Multibody Analysis Guide
Copyright and Trademark Information

© 2009 SAS IP, Inc. All rights reserved. Unauthorized use, distribution or duplication is prohibited.

ANSYS, ANSYS Workbench, Ansoft, AUTODYN, EKM, Engineering Knowledge Manager, CFX, FLUENT, HFSS and any and all ANSYS, Inc. brand, product, service and feature names, logos and slogans are registered trademarks or trademarks of ANSYS, Inc. or its subsidiaries in the United States or other countries. ICEM CFD is a trademark used by ANSYS, Inc. under license. CFX is a trademark of Sony Corporation in Japan. All other brand, product, service and feature names or trademarks are the property of their respective owners.

Disclaimer Notice

THIS ANSYS SOFTWARE PRODUCT AND PROGRAM DOCUMENTATION INCLUDE TRADE SECRETS AND ARE CONFIDENTIAL AND PROPRIETARY PRODUCTS OF ANSYS, INC., ITS SUBSIDIARIES, OR LICENSORS. The software products and documentation are furnished by ANSYS, Inc., its subsidiaries, or affiliates under a software license agreement that contains provisions concerning non-disclosure, copying, length and nature of use, compliance with exporting laws, warranties, disclaimers, limitations of liability, and remedies, and other provisions. The software products and documentation may be used, disclosed, transferred, or copied only in accordance with the terms and conditions of that software license agreement.
ANSYS, Inc. is certified to ISO 9001:2008.

U.S. Government Rights

For U.S. Government users, except as specifically granted by the ANSYS, Inc. software license agreement, the use, duplication, or disclosure by the United States Government is subject to restrictions stated in the ANSYS, Inc. software license agreement and FAR 12.212 (for non-DOD licenses).

Third-Party Software

See the legal information in the product help files for the complete Legal Notice for ANSYS proprietary software and third-party software. If you are unable to access the Legal Notice, please contact ANSYS, Inc.

Published in the U.S.A.
# Table of Contents

1. **Introduction to Multibody Simulation** ............................................................................................................. 1
   1.1. Benefits of the Finite Element Method for Modeling Multibody Systems .................................................. 1
   1.2. Overview of the ANSYS Multibody Analysis Process ...................................................................................... 2
   1.3. The ANSYS-ADAMS Interface ...................................................................................................................... 3
   1.4. Learning More About Multibody Dynamics .................................................................................................. 3

2. **Modeling in a Multibody Simulation** ............................................................................................................ 5
   2.1. Modeling Flexible Bodies in a Multibody Analysis .......................................................................................... 5
      2.1.1. Element Choices for Flexible Bodies ...................................................................................................... 6
   2.2. Modeling Rigid Bodies in a Multibody Analysis ............................................................................................ 7
      2.2.1. Defining a Rigid Body ............................................................................................................................ 7
         2.2.1.1. Typical Rigid Body Scenarios .......................................................................................................... 7
         2.2.1.2. Target Element Key Option Setting for Defining a Rigid Body ...................................................... 9
         2.2.1.3. Defining a Rigid Body Pilot Node ................................................................................................... 9
         2.2.1.4. Defining Rigid Body Mass and Rotary Inertia Properties .............................................................. 10
      2.2.2. Rigid Body Degrees of Freedom ........................................................................................................... 11
      2.2.3. Rigid Body Boundary Conditions ........................................................................................................ 12
         2.2.3.1. Defining Rigid Body Loads .............................................................................................................. 13
      2.2.4. Representing Parts of a Complex Model with Rigid Bodies ................................................................... 13
      2.2.5. Connecting Joint Elements to Rigid Bodies .......................................................................................... 13
      2.2.6. Modeling Contact with Rigid Bodies .................................................................................................. 14
   2.3. Connecting Multibody Components with Joint Elements ........................................................................... 14
      2.3.1. Joint Element Types ............................................................................................................................ 15
         2.3.1.1. Joint Element Connectivity Definition .......................................................................................... 17
         2.3.1.2. Joint Element Section Definition .................................................................................................. 18
         2.3.1.3. Local Coordinate System Specification for Joint Elements ....................................................... 18
         2.3.1.4. Stops or Limits with Joint Elements .............................................................................................. 19
         2.3.1.5. Joint Mechanism Locks ............................................................................................................... 21
      2.3.2. Material Behavior of Joint Elements .................................................................................................... 22
         2.3.2.1. Stiffness and Damping Behavior of Joint Elements ....................................................................... 22
         2.3.2.2. Frictional Behavior ....................................................................................................................... 23
            2.3.2.2.1. Geometry specifications for Coulomb friction in Joints .......................................................... 25
            2.3.2.2.2. Calculation of Normal Forces for Coulomb Frictional Behavior .......................................... 25
         2.3.3. Reference Lengths and Angles for Joint Elements ............................................................................. 27
      2.3.4. Boundary Conditions for Joint Elements ............................................................................................. 27
      2.3.5. Connecting Bodies to Joints ................................................................................................................ 28

3. **Performing a Multibody Analysis** .................................................................................................................. 33
   3.1. Kinematic Constraints .................................................................................................................................. 33
   3.2. Convergence Criteria ................................................................................................................................... 33
   3.3. Initial Conditions ........................................................................................................................................... 34
      3.3.1. Apply Linear Acceleration in a Dummy Transient Analysis .................................................................. 34
      3.3.2. Apply Large Numerical Damping Over a Short Interval ................................................................... 36
   3.4. Damping ....................................................................................................................................................... 37
      3.4.1. Numerical Damping ............................................................................................................................ 38
      3.4.2. Structural Damping ........................................................................................................................... 38
   3.5. Time-Step Settings ....................................................................................................................................... 38
   3.6. Solver Options ............................................................................................................................................. 38

4. **Reviewing Multibody Analysis Results** ......................................................................................................... 39
   4.1. Reviewing Results in POST1 ...................................................................................................................... 39
   4.2. Reviewing Results in POST26 .................................................................................................................. 40
   4.3. Output of Joint Element Quantities ........................................................................................................... 41
5. Using Component Mode Synthesis Superelements in a Multibody Analysis
5.1. Applicability of CMS Superelements in a Multibody Analysis ................................................................. 43
5.2. Flexible Body Types ....................................................................................................................................... 43
5.3. Substructuring Overview ............................................................................................................................... 44
5.4. Master Degrees of Freedom in a Substructured Multibody Simulation ...................................................... 44
5.5. Steps for Performing a Substructured Multibody Simulation ........................................................................ 45
  5.5.1. Step 1: Prepare the Full Model for a Substructured Multibody Analysis ...................................................... 46
  5.5.2. Step 2: Create the Substructures (Generation Pass) .................................................................................... 46
  5.5.3. Step 3: Build the CMS-based Model (Use Pass) ......................................................................................... 48
  5.5.4. Step 4: Run the Multibody Analysis .......................................................................................................... 49
  5.5.5. Step 5: Expand all Solutions (Expansion Pass) ......................................................................................... 49
  5.5.6. Step 6: Create the Merged Results File ...................................................................................................... 50
  5.5.7. Step 7: Postprocess the Results .................................................................................................................. 51
6. Example Multibody Analysis: Crank Slot Mechanism ..................................................................................... 53
   6.1. Problem Description ..................................................................................................................................... 53
   6.2. Problem Specifications ................................................................................................................................. 53
   6.3. Defining Joints ............................................................................................................................................. 54
   6.4. Performing the Rigid Body Analysis ........................................................................................................... 55
   6.5. Performing the Flexible Body Analysis ....................................................................................................... 56
   6.6. Using Component Mode Synthesis in the Multibody Analysis .................................................................... 58
   6.7. Using Joint Probes ....................................................................................................................................... 59
   6.8. Comparing Processing Times ..................................................................................................................... 60
   6.9. Input Files Used in This Analysis ................................................................................................................ 60
7. Troubleshooting a Flexible Multibody Analysis ............................................................................................... 61
   7.1. Addressing Overconstraint Issues During Modeling .................................................................................... 61
     7.1.1. Overconstraints in Rigid Bodies ................................................................................................................ 61
     7.1.1.1. Standard Four-Bar Mechanism ........................................................................................................... 62
     7.1.1.2. Redundant Rigid Bodies ....................................................................................................................... 62
     7.1.1.3. Redundant Boundary Conditions ......................................................................................................... 63
     7.1.2. Overconstraints Caused by User-Defined Constraint Equations ............................................................ 64
   7.2. Resolving Overconstraint Problems ............................................................................................................. 64
Index ..................................................................................................................................................................... 67

List of Figures

2.1. FE Slider-Crank Mechanism ............................................................................................................................ 6
2.2. Rigid Body Definition With Underlying Elements .......................................................................................... 8
2.3. Rigid Body Definition Without Underlying Elements .................................................................................... 8
2.4. Rigid Body with a Limited Number of Nodes ............................................................................................... 9
2.5. 2-D Rigid Body DOFs Subject to Applied Boundary Conditions ............................................................... 11
2.6. Rigid Sphere Translational DOFs + Rotational DOFs .................................................................................. 12
2.7. Rigid Body Translational DOFs Only ............................................................................................................ 12
2.8. MPC184 Universal Joint Geometry ............................................................................................................. 19
2.9. Stops Imposed on a Revolute Joint ................................................................................................................ 20
2.10. Stops Imposed on a Slot Joint ....................................................................................................................... 21
2.11. Nonlinear Stiffness and Damping Behavior for Joints .................................................................................. 23
2.12. Coulomb’s Law ........................................................................................................................................... 24
2.13. Exponential Friction Law .............................................................................................................................. 25
2.15. Pinned Joint Mesh and Revolute Joint ........................................................................................................ 29
2.16. Pinned Joint Contact Elements ................................................................. 30
2.17. Pinned Joint Constraint Equations .............................................................. 30
2.18. Rigid Constraint (KEYOPT(4) = 2) ............................................................... 31
2.19. Flexible Constraint (KEYOPT(4) = 1) .......................................................... 31
7.1. Overconstrained System: Standard 3-D Four-Bar Mechanism .............................. 62
7.2. Overconstraint Due to Redundant Rigid Components ........................................ 63
7.3. Overconstrained System: Cylindrical Tube Subjected to Bending at One End ................. 64

List of Tables

2.1. Rigid Body vs. Flexible Body Definition .............................................................. 13
2.2. Required Geometric Quantities ...................................................................... 25
Chapter 1: Introduction to Multibody Simulation

Multibody simulation consists of analyzing the dynamic behavior of a system of interconnected bodies comprised of flexible and/or rigid components. The bodies may be constrained with respect to each other via a kinematically admissible set of constraints modeled as joints. These systems can represent an automobile, a space structure with antenna deployment capabilities, an aircraft as an assemblage of rigid and flexible parts, a robot with manipulator arms, and so on. In all such cases, the components may undergo large rotation, large displacement, and finite strain effects.

