is now green. Additionally, there will be a new card in the Model Tree. Click “return” to go back to the main project menu.

- By defining MBMNT card with the SID number of the MLOAD load collector, it will automatically be included with it in simulation.

How to add grid point Moment (MBMNTC)

Force MBMNTC Defines a curve moment at a grid point by specifying a vector

Method 1 – Using GUI:

- Click **BCs > Create > Moments** to open the moments panel. Or click on forces button in the analysis page

  - Click the down arrow button to the left of “magnitude=” and select “curve, vector” from the pop-up menu
  - Make sure **nodes** are selected from the entity selection switch.
  - Set the coordinate system toggle to global system.
  - Enter the magnitude of the moment in the “magnitude=” box and select a suitable axis from the pop-up menu.
  - Select an MBD curve by clicking on the “curve” button.
  - Change the “load types” to **MBMNTC** by selecting it from the pop-up menu.
How do I?

Method 2 – Using BULK_UNSUPPORTED_CARDS:

- In the Analysis page of project menu click on “control cards”

- On the first page of options, click on the “BULK_UNSUPPORTED_CARDS” button.

- This will open the control card edit window where different Bulk Data entries can be defined.

- Begin the first field with the name of the card, MBMNCTC, followed by the load set identification number (S1D), Grid point identification number where the action moment is applied (G1), Coordinate system identification number (C1D) (blank or 0 infer the basic coordinate system), Set identification number CVID of MBCRV entry that gives the load vs. independent variable measured in the coordinate system defined by C1D, Grid point identification number (G3) to optionally supply N1, N2, N3 in conjunction with G1. (N1; N2; N3) to define the direction of the axis of the moment (At least one of the vector components must be non-zero), Grid point identification number G2 where the reaction moment is applied (if blank or 0, then the moment is an action-only moment), Interpolation type INT - Default = AKIMA (Character: LINEAR, CUBIC, AKIMA), Set identification number E1D of the MBVAR for the independent variable expression

- Format of the control card is as shown below

- Click create and then click return to go to the main menu
Click “ok”. You will notice that the “BULK_UNSUPPORTED_CARDS” button is now green. Additionally, there will be a new card in the Model Tree. Click “return” to go back to the main project menu.

By defining MBMNTC card with the SID number of the MLOAD load collector, it will automatically be included with it in simulation.

**How to add grid point Moment (MBMNTE)**

Expression moment for Multi-body Solution Sequence

**Method 1 - Using BULK_UNSUPPORTED_CARDS:**

- Firstly, create a load collector with no card image by entering the name in the “loadcol name=” field.
- In the Analysis page of project menu click on “ control cards’
On the first page of options, click on the “BULK_UNSUPPORTED_CARDS” button.

This will open the control card edit window where different Bulk Data entries can be defined.

In the control card window, give MBVAR entry by beginning the first field with the name of the card, MBVAR, followed by the Variable identification number VID i.e. the number against the load collector created in the first step and then a character string expression

Begin the first field with the name of the card, MBMNTE, followed by the load set identification number (SID), Grid point identification number where the action moment is applied (G1), Coordinate system identification number (CID) (blank or 0 infer the basic coordinate system), Set identification number EID of the MBVAR entry that gives the load measured in the coordinate system defined by CID, Grid point identification number (G3) to optionally supply N1, N2, N3 in conjunction with G1. (N1; N2; N3) to define the direction of the axis of the moment (At least one of the vector components must be non-zero), Grid point identification number G2 where the reaction force is applied (if blank or 0, then the moment is an action-only moment.

Format of the control card is as shown below

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MBMNTE</td>
<td>SID</td>
<td>G1</td>
<td>CID</td>
<td>EID</td>
<td>G3/N1</td>
<td>N2</td>
<td>N3</td>
<td>G2</td>
<td></td>
</tr>
</tbody>
</table>
How do I?

Click “ok”. You will notice that the “BULK_UNSUPPORTED_CARDS” button is now green. Additionally, there will be a new card in the Model Tree. Click “return” to go back to the main project menu.