This animated model of an aircraft landing gear is a typical example of a multibody simulation:

The following additional topics offer more information to help you understand multibody simulation and how the ANSYS program supports it:

1.2. Overview of the ANSYS Multibody Analysis Process
1.3. The ANSYS-ADAMS Interface
1.4. Learning More About Multibody Dynamics


Multibody systems have conventionally been modeled as rigid body systems with superimposed elastic effects of one or more components. These methods have been well documented in multibody dynamics literature. A major limitation of these methods is that nonlinear large-deformation, finite strain effects, or nonlinear material cannot be incorporated completely into model.

The finite element (FE) method used in ANSYS offers an attractive approach to modeling a multibody system. While the ANSYS multibody analysis method may require more computational resources and modeling time compared to standard analyses, it has the following advantages:

- The finite element mesh automatically represents the geometry while the large deformation/rotation effects are built into the finite element formulation.
- Inertial effects are greatly simplified by the consistent mass formulation or even point mass representations.
- Interconnection of parts via joints is greatly simplified by considering the finite motions at the two nodes forming the joint element.
The parameterization of the finite rotation has been well documented in the literature and can be easily incorporated into the joint element formulations thereby enabling complete simulation of a multibody system.

ANSYS has an extensive library of elements available for modeling the flexible, rigid, and joint components. You can model the material behavior of the flexible components using one of several material models. ANSYS also provides modal and transient dynamics capabilities to analyze the spatial and temporal effects in a multibody simulation. Extensive postprocessing capabilities are also available to interpret the analysis results.

You can perform multibody simulation on a wide variety of mechanical systems. Typical applications include automobiles and automobile components, aircraft assemblages, spacecraft applications, and robotics.

### 1.2. Overview of the ANSYS Multibody Analysis Process

A multibody simulation involves the same general steps necessary for any ANSYS nonlinear analysis. The following table describes the multibody analysis process:

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Build the model.</td>
<td>A flexible mechanism usually consists of flexible and/or rigid parts</td>
</tr>
<tr>
<td></td>
<td></td>
<td>connected together with joint elements. You can model the flexible</td>
</tr>
<tr>
<td></td>
<td></td>
<td>parts with any of the 3-D solid, shell, or beam elements available in</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the ANSYS element library. (For more information, see Building the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Model in the Basic Analysis Guide, and Chapter 2, Modeling in a Multibody</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Simulation (p. 5) in this document.)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rigid bodies are modeled using MPC184 Rigid Link or Rigid Beam elements,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>or by using the extensive contact capabilities available in ANSYS.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The flexible and/or rigid parts are connected using MPC184 joint elements.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>For example, two parts may be simply connected such that the displacements</td>
</tr>
<tr>
<td></td>
<td></td>
<td>at the joining position are identical. In other cases, the connection</td>
</tr>
<tr>
<td></td>
<td></td>
<td>between two parts may involve a more sophisticated joint such as the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>planar joint or universal joint. In modeling these joints, suitable</td>
</tr>
<tr>
<td></td>
<td></td>
<td>kinematic constraints are imposed on the relative motion (displacement</td>
</tr>
<tr>
<td></td>
<td></td>
<td>and rotation) between the two nodes forming the joint.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>An overview of the types of joint elements used in a multibody analysis</td>
</tr>
<tr>
<td></td>
<td></td>
<td>is available in Connecting Multibody Components with Joint Elements (p.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>14).</td>
</tr>
<tr>
<td>2.</td>
<td>Define element types.</td>
<td>To properly perform a flexible multibody simulation, which involves</td>
</tr>
<tr>
<td></td>
<td></td>
<td>flexible and rigid components joined together with some form of kinematic</td>
</tr>
<tr>
<td></td>
<td></td>
<td>constraints, use appropriate structural, joint, and contact element</td>
</tr>
<tr>
<td></td>
<td></td>
<td>types. For more information about element selection, see Chapter 2,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Modeling in a Multibody Simulation (p. 5).</td>
</tr>
<tr>
<td>3.</td>
<td>Define materials.</td>
<td>Defining the material properties for multibody components is no different</td>
</tr>
<tr>
<td></td>
<td></td>
<td>than defining them in any other ANSYS analysis. Define linear and</td>
</tr>
<tr>
<td></td>
<td></td>
<td>nonlinear material properties via the MP or the TB command. For more</td>
</tr>
<tr>
<td></td>
<td></td>
<td>information, see Defining Material Properties in the Basic Analysis Guide.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The MPC184 joint elements also allow you to define material properties</td>
</tr>
<tr>
<td></td>
<td></td>
<td>so that you can control their behavior along the “free” or “uncon-</td>
</tr>
</tbody>
</table>
1.3. The ANSYS-ADAMS Interface

The ADAMS software marketed by MSC Software is one of several special-purpose, third-party programs used to simulate the dynamics of multibody systems.

A drawback of the ADAMS program is that all components are assumed to be rigid. In the ADAMS program, tools to model component flexibility exist only for geometrically simple structures. To account for the flexibility of a geometrically complex component, ADAMS relies on data transferred from finite-element programs such as ANSYS. The ANSYS-ADAMS Interface is a tool provided by ANSYS, Inc. to transfer data from the ANSYS program to the ADAMS program.

For more information, see "Rigid Body Dynamics and the ANSYS-ADAMS Interface" in the Advanced Analysis Techniques Guide.

Current versions of ANSYS support multibody analysis without the need for third-party tools. In addition, ANSYS allows both rigid and flexible components.

1.4. Learning More About Multibody Dynamics

A considerable body of literature exists concerning multibody dynamics simulation. The following list of resources offers a wealth of information but is by no means exhaustive:


<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.</td>
<td>Mesh the model.</td>
<td>Use the ANSYS meshing commands to mesh multibody components. For more information, see <em>Modeling and Meshing Guide</em>. Issue the LMESH command to mesh rigid bodies defined by MPC184 Rigid Beam or Rigid Link elements. Use the Contact Wizard to mesh rigid bodies defined via the contact capabilities in ANSYS. For more information, see Chapter 2, <em>Modeling in a Multibody Simulation</em> (p. 5). Two nodes define joint elements and no special meshing commands are necessary to define them.</td>
</tr>
<tr>
<td>5.</td>
<td>Solve the model.</td>
<td>The solution phase of a multibody analysis adheres to standard ANSYS conventions. For multibody-specific solver information, see <em>Solver Options</em> (p. 38).</td>
</tr>
<tr>
<td>6.</td>
<td>Review the results.</td>
<td>You can use POST1 (the general postprocessor) and POST26 (the time-history postprocessor) to review results. For more information, see Chapter 4, <em>Reviewing Multibody Analysis Results</em>.</td>
</tr>
</tbody>
</table>
Chapter 2: Modeling in a Multibody Simulation

A variety of issues can arise when modeling a multibody mechanism. The finite element modeling of a multibody mechanism depends on the degree of complexity that you require. For example, it is often possible to create a quick, initial approximation of the flexible and rigid parts of a mechanism using standard beam elements and rigid beam/link elements. Alternatively, you can perform detailed modeling of the flexible part using 3-D solid elements (or shell or solid-shell elements), and the rigid part using the ANSYS program’s extensive contact capabilities.

The following topics related to multibody analysis modeling are available:

2.1. Modeling Flexible Bodies in a Multibody Analysis
2.2. Modeling Rigid Bodies in a Multibody Analysis
2.3. Connecting Multibody Components with Joint Elements

2.1. Modeling Flexible Bodies in a Multibody Analysis

Consider a slider-crank mechanism as shown in the following figure. The crank is considered to be rigid and the connecting link is assumed to be flexible. The link connects the crank to the sliding block (or piston). The simplified finite element model of the slider-crank mechanism is also shown.
The slider-crank mechanism has these characteristics:

- The rigid crank is modeled with an MPC184 Rigid Beam element.
- The rigid crank is connected to ground with a “grounded” MPC184 Revolute Joint element.
- The connecting link is flexible and modeled with BEAM188 elements.
- The rigid crank and the connecting link are connected to each other by a MPC184 Revolute Joint element.
- The connecting link moves within a “grounded” MPC184 Slot Joint that approximates a slider block.

As a quick first attempt, you can model the flexible mechanism with some simple approximations to the flexible and rigid parts. You can also model the connecting link in detail to study the deformation, stresses, etc.

ANSYS offers an extensive library of beam, shell, solid-shell, and solid elements for modeling the flexible parts, and the extensive contact capability to model the rigid part and any other contact conditions. Joint elements implemented via the Lagrange multiplier method offer the required kinematic connectivity between any two parts or components.

### 2.1.1. Element Choices for Flexible Bodies

ANSYS offers a rich suite of beam, shell, and solid elements to model the flexible structural components. Each element has a prefix that identifies the element category and a unique number (for example, BEAM188 and SHELL181).
To model mass and rotary inertia, use the MASS21 element. The element is also appropriate for use in a lumped approximation of rigid bodies.

Detailed information about element selection for flexible components is available in the Basic Analysis Guide and the Element Reference.

2.2. Modeling Rigid Bodies in a Multibody Analysis

Rigid bodies are widely used for numerical simulation of multibody dynamic applications. A rigid body can be connected to other rigid bodies via joint elements. It can also be connected to flexible bodies to model mixed rigid-flexible body dynamics.

In a finite-element model, certain relatively stiff parts can be represented by rigid bodies when stress distributions and wave propagation in such parts are not critical. An advantage of using rigid bodies rather than deformable finite elements is computational efficiency. Elements that belong to the rigid bodies have no associated internal forces or stiffness. The motion of the rigid body is determined by a maximum of six degrees of freedom (DOFs) at the pilot node.

For transient dynamic analyses, stiff bodies can excite high-frequency modes, resulting in a small time increment in order to obtain a stable solution. Rigid bodies do not, however, excite any frequency modes; therefore, using rigid bodies to represent stiff regions may allow a relatively large time increment.

The following topics about rigid body modeling are available:

2.2.1. Defining a Rigid Body
2.2.2. Rigid Body Degrees of Freedom
2.2.3. Rigid Body Boundary Conditions
2.2.4. Representing Parts of a Complex Model with Rigid Bodies
2.2.5. Connecting Joint Elements to Rigid Bodies
2.2.6. Modeling Contact with Rigid Bodies

2.2.1. Defining a Rigid Body

A rigid body in ANSYS consists of a set of target nodes called rigid body nodes and a single pilot node. The associated target elements use the same real constant ID. The motion of the rigid body is governed by the degrees of freedom (DOFs) at the pilot node, allowing accurate representation of the geometry, mass, and rotary inertia of the rigid body.

2.2.1.1. Typical Rigid Body Scenarios

In most applications, rigid bodies start with discretized finite elements. The rigid body can be defined on the exterior of a pre-meshed body discretized by solid, shell, and beam elements (called underlying elements), as shown:
Figure 2.2: Rigid Body Definition With Underlying Elements

The 3-D target element (TARGE170) and 2-D target element (TARGE169) are applied on the exterior surface of the rigid body. To generate the target elements, issue an **ESURF** command.

The rigid body can also be a simple standalone body when the target elements do not overlap other elements (that is, have no underlying elements), as shown:

Figure 2.3: Rigid Body Definition Without Underlying Elements

You can generate target elements TARGE170 for a standalone 3-D rigid body (**AMESH**) or target elements TARGE169 for a standalone 2-D rigid body (**LMESH**).

The most efficient rigid body should contain a limited number of nodes which are either connected to other elements or subject to boundary conditions, as shown:
Figure 2.4: Rigid Body with a Limited Number of Nodes

The rigid body shown above contains three nodes which connect five elements (two 3-D line segments, one pilot node segment, one **MASS21**, and one **MPC184-Revolute**).

Target element POINT segments (**TSHAP,POINT**) can be defined and used to apply boundary conditions (point loads, displacement constraints, etc.) on the rigid body surface where no predefined nodes exist.

### 2.2.1.2. Target Element Key Option Setting for Defining a Rigid Body

Each rigid body contains target elements defined by the same real constant ID. The target elements can be defined via different element type IDs, however, you must set KEYYOPT(2) = 1 on all of the target elements. This KEYYOPT setting causes ANSYS to build internal multipoint constraints (MPC) to enforce kinematics of the entire rigid body.

You can also combine different target segment types for each rigid body. However, you cannot mix 2-D with 3-D target elements.

### 2.2.1.3. Defining a Rigid Body Pilot Node

In addition to the rigid body nodes, each rigid body also must be associated with a rigid body pilot node. The target element defining the pilot node must use the same real constant ID as the other target elements which constitute the rigid body. The real constant ID identifies each rigid body, and ANSYS builds internal multipoint constraints (surface-based rigid constraints) during solution.
The pilot node, unlike the other segment types, is used to define the degrees of freedom for the entire rigid body. This node can be any of the target element nodes, but it does not have to be. All possible rigid motions of the rigid body will be a combination of a translation and a rotation around the pilot node. The pilot node provides a convenient and powerful way to assign boundary conditions such as rotations, translations, moments, temperature, voltage, and magnetic potential on the entire rigid body. The pilot node can be connected to point mass, follower, and deformable elements. For a transient analysis, you can simply locate the pilot node at the gravity center of the rigid body if the center of mass is known.

### 2.2.1.4. Defining Rigid Body Mass and Rotary Inertia Properties

For multibody dynamics, the mass and rotary inertia of the rigid body play important roles in the dynamic response. In ANSYS, the target elements which define rigid bodies do not contribute mass to the finite element system. The most effective way to contribute mass is to add the point mass element MASS21 on the gravity center of the rigid body when the center of mass and rotary inertia properties of the actual rigid body can be estimated. You can specify the rigid body mass and rotary inertia for MASS21. The node of the MASS21 element is usually connected to the pilot node, although it can be connected to any one of the rigid body nodes. The point mass node is often defined in a local coordinate system which is parallel to the rotary principal axes.

Sometimes, the location of gravity center, the mass, and rotary inertia cannot be easily estimated. In such cases, you can use the premeshed body to account for mass distribution for the rigid body (as shown in Figure 2.2: Rigid Body Definition With Underlying Elements (p. 8)). The discretized elements can be pure elastic solid, shell, or beam elements.

For each rigid body, you can perform the following steps:

1. Select the associated elements (ESEL)
2. Specify the option for precalculating masses (IRLF, -1).
3. Perform a partial element solution (PSOLVE, ELFORM).
4. Calculate inertia relief terms and print a summary of the mass properties (PSOLVE, ELPREP)
5. Get the mass properties (*GET), as follows:

<table>
<thead>
<tr>
<th>Item1</th>
<th>IT1NUM</th>
<th>Description</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTOT</td>
<td>X, Y, Z</td>
<td>Total mass components.</td>
<td>(M_x, M_y, M_z)</td>
</tr>
<tr>
<td>MC</td>
<td>X, Y, Z</td>
<td>Mass centroid components.</td>
<td>(X_c, Y_c, Z_c)</td>
</tr>
<tr>
<td>IPRIN</td>
<td>X, Y, Z</td>
<td>Principal centroidal moments of inertia.</td>
<td>(I_{xx}, I_{yy}, I_{zz})</td>
</tr>
<tr>
<td>IANG</td>
<td>XY, YZ, ZX</td>
<td>Angles of the principal axes.</td>
<td>(\theta_{xy}, \theta_{yz}, \theta_{xz})</td>
</tr>
</tbody>
</table>

Based on the precalculated mass properties, you can easily define the point mass element. The node is defined in the local coordinate system, as shown:

\[X_c, Y_c, Z_c, \theta_{xy}, \theta_{yz}, \theta_{xz}\]

The mass properties are specified by real constants:

\[M_x, M_y, M_z, I_{xx}, I_{yy}, I_{zz}\]

Set MASS21 KEYOPT(2) = 1 so that the point mass element coordinate system is initially parallel to the nodal coordinate system and rotates with the nodal coordinate rotations during a large-deflection analysis.
2.2.2. Rigid Body Degrees of Freedom

The pilot node has both translational and rotational degrees of freedom (DOFs). The active DOFs at the pilot node depend on the defined type of target elements. Use TARGE169 for a 2-D rigid body which contains UX, UY and ROTZ DOFs. Use TARGE170 for 3-D rigid body which contains UX, UY, UZ and ROTX, ROTY, ROTZ DOFs.

The DOFs of rigid body nodes are based on the DOFs of the connected elements and applied boundary conditions (BCs). Rigid body nodes that connect to solid elements involve only the translational degrees of freedom. Rigid body nodes that connect to shell, beam, follower, and joint elements also involve the rotational DOFs.

For standalone rigid body nodes not connected to any other elements, the associated DOFs are subject to applied boundary conditions, as shown:

**Figure 2.5: 2-D Rigid Body DOFs Subject to Applied Boundary Conditions**

The node has DOF UX if a constraint or a force is applied in the X direction. If there are no applied BCs, the standalone rigid body nodes have no DOFs; in such a case, ANSYS simply updates the position of the nodes based on the kinematics of the rigid body.

The DOFs for a rigid body can also be controlled via KEYOPT(4) of the target element (TARGE169 or TARGE170). The key option offers additional flexibility by fully or partially constraining the DOFs for the rigid body.

**Examples**

In the following figure, a rigid sphere is defined by 8-node quadrilateral segments and a pilot node. Two beam elements are connected to the rigid surface in the XY plane, as shown by the dotted lines. The pilot node is located at the global Cartesian origin and is subjected to rotation ROTZ.

For the DOFs of the rigid body, selecting three rotational DOFs along with three translational DOFs rotates the beams, as shown. Because the beams are fully connected to the rigid sphere, they rotate with the sphere.
Selecting only the three translational DOFs for the rigid body, as shown in the following figure, does not rotate the beams because they are connected only in their translational DOFs; therefore, the connection acts as a hinge.

Determining the DOFs for each rigid body node is important because the internal multipoint constraints are built solely on the resulting DOFs.