By defining MBMNTE card with the SID number of the MLOAD load collector, it will automatically be included with it in simulation.

Example models:

Slider crank with MBMNT control card: slider_crank_MBMNT.fem
Slider crank with MBFRC control card: SC_V_FRC.fem

Slider crank with MBMNTE control card: slider_crank_MBMNTE.fem
Slider crank with MBFRCE control card: slider_crank_MBFRCE.fem

Slider crank with MBFRCC: slider_crank_MBFRCC.fem
How do I?

Slider crank with MBMNT: slider_crank_MBMNTC.fem

Comments:

Additional Information:

file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/hwsolvers.htm?subcase_information_section_1.htm

file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/mbfrc.htm

file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/mbfrcc.htm

file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/mbfrce.htm

file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/mbmnt.htm

file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/mbmntc.htm

file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/mbmnte.htm
How do I create an Initial Condition?

Brief introduction to Initial Conditions

How to create an Initial Condition
The INVELB card is defined as a load collector, and can be defined through the load collector panel.

Go to the load collector panel by clicking the load collector button above the Project Menu.

Make sure the radio button on the left of the panel is set to “create”, and click on “card image =”. Select INVELB from the menu that pops up.

Give a name to the load collector, and click the “create” button to the right.

To create an initial velocity for a joint through the INVELJ card, the bulk data editor must be used.

From the analysis panel in the Project Menu, click on the “control cards” button.

From the control cards menu, click on the “BULK_UNSUPPORTED_CARDS” button.

In the Control Cards window that pops up, input the information in the following format:
How do I?

<table>
<thead>
<tr>
<th></th>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>INVELJ</td>
<td>SID</td>
<td>JID</td>
<td>JTYPE</td>
<td>VT</td>
<td>VR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Where SID is the Load Set ID number, JID is the Joint ID number, JTYPE is the Joint Type, VT is the Translational Velocity, and VR is the Rotational Velocity. An example is shown below:

<table>
<thead>
<tr>
<th></th>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
<th>(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>INVELJ</td>
<td>1</td>
<td>3</td>
<td>REV</td>
<td></td>
<td>10.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Click “ok” to create the control card, then click “return” to go back to the Project Menu.

Example model

Comments

1. Only one initial velocity per body can be defined in a load set.
2. Initial velocities defined by INVELB will overwrite initial velocities defined by INVELG.
3. A CID of zero or blank references the basic coordinate system.
4. Only one initial velocity per joint can be defined in a load set.
5. Initial velocities of joints defined by INVELJ will be overwritten by MOTNJ, MOTNJE, or MOTNJC.
6. This card is represented as a load collector in HyperMesh.

Additional info

file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/invelb.htm

file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/invelj.htm
**Bulk data input format**

Format for individual cards can be found here: file:///C:/Program%20Files/Altair/11.0/help/hwsolvers/bulk_data_section.htm

The following guidelines apply to all entries in the bulk data section:

- Data may contain 80 characters per line at most. All characters after the 80th are ignored. The only exception is for the INCLUDE data entry. SYSSETTING,CARDLENGTH can be used to change the number of characters allowed in each line.

- Each line of data contains up to nine fields in one of the three accepted formats:

  **Fixed Format**
  Each field consists of eight characters (shown below).

<table>
<thead>
<tr>
<th>----1----</th>
<th>----2----</th>
<th>----3----</th>
<th>----4----</th>
<th>----5----</th>
<th>----6----</th>
<th>----7----</th>
<th>----8----</th>
<th>----9----</th>
<th>----10----</th>
</tr>
</thead>
</table>

  **Large Field Fixed Format**
  Each field consists of 16 characters; two consecutive lines form nine fields, similar to other formats (shown below). Large field format is recognized by the first character after the keyword, or by the first character in each continuation line, which must be ‘*’. The second line (‘half line’), if present, must also contain ‘*’ in the first column. The first and last field in each half line is eight characters long. The last field on each first half-line and the first field on each second half-line are ignored.