**2.2.3. Rigid Body Boundary Conditions**

Constrained boundary conditions (BCs) for the rigid body are usually applied on the rigid body pilot node. Reaction forces can be obtained for DOFs at the constrained nodes. A combination of rigid body constraints and constrained boundary conditions applied to several rigid body nodes other than the pilot node can lead to overconstrained models. In such cases, ANSYS issues overconstraint warnings and attempts to remove the redundant constraints if possible. If the specified BCs are not consistent with the rigid body constraint, the model becomes inconsistently overconstrained. You must verify the overconstrained model and prevent conflicting overconstraints.
2.2.3.1. Defining Rigid Body Loads

You can apply point loads on any rigid body nodes and pilot node. Follower force (FOLW201) can be defined at those nodes, and the direction of forces is determined by the rotation of the nodes.

You can apply surface loads on surface effect elements SURF153 and SURF154 which fully or partially override loads on the surface of the rigid body.

Loads on a rigid body are assembled from contributions of all loads on nodes and elements connected to the rigid body.

2.2.4. Representing Parts of a Complex Model with Rigid Bodies

Using rigid bodies to represent certain portions of a complex model is more efficient than using flexible finite elements. In the early stage of finite element model development, you can treat certain stiff parts or discretized elements that are far away from the region of interest as the rigid bodies. In a later stage, you can remove the rigid body definition and add the flexible discretized elements back for a detailed and accurate finite element analysis.

By selecting or deselecting target elements or the flexible finite elements, you can easily switch back and forth between rigid body and flexible body definition.

The following table shows the general steps involved when defining a rigid body as compared to defining a flexible body:

Table 2.1 Rigid Body vs. Flexible Body Definition

<table>
<thead>
<tr>
<th>Rigid Body Definition Process</th>
<th>Flexible Body Definition Process</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the associated finite elements with defined mass density.</td>
<td>1. Unselect the relevant point mass and target elements.</td>
</tr>
<tr>
<td>2. Perform a partial element solution to obtain mass properties.</td>
<td>2. Reselect the associated finite elements.</td>
</tr>
<tr>
<td>3. Add a point mass element to the center of rigid body.</td>
<td>3. Define the material properties for the flexible body.</td>
</tr>
<tr>
<td>4. Add a target element whose node (pilot node) shares the point mass node.</td>
<td>4. Define a pilot node at one end of the joint. The pilot node connects the joint to the rest of the body.</td>
</tr>
<tr>
<td>5. Generate target elements on the exterior surface of the pre-mesh body.</td>
<td>5. Select the nodes on the exterior surface of the body that you want to connect to this pilot node.</td>
</tr>
<tr>
<td>6. Unselect the associated finite elements.</td>
<td>6. Create target elements on this surface.</td>
</tr>
<tr>
<td>7. Connect joint elements to target nodes.</td>
<td>For each body-joint connection, repeat steps 4 through 6. For more information, see Connecting Bodies to Joints (p. 28).</td>
</tr>
</tbody>
</table>

2.2.5. Connecting Joint Elements to Rigid Bodies

Joint elements can be connected to any rigid body nodes and the pilot node. You can define connections between rigid bodies, or between a rigid body and a flexible body.
Caution

Redundant constraints are most likely to occur when two rigid bodies are connected to more than one joint element.

2.2.6. Modeling Contact with Rigid Bodies

Contact between two rigid bodies is modeled by specifying a contact surface on one rigid body and a target surface on another rigid body. Use either the augmented Lagrange algorithm or penalty algorithm (KEYOPT(2) on the contact element) for modeling contact between rigid bodies to avoid redundant overconstraint between rigid body constraints and contact constraints.

You cannot use the multipoint constraint (MPC) algorithm (KEYOPT(2)) and bonded or no-separation contact behavior (KEYOPT(12)) to connect two rigid surfaces; doing so would cause the model to be overconstrained, resulting in an abnormal termination of the analysis. You can simply replace the bonded contact pair by adding an additional rigid body which connects two pilot nodes.

ANSYS allows two rigid bodies that are connected or overlap each other through rigid body nodes or the pilot node. To prevent overconstraints, ANSYS merges two rigid bodies into one rigid body internally and treats the second pilot node as a regular rigid body node.

MPC bonded contact between a flexible body and a rigid body is possible. The contact surface in an MPC bonded contact pair, however, should always belong to the flexible body; otherwise, the MPC bonded constraints and rigid body constraints are redundant.

2.3. Connecting Multibody Components with Joint Elements

The **MPC184** family of elements serves to connect the flexible and/or rigid components to each other in a multibody mechanism.

An **MPC184** joint element is defined by two nodes with six degrees of freedom at each node (for a total of 12 DOFs). The relative motion between the two nodes is characterized by six relative degrees of freedom. Depending on the application, you can configure different kinds of joint elements by imposing appropriate kinematic constraints on any or some of these six relative degrees of freedom. For example, to simulate a **revolute joint**, the three relative displacement degrees of freedom and two relative rotational degrees of freedom are constrained, leaving only one relative degree of freedom available (the rotation around the revolute axis). Similarly, constraining the three relative displacement degrees of freedom and one relative rotational degree of freedom can simulate a universal joint. Two rotational degrees of freedom are “unconstrained” in this joint.

The kinematic constraints in the joint elements are imposed using the Lagrange multiplier method. Because the Lagrange multiplier method is used to impose the constraints, the constraint forces are available for output purposes.

The following topics about using joint elements in a multibody analysis are available:

- **2.3.1. Joint Element Types**
- **2.3.2. Material Behavior of Joint Elements**
- **2.3.3. Reference Lengths and Angles for Joint Elements**
- **2.3.4. Boundary Conditions for Joint Elements**
- **2.3.5. Connecting Bodies to Joints**
2.3.1. Joint Element Types

All joint elements are classified as MPC184 elements. The various elements are available via the MPC184 element’s KEYOPT(1) setting and, in some cases, the KEYOPT(4) setting.

The following table lists the different types of joint elements and the required key option settings. The relevant element section in the *Element Reference* is also indicated.

<table>
<thead>
<tr>
<th>Joint Element Type</th>
<th>KEYOPT(1)</th>
<th>KEYOPT(4)</th>
<th>MPC184 Element</th>
<th>Constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>Revolute</td>
<td>6</td>
<td>---</td>
<td>Revolute Joint</td>
<td>5</td>
</tr>
<tr>
<td>Z-axis revolute</td>
<td>6</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Universal</td>
<td>7</td>
<td>---</td>
<td>Universal Joint</td>
<td>4</td>
</tr>
<tr>
<td>Slot</td>
<td>8</td>
<td>---</td>
<td>Slot Joint</td>
<td>2</td>
</tr>
<tr>
<td>Point-in-plane</td>
<td>9</td>
<td>---</td>
<td>Point-in-Plane Joint</td>
<td>1</td>
</tr>
<tr>
<td>Translational</td>
<td>10</td>
<td>---</td>
<td>Translational Joint</td>
<td>5</td>
</tr>
<tr>
<td>Cylindrical</td>
<td>11</td>
<td>---</td>
<td>Cylindrical Joint</td>
<td>4</td>
</tr>
<tr>
<td>Z-axis cylindrical</td>
<td>11</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spherical</td>
<td>5</td>
<td>---</td>
<td>Spherical Joint</td>
<td>3</td>
</tr>
<tr>
<td>Planar</td>
<td>12</td>
<td>---</td>
<td>Planar Joint</td>
<td>3</td>
</tr>
<tr>
<td>Z-axis planar</td>
<td>12</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Weld</td>
<td>13</td>
<td>---</td>
<td>Weld Joint</td>
<td>6</td>
</tr>
<tr>
<td>Orient</td>
<td>14</td>
<td>---</td>
<td>Orient Joint</td>
<td>3</td>
</tr>
<tr>
<td>General</td>
<td>16</td>
<td>---</td>
<td>General Joint</td>
<td>Depends on number of fixed relative DOFs</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Minimum constraints = 0 (No DOF is fixed)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Maximum constraints = 6 (All DOFs are fixed)</td>
</tr>
<tr>
<td>Screw</td>
<td>17</td>
<td>---</td>
<td>Screw Joint</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Relative axial motion and rotational motion are linked via the pitch of the screw</td>
</tr>
</tbody>
</table>

Following are some examples of joint element types:

- **Revolute Joint**: Allows rotation about a single axis.
- **Universal Joint**: Allows rotation about two perpendicular axes.
- **Slot Joint**: Allows translation along two perpendicular axes.
- **Point-in-Plane Joint**: Allows translation along one axis and rotation about two perpendicular axes.
- **Translational Joint**: Allows translation along three axes.
- **Cylindrical Joint**: Allows translation along the axis and rotation about two perpendicular axes.
- **Spherical Joint**: Allows translation along three axes and rotation about three axes.
- **Planar Joint**: Allows translation along two axes and rotation about one axis.
- **Weld Joint**: Used to simulate a weld joint.
- **Orient Joint**: Used for orientation.
- **General Joint**: Allows any combination of translation and rotation.
- **Screw Joint**: Allows translation along the axis and rotation about the axis.

*Release 12.1 © 2009 SASIP Inc. All rights reserved. Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.*
**Revolute Joint**  Constrained degrees of freedom: UX, UY, UZ, ROTX, ROTY

**Universal Joint**  Constrained degrees of freedom: UX, UY, UZ, ROTY

**Slot Joint**  Constrained degrees of freedom: UY, UZ

**Translational Joint**  Constrained degrees of freedom: UY, UZ, ROTX, ROTY, ROTZ

**Cylindrical Joint**  Constrained degrees of freedom: UX, UY, ROTX, ROTY
2.3.1.1. Joint Element Connectivity Definition

A joint element is typically defined by specifying two nodes, I and J. These nodes may be arbitrarily located in space. There are instances, however, when one of the nodes needs to be considered as a "grounded" node. In such cases, specify either node I or node J appropriately. In cases when the node is grounded, the location of the grounded node is taken to be that of the other specified node.

Example

If the first node of the joint element is a grounded node, then the element definition is: \( E_{,J} \) or \( EN_{,Element-Number_{,J}} \)

Similarly, if the second node is the grounded node, then the element definition is: \( E_{,I} \), or \( EN_{,ElementNumber_{,I}} \)
2.3.1.2. Joint Element Section Definition

Each joint element must have an associated section definition. Use the SECTYPE command to define the section type and subtype.

Example

The universal joint section definition is: SECTYPE,JOINT,UNIV,UNIV-01

2.3.1.3. Local Coordinate System Specification for Joint Elements

Local coordinate systems at the nodes are required to define the kinematic constraints of a joint element. Use the SECJOINT command to do so.

The local coordinate systems and their required orientation vary from one joint element to another. Input data requirements for each joint element differ. Typically, the local coordinate system is always defined at the first node of a joint element.

The local coordinate system at the second node may be optional. If it is not specified, then the local coordinate system at the first node is usually assumed.

The rotational components of the relative motion between the two nodes of the joint elements are quantified in terms of Bryant (or Cardan) angles that are evaluated based on these coordinate systems.

Example

The following figure illustrates the specification of the local coordinate system for a universal joint element:
The local coordinate system specification is:

\[
\text{LOCAL,11,0} \\
\ldots \\
\text{LOCAL,12,0} \\
\ldots \\
\text{SECJOINT,LSYS,11,12}
\]

**2.3.1.4. Stops or Limits with Joint Elements**

Stops or limit constraints in joints are imposed on the available components of relative motion between the two nodes of a joint element. The Lagrange multiplier method is used to implement these constraints. For
static analysis, the stop constraints are based on the relative displacements (or relative rotations) of the free degrees of freedom.

**Stops in Transient Dynamic Analysis**

In a transient dynamic analysis, if relative displacement-based (or rotation-based) stop constraints are used, then the relative velocities and relative accelerations become inconsistent (oscillatory velocity and/or accelerations are observed in many cases), implying that the energy and momentum due to the impact-like nature of the stops is not conserved. These inconsistencies are reasonably suppressed by imposing a numerical damping. However, numerical damping does not work appropriately in some cases. Thus, for the transient dynamic case, an energy-momentum conservation scheme is adopted. By this method, the user specified relative DOF stop values are taken into account, and constraints based on the relative velocity are imposed in such a way that the overall energy and momentum balance is achieved in a finite element sense.

Irrespective of the integration scheme specified for the transient dynamic analysis, the Newmark method is used for the joint element when stops are specified.

The energy-momentum conservation scheme for stops is implemented for all joints except the screw joint. In the case of the screw joint, the stops are imposed based on the relative displacements (or rotations).

**Defining Stops for Joint Elements**

You can impose stops or limits on the available components of relative motion between the two nodes of a joint element. The stops or limits essentially constrain the values of the free DOFs within a certain range. To specify minimum and maximum values, issue the `SECSTOP` command.

The following figure shows how stops can be imposed on a revolute joint such that the motion is constrained. The axis of the revolute is assumed to be perpendicular to the plane of paper and is along the $e_3$ direction.

**Figure 2.9: Stops Imposed on a Revolute Joint**

The local coordinate system specified at node I is assumed to be fixed in its initial configuration. However, the local coordinate system specified at node J evolves with the rotation of that node. The relative angle of rotation is given by:

$$\psi = - \tan^{-1} \left( \frac{e_{1I} \cdot e_{2J}}{e_{1I} \cdot e_{1J}} \right)$$

Let the link with node J rotate with respect to the link with node I. This characteristic implies that the local coordinate system at node J rotates with respect to the local coordinate system at node I.
For the configuration shown, the initial relative angle of rotation is zero degrees. A counterclockwise motion results in positive angles of rotation. Clockwise motion results in negative angles of rotation.

If stops limit the movement of the link with node J (as shown), the stop conditions are specified as follows:

\[
\text{SECSTOP}, 6, \phi_{\text{min}}, \phi_{\text{max}}
\]

The next figure shows how stops can be imposed in a slot joint which involves displacements in the local \( e_1 \) axis of node I. The relative distance between node J and node I is given by:

\[ \ell = e_1 \cdot (x^J - x^I) \]

where \( x^I \) and \( x^J \) are the position vectors of nodes I and J. The initial distance between the nodes I and J is \( \ell_0 \) and is a positive value.

**Figure 2.10: Stops Imposed on a Slot Joint**

The stops are defined as:

\[
\text{SECSTOP}, 1, \ell_{\text{min}}, \ell_{\text{max}}
\]

where \( \ell_{\text{min}} \) and \( \ell_{\text{max}} \) are both positive.

The stops are defined as:

\[
\text{SECSTOP}, 1, -\ell_{\text{min}}, \ell_{\text{max}}
\]

where \( \ell_{\text{min}} \) is negative and \( \ell_{\text{max}} \) is positive.

### 2.3.1.5. Joint Mechanism Locks

Locks or locking limits may also be imposed on the available components of relative motion between the two nodes of a joint element. Locks are basically used in joint mechanisms to “freeze” the joint in a desired configuration during the course of deformation. When the locks are activated on a particular component of relative motion, that component remains locked for the rest of the analysis. Issue the \text{SECLOCK} command to define lock limits.
Referring to Figure 2.9: Stops Imposed on a Revolute Joint (p. 20), the locks for a revolute joint are specified as SECLOCK,6 , Phi_Min,Phi_Max

Referring to Figure 2.10: Stops Imposed on a Slot Joint (p. 21), the locks for the slot joint are specified as SECLOCK,1,l_Min,l_Max

2.3.2. Material Behavior of Joint Elements

The following topics related to the material behavior of joint elements in a multibody analysis are available:

2.3.2.1. Stiffness and Damping Behavior of Joint Elements
2.3.2.2. Frictional Behavior

For more information, see MPC184 Joint Materials in the Element Reference.

2.3.2.1. Stiffness and Damping Behavior of Joint Elements

Linear or nonlinear stiffness and damping behavior can be associated with the free or unrestrained components of relative motion of the joint elements. In the case of linear stiffness or linear damping, the values are specified as coefficients of a 6 x 6 elasticity matrix using the TB,JOIN command with TBOPT = STIF or TBOPT = DAMP. The stiffness and damping values can be temperature-dependent. Depending on the joint element in use, only the appropriate coefficients of the stiffness or damping matrix are used in the joint element constitutive calculations.

The nonlinear stiffness and damping behavior is specified using the TB,JOIN command with an appropriate TBOPT label. In the case of nonlinear stiffness, relative displacement (rotation) versus force (moment) values are specified using the TBPT command. For nonlinear damping behavior, velocity versus force behavior is specified using the TBPT command. (See Figure 2.11: Nonlinear Stiffness and Damping Behavior for Joints (p. 23) for a representation of the nonlinear stiffness or damping curve.) In either case, these values may be temperature dependent; use the TBTEMP command to define the temperature for the data table.
You can specify the linear or nonlinear stiffness or damping behavior independently for each component of relative motion. However, if you specify linear stiffness for an unrestrained component of relative motion, you cannot specify nonlinear stiffness behavior on the same component of relative motion. The damping behavior is similarly restricted. If a joint element has more than one free or unrestrained component of relative motion—for example, the universal joint has two free components of relative motion—then you can independently specify the stiffness or damping behavior as linear or nonlinear for each of the unrestricted components of relative motion.

**2.3.2.2. Frictional Behavior**

Frictional behavior along the unrestrained components of relative motion influences the overall behavior of the joints. You can model Coulomb friction for joint elements via the \texttt{TBJOIN} command with an appropriate \texttt{TBOPT} label. Frictional behavior can be specified only for the following joints:

- Revolute Joint (x-axis and z-axis)
- Slot Joint
- Translational Joint

The laws governing the frictional behavior of the joint are described below.

**Coulomb’s Law**

The classical Coulomb friction model is implemented for joints using a penalty formulation. The Coulomb friction model for joints is defined as:

\[
F_{\text{lim}} = \mu F_n
\]
\[ |F_s| \leq F_{\text{lim}} \]

Where, \( F_s \) is the equivalent tangential force (or moment), \( F_n \) is the normal force (or moment) in the joint, and \( \mu \) is the current value of the coefficient of friction. The calculation of the normal force depends on the joint under consideration.

If the equivalent tangential force \( F_s \) is less than \( F_{\text{lim}} \), the state is known as the sticking state. If \( F_s \) exceeds \( F_{\text{lim}} \), sliding occurs and the state is known as the sliding state. The sticking/sliding calculations determine when a point transitions from sticking to sliding or vice versa.

**Figure 2.12: Coulomb's Law**

Exponential Friction Law

The exponential friction law is used to smooth the transition between the static coefficient of friction and the dynamic coefficient of friction according to the formula (Benson and Hallquist):

\[
\mu = \mu_d + (\mu_s - \mu_d) e^{-c |V_{\text{rel}}|}
\]

where:

- \( V_{\text{rel}} \) = the relative slip rate
- \( \mu_s \) = the coefficient of friction in the static regime (stiction)
- \( \mu_d \) = the coefficient of friction in the dynamic regime
- \( c \) = decay coefficient
2.3.2.2.1. Geometry specifications for Coulomb friction in Joints

The modeling of Coulomb friction in joints requires some geometry specifications, depending on the type of joint under consideration. These quantities are used in the computation of the normal force (or moment) for Coulomb friction calculations. The **SECJOINT** command is used to specify these quantities. The following table outlines the required geometric quantities:

<table>
<thead>
<tr>
<th>Joint Type</th>
<th>Geometric Quantities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Revolute Joint (x-axis and z-axis)</td>
<td>Outer radius, Inner radius, Effective Length</td>
</tr>
<tr>
<td>Slot Joint</td>
<td>None required</td>
</tr>
<tr>
<td>Translational Joint</td>
<td>Effective Length, Effective Radius</td>
</tr>
</tbody>
</table>

If appropriate geometric quantities are not specified, then the corresponding normal force contributions will not be considered. The following section explains the normal force calculations and the geometric quantities required.