<table>
<thead>
<tr>
<th>---1A----</th>
<th>---2----</th>
<th>---3----</th>
<th>---4----</th>
<th>---5----</th>
<th>---6----</th>
<th>---7----</th>
<th>---8----</th>
<th>---9----</th>
<th>---10A----</th>
</tr>
</thead>
<tbody>
<tr>
<td>---1B----</td>
<td>---2----</td>
<td>---3----</td>
<td>---4----</td>
<td>---5----</td>
<td>---6----</td>
<td>---7----</td>
<td>---8----</td>
<td>---9----</td>
<td>---10B----</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------</td>
<td>-----------</td>
<td>-----------</td>
<td>-----------</td>
<td>-----------</td>
<td>-----------</td>
<td>-----------</td>
<td>-----------</td>
<td>-----------</td>
</tr>
</tbody>
</table>

  The following examples show the same card in fixed and large field formats:

<table>
<thead>
<tr>
<th>GRID----</th>
<th>----1----</th>
<th>----2----</th>
<th>----3----</th>
<th>----4----</th>
<th>----5----</th>
<th>----6----</th>
<th>----7----</th>
<th>----8----</th>
<th>----9----</th>
<th>----10----</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>GRID*----</th>
<th>----1----</th>
<th>----2----</th>
<th>----3----</th>
<th>----4----</th>
<th>----5----</th>
<th>----6----</th>
<th>----7----</th>
<th>----8----</th>
<th>----9----</th>
<th>----10----</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>*----------</th>
<th>----1----</th>
<th>----2----</th>
<th>----3----</th>
<th>----4----</th>
<th>----5----</th>
<th>----6----</th>
<th>----7----</th>
<th>----8----</th>
<th>----9----</th>
<th>----10----</th>
</tr>
</thead>
</table>

  **Free Format**
  Fields are separated by commas; blank characters surrounding commas are not significant. Two consecutive commas define empty (blank) fields. If a comma is present in a line of data, it is assumed to be free format data. Continuation lines for free format start with a blank, ‘+’ or ‘*’.

  Large field free format and short field free format are available, but there is no limit on the length of entries, and all floating numbers are read and stored with full precision (64-bit REAL*8) in either case.

  The only difference between large and short free format is that the latter allows for 8 data fields in each line (in positions 2 – 9), while the former allows for 4 data fields per line (similar to the large field fixed format).
format detailed above).

If there is a comma within the first 10 characters in a line, the line is assumed to be in free format; otherwise, if there is an '*' immediately after the card name or a continuation line starts with '*', then the line is accepted as large field fixed format. All other lines are read in as fixed format.

Use of fixed format limits the range of integer data (-9,999,999 .. 99,999,999) and the accuracy of floating point numbers, but does not influence the internal storage of data – in particular all floating point numbers are always read and stored with full precision (64-bit REAL*8).

Bulk data is always limited to 9 fields per line. Content of 10th field and the first field of each continuation line are silently skipped when fixed format is used (other codes can use these fields for special purposes, such as to mark matching continuation lines).

Extensions of free format (which may allow more than 9 fields in a line) are not accepted. An error message will be issued when a free format card contains more than 9 fields. This error can be disabled (changed to non-fatal warning) through the use of SYSSETTING, SKIP10FIELD.

Dollar signs, $, in any column denote comments. All characters after the dollar sign until the end of the line are ignored. Dollar signs can only appear in quoted files names.

Lines which begin with two slashes, //, or a pound symbol, #, are read as comment lines. Blank lines are also assumed to be comment lines.

The full keyword of each bulk data entry must be given starting from the first column. Abbreviated keywords are not allowed.

The format of most bulk data entries is similar to that for Nastran. Not all entry options are supported by RADIOSS and OptiStruct. Consult the list of fields and options supported.

Continuation entries must follow the parent entries. If the first character of any entry is either a blank, a ‘+’, or an “*”, it is treated as a continuation of the previous entry. If the entire line is blank, it is treated as a comment line.

An ENDDATA entry or end-of-file denotes end of data. Lines after the ENDDATA entry are ignored.

All Bulk Data entries must appear after the BEGIN BULK statement in the input data. The content of the tenth field in each card, and that of the first field in each continuation card, is disregarded.