2.3.2.2.2. Calculation of Normal Forces for Coulomb Frictional Behavior

The normal force (or moment) that is used in the Coulomb frictional behavior is based on the following forces that arise in a joint:

- Lagrange Multiplier forces (or moments) due to the constraints
- Interference fit forces (or moments)
Revolute Joint

In order to compute the normal moment in a revolute joint, the revolute joint is visualized as a cylinder-pin assembly (for example, a door hinge consisting of a pin with a head inserted into a cylinder).

The following geometric quantities are required in the calculations below. Note that the specification of these quantities is optional. If some of these geometric quantities are not specified, then the corresponding contribution to the normal moment calculations is ignored.

- $R_{outer}$ = Outer radius of the cylinder
- $R_{inner}$ = Inner radius of the cylinder or outside radius of pin
- $L_{eff}$ = The effective length is the length over which the cylinder and pin are in contact with each other

The contributions to the normal moment in an x-axis revolute joint are as follows:

- An axial moment due to the axial component of the constraint Lagrange Multiplier force ($\lambda_1$).
  This force acts in such a way as to push the cylinder against the pin head, thereby causing a frictional moment to develop.
  \[ M_{axial} = \lambda_1 R_{eff} \]
  where,
  \[ R_{eff} = 0.5(R_{outer} + R_{inner}) \]

- A tangential moment due to the constraint Lagrange Multiplier forces, $\lambda_2$ and $\lambda_3$:
  \[ \lambda_{eff} = \sqrt{\lambda_2^2 + \lambda_3^2} \]
  \[ M_{tangential} = \lambda_{eff} R_{inner} \]

- A bending moment that is generated as a consequence of the constraint Lagrange Multiplier moments ($\lambda_5$ and $\lambda_6$):
  \[ M_{eff} = \sqrt{\lambda_5^2 + \lambda_6^2} \]
  Leading to a bending moment:
  \[ M_{bending} = 2.0R_{inner}M_{eff} / L_{eff} \]

Additionally, if interference fit moment ($M_{interference}$) is defined, the normal moment for frictional calculations is given by:

\[ M_n = |M_{interference} + M_{axial} + M_{tangential} + M_{bending}| \]

A similar calculation is carried out for the z-axis revolute joint by choosing the appropriate constraint Lagrange multiplier forces in the above equations.

Slot Joint

The two displacement constraint Lagrange Multiplier forces ($\lambda_2$ and $\lambda_3$) in the slot joint contribute to a tangential force as follows:

\[ F_t = \sqrt{\lambda_2^2 + \lambda_3^2} \]
Additionally, if interference fit force ($F_{\text{interference}}$) is defined, the normal force for frictional calculations is given by:

$$F_n = |F_{\text{interference}} + F_i|$$

Geometric quantities are not required for the slot joint.

**Translational Joint**

The geometric quantities required for the translation joint are:

- $L_{\text{eff}}$ = Effective length. The effective length is the length over which the two parts of the translation joint overlap. It is assumed that the change in this length is small.

- $R_{\text{eff}}$ = Effective radius. To simplify calculations, an effective radius is used in torsional moment calculations, even though the cross section in a translational joint is rectangular. The effective radius is used in computing the force that arises due to the torsional moment.

The normal force used in frictional calculations is computed as follows:

- An effective radial force due to the constraint forces ($\lambda_2$ and $\lambda_3$):
  $$F_{\text{eff}} = \sqrt{\lambda_2^2 + \lambda_3^2}$$

- Bending force due to in-plane constraint moments ($\lambda_5$ and $\lambda_6$):
  $$M_{\text{eff}} = \sqrt{\lambda_5^2 + \lambda_6^2}$$

Leading to a bending force

$$F_{\text{bending}} = \frac{2M_{\text{eff}}}{L_{\text{eff}}}$$

- Force due to the torsional constraint moment, $\lambda_4$:
  $$F_{\text{torsional}} = \frac{\lambda_4}{R_{\text{eff}}}$$

Additionally, if interference fit force ($F_{\text{interference}}$) is defined, the normal force for frictional calculations is given by:

$$F_n = |F_{\text{interference}} + F_{\text{eff}} + F_{\text{bending}} + F_{\text{torsional}}|$$

### 2.3.3. Reference Lengths and Angles for Joint Elements

The initial configuration of the joint element may be such that nonzero forces or moments is necessary. In such cases, you can define the constitutive behavior with respect to a reference configuration such that these forces or moments are zero. To do so, define a “reference angle” or a “reference length” (SECDATA).

If you do not define reference lengths and angles, ANSYS calculates the values from the initial configuration of the joints. ANSYS uses the reference lengths and angles in the stiffness and frictional behavior calculations.

### 2.3.4. Boundary Conditions for Joint Elements

Issue the DJ command to impose boundary conditions on the available components of relative motion of the joint element. You can list the imposed values via the DJLIST command. To delete the values, issue the DJDELE command.
To apply concentrated forces on the available components of relative motion of the joint element, issue the **FJ** command. You can list the imposed values via the **FJLIST** command. To delete the values, issue the **FJDELE** command.

### 2.3.5. Connecting Bodies to Joints

Other than in idealized geometry (such as that shown in *Figure 2.1: FE Slider-Crank Mechanism (p. 6)*), an **MPC184** joint element is defined by one or two nodes in space and requires special modeling techniques to connect the joint to the body appropriately.

*Figure 2.14: Pinned Joint Geometry (p. 28)* shows a 3-D model of a pinned joint where the geometry of the joint (the pin) is explicitly modeled. To perform a multibody analysis, the pin geometry is ignored and the behavior replaced by the appropriate **MPC184 joint element**.

---

*Figure 2.14: Pinned Joint Geometry* (p. 28)

---

*Figure 2.15: Pinned Joint Mesh and Revolute Joint (p. 29)* shows the meshed model including the revolute joint. To connect the bodies to the joint, you must use either elements (such as beams) or constraint equations. The easiest way to do so is to use contact elements to create surface-based constraints (multipoint constraints, or MPCs), as follows:

1. Define a pilot node at one end of the joint. The pilot node connects the joint to the rest of the body.
2. Select the nodes on the surface of the body that you want to connect to this pilot node.
3. Create contact surface elements on this surface. By sharing the same real constant number (REAL, N), MPCs between the surface nodes and the pilot node are generated during the solution.

Repeat the steps for each body-joint connection.

**Figure 2.15: Pinned Joint Mesh and Revolute Joint**

*Figure 2.16: Pinned Joint Contact Elements (p. 30)* shows the contact elements and *Figure 2.17: Pinned Joint Constraint Equations (p. 30)* shows the MPCs (constraint equations) created during the solution for the lower body.

Create the pilot node using the **TARGE170** element--setting KEYOPT(2) = 1 so as not to allow the program to constrain any DOFs--and issuing the **TSHAHP,PILO** command.

If you mesh the body with elements having no midside nodes (such as **SOLID185**), use **CONTA173** as the element type for the surface mesh. For elements with midside nodes (such as **SOLID186** or **SOLID187**), use **CONTA174**. Set the following element key options to create the necessary constraints:

- **KEYOPT(2) = 2** Constraint (MPC) option.
- **KEYOPT(4) = 2** Generate rigid MPC constraints.
- **KEYOPT(12) = 5** Bonded behavior between the pilot node and the contact surface.
Instead of the rigid option, you can also choose a flexible (force-distributed or RBE3-type) constraint option by setting KEYOPT(4) = 1. The following figures illustrate the difference in behaviors:
2.3.5. Connecting Bodies to Joints

**Figure 2.18: Rigid Constraint (KEYOPT(4) = 2)**

- Imposed displacement at Pilot node (UX, UY)
- Constraint surface remains rigid
- Contact elements

**Figure 2.19: Flexible Constraint (KEYOPT(4) = 1)**

- Imposed displacement at Pilot node (UX, UY)
- Deformed constraint surface
- Contact elements

**Typical Command Sequence**

Following is a typical command sequence for connecting bodies to joints:

```
! Step 1: Define a pilot node at the joint node
et,59,170          ! type ID=59 is an available ID
keyopt,59,2,1      ! do not allow program to constrain DOFs
```

Release 12.1 - © 2009 SASIP, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.
real,59                   ! real ID=59 is an available ID
tshap,pilot               ! "9536" is the joint node
e,9536                    ! Step 2: Select the nodes of the corresponding surface
csys,15                   ! CS at center of pin
nsel,s,loc,x,15           ! nodes at r=15
! Step 3: Create the contact elements on the surface
et,60,173                 ! constraint (MPC) option
keyopt,60,2,2             ! rigid MPC
keyopt,60,12,5            ! bonded always contact
type,60
real,59                   ! same real ID: this connects the pilot
! to this surface
esurf                     ! generate the contact elements on the surface
nsel,all

Additional Information

For more information about using contact elements to generate constraints, see Surface-Based Constraints in the Contact Technology Guide.
Chapter 3: Performing a Multibody Analysis

A multibody in ANSYS refers to a structural system consisting of flexible and rigid components. The following structural analysis types are available for multibody analysis: static, modal, harmonic, transient dynamic, spectrum, and buckling. For more information about each supported structural analysis type, see the Structural Analysis Guide.

The following topics present information necessary for performing a successful multibody analysis:

3.1. Kinematic Constraints
3.2. Convergence Criteria
3.3. Initial Conditions
3.4. Damping
3.5. Time-Step Settings
3.6. Solver Options

3.1. Kinematic Constraints

Kinematic constraints define how the structural system is held together geometrically. From a physical standpoint, a sufficient number of kinematic constraints including multipoint constraints (MPC), constraint equations (CE), coupling (CP) and boundary conditions (BC) are necessary for the system to be in stable equilibrium.

Providing sufficient kinematic constraints for a finite element model would lead to a full rank system of equations which would give a unique solution. Lack of sufficient kinematic constraints would make the system unstable. A finite element solution for such a system would fail to converge.

If more than sufficient kinematic constraints are specified for the structural system, the system may remain stable or become unstable. If the extra constraints conflict with the basic constraints necessary to keep the system in stable equilibrium, the system becomes unstable and the finite element solution fails with convergence problems. If the extra constraints do not conflict with the basic constraints, the system is consistently overconstrained and the extra constraints become redundant constraints. The system remains stable; however, there is no unique solution. Depending on how the equations for the finite element model are solved, the solution may or may not converge.

To ensure convergence of the finite element solution, the system must not be underconstrained or overconstrained. Checking for either lack of sufficient constraints or overconstraints can be difficult for complex systems, so ANSYS recommends performing a modal analysis on the system. If the modal analysis yields more zero eigenvalues than the rigid body modes of the system, the system lacks sufficient constraints; if there are fewer eigenvalues than rigid body modes, the system is overconstrained. A closer look at the unwanted eigenmodes can point to the missing or extra constraints.

3.2. Convergence Criteria

ANSYS provides suitable convergence checks by default, depending on the active degrees of freedom in the problem. You can activate additional convergence checks via the CNVTOL command.
3.3. Initial Conditions

Initial conditions define the state of the system at the start of the analysis. In structural finite element analyses, initial conditions are defined in terms of initial displacements, velocities, and accelerations at all independent degrees of freedom (DOFs).

Because all time-integration schemes (such as the Newmark method and the HHT method) rely on the history of displacements, velocities and accelerations, it is important to define consistent initial conditions. By default, a zero value is assumed for initial displacements, velocities, and accelerations at DOFs that are not otherwise specified (via the IC command).

Inconsistencies in initial conditions introduce errors into the time-integration scheme and lead to excitation of undesired (spurious) modes. Accumulation of these errors over several time increments adversely affects the solution and very often causes the time-integration scheme to fail. Applying numerical damping or other forms of damping can suppress the growth of these errors. However, such additions also affect the solution, especially, when long term transient behavior is being studied in the analysis.

It is not always possible, however, to have complete information about the initial state of a system being modeled for transient analysis. In such situations, it is helpful to run a dummy load step before the actual transient analysis of interest to bring the system into a consistent initial state. The purpose of such a load step is to eliminate the error introduced by inconsistent initial conditions.

Following are two ways to run a dummy load step:

- **3.3.1. Apply Linear Acceleration in a Dummy Transient Analysis**
- **3.3.2. Apply Large Numerical Damping Over a Short Interval**

### 3.3.1. Apply Linear Acceleration in a Dummy Transient Analysis

This technique is useful in cases where initial accelerations are non-zero, are known, and are uniform over the entire model. Applying acceleration loading (via the ACCEL command) introduces non-zero accelerations into the system. After the analysis has run through one substep, the actual transient analysis can be carried out without the acceleration loading.

**Example**

Consider a rigid beam of length $l$ rotating in the x-y plane about a pinned end at a constant angular velocity $\omega$. The free end of the beam has a tangential velocity of $\omega l$ and a centripetal acceleration of $\omega^2 l$. The beam is assumed to have all of its mass concentrated at the free end. To perform the analysis in ANSYS, model the rigid beam using the MPC184 element with Lagrange multipliers to enforce the rigid beam constraints. With one end of the rigid beam pinned, apply initial velocity normal to the beam axis at the free end. To introduce centripetal acceleration, use acceleration loading as illustrated in the following input file:

**Transient Analysis of a Rigid 3-D Beam Rotating About a Fixed Node**

```
/title, Transient analysis of a rigid 3-D beam rotating about a fixed node
/prep7

et, 1, mass21
keyopt, 1, 3, 2 ! 3d mass without rotary inertia
et, 2, mpc184
keyopt, 2, 1, 1 ! rigid beam
keyopt, 2, 2, 1 ! lagrange multiplier
n, 1, 0, 0, 0.0 ! pinned end (node 1)
n, 2, 1, 0, 0.0 ! free end (node 2)
```
3.3.1. Apply Linear Acceleration in a Dummy Transient Analysis

type,1
real,1
m = 1.0
r,1,m
en,1,2        ! 3D mass at free end (node 2)

finish
/solu
vel = 6.2831853072           ! tangential velocity
ic,2,uy,0.0,vel             ! initial condition for velocity
antype,trans
time,1,e-9
acel,0.0,-vel*vel,0.0       ! apply centripetal acceleration
kbc,1                       ! step loading
nlgeom,on
nsub,1,1,1                 ! use 1 substep for analysis
trnopt,full, , , , ,HHT     ! use HHT time integration
tintp,0.0                   ! no numerical damping
outres,all,all
solve
d,1,all
ddel,1,rotz
d,2,uz
d,2,rotx
d,2,roty

time,6.0
acel,0.0,0.0,0.0,0.0        ! remove centripetal acceleration
kbc,1                       ! automatic time stepping with MIDTOL
midtol,on,1e2
nsub,600,1e7,400            ! automatic time stepping with MIDTOL
trnopt,full, , , , ,HHT     ! use HHT time integration
tintp,0.05                  ! small numerical damping for HHT
outres,all,all
solve
finish
/post26
/xrange,0.,6.0
nsol,2,2,u,x,ux             ! x displacement for node 2
nsol,3,2,u,y,uy             ! y displacement for node 2
nsol,4,2,v,x,vx             ! x velocity for node 2
nsol,5,2,v,y,vy             ! y velocity for node 2
nsol,6,2,a,x,ax             ! x acceleration for node 2
nsol,7,2,a,y,ay             ! y acceleration for node 2
/axlab,x,Time T
/axlab,y,D/V/A
/gropt,divx,10
/gropt,divy,10
/gthk,curve,2
/title,Transient analysis of a rigid 3D beam rotating about a fixed node
plvar,ux,uy,vx,vy,ax,ay

finish
### 3.3.2. Apply Large Numerical Damping Over a Short Interval

This technique is of a more general nature and uses numerical damping to eliminate errors or numerical noise due to inconsistent initial conditions. After the noise has been damped out over several substeps, you can perform the actual transient analysis with smaller numerical damping.

Some potential drawbacks exist in cases where high frequency content of flexible multibody systems is important for analysis. Applying high numerical damping in the dummy analysis can affect the desired high-frequency response. ANSYS recommends using the HHT method for this technique because the integration scheme shows good dissipation properties with numerical damping.

#### Example

Consider a rigid-flexible double pendulum made up of a rigid and a flexible beam. One end of the rigid beam is pinned and the other end is hinged to the flexible beam. The other end of the flexible beam is free. The rigid beam is assumed to have all of its mass concentrated at the end that is hinged to the flexible beam. The system is given an initial velocity tangential to the flexible beam axis at its free end, as shown in the following input file:

**Transient Analysis of a Rigid-Flexible Double Pendulum**

```
/title,Transient analysis of a rigid-flexible double pendulum
/prep7

et,1,mass21
keyopt,1,3,2   !3d mass without rotary inertia
et,2,mpc184
keyopt,2,2,1,1 !rigid beam
keyopt,2,2,2,1 !lagrange multiplier
et,3,mpc184
keyopt,3,3,1,6 !revolute joint between rigid and flexible beam
et,4,beam188 !flexible beam

n,1,0.0,0.0    !pinned (supported) end of rigid beam
n,2,1.0,0.0    !hinged end of rigid beam (node 2)
n,3,1.0,0.0    !hinged end of flexible beam
n,4,1.25,0.0
n,5,1.5,0.0
n,6,1.75,0.0
n,7,2.0,0.0    !free end of flexible beam (node 7)

type,1
real,1
m,390
r,1,m
en,1,2         !3d mass at the end of rigid beam

type,2
real,2
en,2,1,2       !rigid beam

local, 11, 0, 0.0, 0.0, 0.0, , , 90
sectype, 3, JOIN, REVO, TESTREVO
secjoin, , 11, 11
type,3
real,3
secnum,3
en,3,3,2       !revolute joint

mp,ex,1,2e11   !material properties for flexible beam
mp,nuxy,1,0.3
mp,density,1,7.8e3

sectype,4,beam,csolid
secdata,1,0.1784124116 !c-s area is 0.1
```
3.4. Damping

You can specify two types of damping in ANSYS:

3.4.1. Numerical Damping
3.4.2. Structural Damping
3.4.1. Numerical Damping

Numerical damping is associated with the time-stepping schemes used for integrating second-order systems of equations over time. ANSYS provides the Newmark method and the HHT method for transient dynamic analysis of structural systems. Numerical damping for these schemes is determined by the parameter values specified via the TINTP command.

Numerical damping stabilizes the numerical integration scheme by damping out the unwanted high frequency modes. For the Newmark method, numerical damping also affects the lower modes and reduces the accuracy of integration scheme from second order to first order. For the HHT method, numerical damping affects only the higher modes and always maintains second-order accuracy.