Each entry can be placed anywhere within the field. For example, blanks preceding and following an entry are ignored, except the keyword entry, which must be left justified in its field.

No entry can contain blanks within the data.

Character entries (labels) must start with a letter.

Numeric entries must start with a digit, ‘+’ or ‘-‘.

Integer entries may not contain a decimal point or an exponent part, and must fit in the range of values allowed for INTEGER*4 (usually –2**31<x<2**31).

Integer data placed in the field reserved for real valued data is accepted and converted to a double precision. However, certain fields have alternate functions where the nature of the number entered indicates the desired function; one function requires an integer while the other requires a real number – in this case, no conversion is performed.

Real valued data can be entered without exponent part, with exponent part and explicit letter ‘E’ or ‘D’ or with exponent part starting with a sign (without ‘E’ or ‘D’). All real values are stored internally as double precision data (64-bit REAL*8) without regards to which format was used to enter them.

Following are valid examples of input for real valued data:

1.
0.1
Character entries longer than eight characters are silently truncated in large field and free field formats, with the exception of file names on the INCLUDE entry (see documentation for INCLUDE entry) and the “LABEL” field on DESVAR, DRESP1, DRESP2, DRESP3, and DTABLE entries (allows up to 16 characters).

Continuation lines do not have to be in the same format as the parent entries. It is allowed to mix lines in different formats within a single bulk data card. Invisible tab characters are equivalent to the number of spaces needed to advance to the nearest tab stop. Tab stops are placed at the beginning of each eight-character field. Use the SYSSETTINGS,TABSTOPS option to change this value, e.g. to tab stops at 4-character fields.

Replication of GRID data:
- Replication is a limited data generation capability which may be used for GRID data only.
- Duplication of fields from the preceding GRID entry is accomplished by coding the symbol =.
- Duplication of all trailing fields from the preceding entry is accomplished by coding the symbol ==.
- Incrementing a value from the previous entry is indicated by coding *x or *(x), where x is the value of the increment. “x” should be a real number for real fields or an integer for integer fields. The parentheses will be ignored and removed.
- Only the fields for ID, CP, X, Y, Z, and CD can be incremented. The PS data cannot be incremented.
- Replication data can follow other replication data.

Entered entries:
GRID,101,17,1.0,10.5,,17,3456
GRID,*1,=,* (0.2),==
GRID,*100,,=,=,*10.0,==
GRID,20,17,==

Generated entries:

<table>
<thead>
<tr>
<th>GRID</th>
<th>101</th>
<th>17</th>
<th>1.0</th>
<th>10.5</th>
<th>17</th>
<th>3456</th>
</tr>
</thead>
<tbody>
<tr>
<td>GRID</td>
<td>102</td>
<td>17</td>
<td>1.2</td>
<td>10.5</td>
<td>17</td>
<td>3456</td>
</tr>
<tr>
<td>GRID</td>
<td>202</td>
<td>1.2</td>
<td>10.5</td>
<td>10.0</td>
<td>17</td>
<td>3456</td>
</tr>
<tr>
<td>GRID</td>
<td>202</td>
<td>17</td>
<td>1.2</td>
<td>10.5</td>
<td>10.0</td>
<td>17</td>
</tr>
</tbody>
</table>

Removal of Duplicate Entries:
- Removal of duplicate entries is a limited to GRID, CORD1R, CORD1C, CORD1S, CORD2R, CORD2C, CORD2S, CORD3R, and CORD4R entries only.
- To be considered duplicates, the GRID ID, CP, CD, and PS fields must be the same. The GRID coordinates should be the same within the setting determined by PARAM,DUPTOL.
- For the coordinate information to be considered duplicate, the CID and GID must be the same and the vector components and axis locations must be the same within the setting determined by
PARAM, DUPTOL.

The removal of duplicated GRID data is performed after any GRID data is generated using the GRID replication feature.

- For all other cards which require a unique ID, it is an error if any given ID appears more than once. However, to facilitate application of changes resulting from optimization, it is possible to redefine content of some cards using a separate file defined with ASSIGN, UPDATE.