ANSYS uses a default value (TINTP,GAMMA) of 0.005. The value that you select should be based on the problem at hand. A sensible value to try initially is 0.1. Use the lowest possible value that damps out non-physical response without significantly affecting the final solution. Problems involving rigid body translational motion, other forms of damping, or dissipative mechanisms like plasticity or friction typically require smaller values for numerical damping. Larger numerical damping values are usually necessary for problems involving rigid body rotational motion, elastic collisions (dynamic contact/impact), and large deformations with frequent changes in substep size.

3.4.2. Structural Damping

Structural damping refers to physical damping present in the system. You can specify the damping at the material level via viscous material models or dashpots (for example, COMBIN14 elements). At the structural level, you can specify it as modal damping or Rayleigh damping. For more information, see Damping in the Structural Analysis Guide.

3.5. Time-Step Settings

Transient dynamic analyses involving large deformations or large rotations exhibit significant changes in stiffness and inertia properties. The default response-frequency-based automatic time-stepping criterion may not be suitable for such nonlinear analyses. Use the MIDTOL command to automatically adjust the time increment based on convergence at the middle of the substep and convergence at the end of the substep. For more information, see "Nonlinear Structural Analysis" in the Structural Analysis Guide.

3.6. Solver Options

Multibody analyses generally involve large rotations in static or transient dynamics analysis, so nonlinear geometric effects must be accounted for. To do so, issue the NLGEOM,ON command.

For faster convergence in a full transient dynamic analysis where mass elements such as MASS21 are used, issue the NROPT,UNSYM command. The command activates the Newton-Raphson option for solving the nonlinear equations in the analysis, necessary due to the nonsymmetric stiffness contribution resulting from gyroscopic effects.
Chapter 4: Reviewing Multibody Analysis Results

Results from a flexible multibody analysis consist mainly of displacements, velocities, accelerations, stresses, strains, and reaction forces in structural components. Constraint forces, current relative positions, relative velocities, and relative accelerations in joint elements are also available.

Results are available for viewing in POST1, the general postprocessor (/POST1), or in POST26, the time-history postprocessor (/POST26).

For a description of the available output components, see the Output Data sections of the element descriptions for any of the elements that model the flexible components, rigid components, and joint elements.

The following topics concerning how to review flexible multibody analysis results are available:

4.1. Reviewing Results in POST1
4.2. Reviewing Results in POST26
4.3. Output of Joint Element Quantities
4.4. Energy Output

4.1. Reviewing Results in POST1

In the POST1 general postprocessor, only one substep at a time can be read, and the results from that substep must exist in the Jobname.RST file. The load step option command OUTRES controls which substep results are stored in Jobname.RST.

To review results in POST1:

- The database must contain the same model for which the solution was calculated.
- The Jobname.RST results file must be available.

A typical POST1 postprocessing sequence follows:

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
<th>Comments</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Verify from your output file (Jobname.OUT) whether the analysis converged at all load steps.</td>
<td>If not, you will likely not wish to postprocess the results, other than to determine why convergence failed. If your solution converged, then continue postprocessing.</td>
<td>---</td>
</tr>
<tr>
<td>2.</td>
<td>Enter the POST1 postprocessor.</td>
<td>If your model is not currently in the database, first issue a RESUME command.</td>
<td>/POST1</td>
</tr>
<tr>
<td>3.</td>
<td>Read the results for the desired load step and substep.</td>
<td>You can identify them by load step and substep numbers or by time.</td>
<td>SET</td>
</tr>
<tr>
<td>4.</td>
<td>View the results.</td>
<td>Use any of these options:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Display the deformed shape.</td>
<td>PLDISP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Display contours of stresses, strains, or any other applicable item.</td>
<td>PLNSOL or PLESOL</td>
<td></td>
</tr>
</tbody>
</table>
### 4.2. Reviewing Results in POST26

You can review the load-history response of a nonlinear structure using POST26, the time-history postprocessor (/POST26). Use POST26 to compare one ANSYS variable to another. For example, you could graph the relative rotation of a joint element versus time or any other variable.

A typical POST26 postprocessing sequence for a flexible multibody analysis is similar to the sequence for a typical nonlinear analysis, as follows:

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
<th>Comments</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Verify from your output file (Jobname.OUT) whether the analysis converged at all load steps.</td>
<td>Do not base design decisions on unconverged results. If your solution converged, continue postprocessing.</td>
<td>---</td>
</tr>
<tr>
<td>2.</td>
<td>Enter the POST26 postprocessor.</td>
<td>If your model is not currently in the database, first issue a RESUME command.</td>
<td>/POST26</td>
</tr>
<tr>
<td>3.</td>
<td>Define the variables to be used in your postprocessing session.</td>
<td>The SOLU command causes various iteration and convergence parameters to be read into the database, where you can incorporate them into your postprocessing.</td>
<td>NSOL, ESOL, RFORCE</td>
</tr>
<tr>
<td>4.</td>
<td>Graph or list the variables.</td>
<td>---</td>
<td>PLVAR (graph), PRVAR (list), EXTREM (list)</td>
</tr>
</tbody>
</table>

Many other postprocessing functions are available in POST26. For more information, see "The Time-History Postprocessor (POST26)" in the Basic Analysis Guide.
4.3. Output of Joint Element Quantities

Several joint element output quantities are available for review purposes. You can use either POST1 or POST26, or both, to review those results.

**The solution output associated with the element is in two forms:**

- Nodal displacements included in the overall nodal solution
- Additional element output to the results file listed below

**The following output is available for joint elements as SMISC quantities:**

- Constraint forces and moments
- Constraint forces (moments) if stop is specified
- Constraint forces (moments) if lock is specified
- Stop status
- Lock status
- Relative position
- Constitutive displacements and rotations
- Joint elastic forces (moments)
- Joint damping forces (moments)
- Joint friction forces (moments)
- Relative displacement and rotations (cumulative)
- Relative velocities
- Relative accelerations
- Average temperature in the element

**The following output is available for joint elements as NMISC quantities:**

- The components of the bases vectors at the two nodes in the deformed configuration.
  
  The bases vectors are specified as the local coordinate systems via the SECJOINT command and evolve with the rotation of the underlying nodes.

- The constraint forces and moments in the evolved basis at the first node of the joint element.

The ANSYS Workbench Products generally use NMISC output for postprocessing.

See the MPC184 element documentation and the individual joint element descriptions for details about the SMISC component specification and the use of the ETABLE command.

In POST1, you can print joint element output (such as relative reaction forces, relative displacements, relative rotations, etc.) at the free or unconstrained relative degree of freedom via the PRJSOL command. To obtain the nodal forces at the joint element nodes, issue the PRESOL, FORC command.

In POST26, you can use the JSOL command to specify result items (such as relative displacements, velocities, accelerations, etc.) that must be stored for a joint element. Then, you can plot or print the stored items via the PLVAR or PRVAR command, respectively.
4.4. Energy Output

You can monitor the total energies of the entire model in POST1 via the PRENERGY command. The total energy consists of elastic, kinetic, artificial hourglass/drill stiffness energy, and so on.

In POST26, you can use the ENERSOL command to store a specific energy item. Then, you can graph or list the specific energy item in the output file via the PLVAR or PRVAR command, respectively.
Chapter 5: Using Component Mode Synthesis Superelements in a Multibody Analysis

Obtaining the flexible response of a body or bodies to a dynamic motion event typically involves solving hundreds or thousands of time points. If a flexible body has many degrees of freedom (DOFs), a multibody analysis can be time-consuming. To minimize the necessary computing resources, you can use component mode synthesis (CMS) superelements (substructures) to replace the many thousands of DOFs of the flexible body with tens of DOFs that represent the dynamic response, thereby significantly reducing the required multibody analysis run time.

The following topics describe the approach required to perform a substructure-based multibody analysis, including recovering the time-dependent flexible response:

5.1. Applicability of CMS Superelements in a Multibody Analysis
5.2. Flexible Body Types
5.3. Substructuring Overview
5.4. Master Degrees of Freedom in a Substructured Multibody Simulation
5.5. Steps for Performing a Substructured Multibody Simulation

For an example of how to set up and use a substructuring in a multibody analysis, see Chapter 6, Example Multibody Analysis: Crank Slot Mechanism (p. 53).

5.1. Applicability of CMS Superelements in a Multibody Analysis

The flexible body to be substructured is assumed to behave in a linear elastic manner, as follows

• Only linear materials are allowed.
• Nonlinear elements within the body (such as gasket or contact elements) are treated as linear and in their initial state.
• The body may consist only of 3-D structural elements. (You can use 2-D elements with care provided that you follow the guidelines given later, particularly with respect to the number of DOFs at the master DOFs.)
• Element formulations using Lagrange multipliers are not allowed.
• Density or mass of some form must be present in the body.

The body may undergo large rotations, but the strains and relative rotations within the body are presumed to be small.

5.2. Flexible Body Types

A multibody simulation supports two types of flexible bodies:

• Bodies that are excited by the motion of other bodies (rigid or flexible) but do not themselves undergo large motions
An engine block is an example of this type, where the block is excited dynamically from the crankshaft, pistons, and other moving parts attached or linked to the block. This case is a straightforward application of traditional superelements.

- **Bodies that are undergoing large motions**

  A piston rod is an example of a body undergoing large motions; this type also uses superelements but with the additional capability that the superelement can undergo large motions, and large rotations in particular. A large-rotation superelement involves additional considerations.

### 5.3. Substructuring Overview

Substructuring is a technique that condenses a group of finite elements into a single element represented as a matrix. The single-matrix element is called a superelement. You can use the superelement in an analysis as you would any other ANSYS element type.

Substructuring requires three passes:

- A **generation pass**, where the group of elements are condensed down to generate the superelement.
- A **use pass**, where the superelement is used in the analysis. In our case, in the multibody analysis.
- An **expansion pass**, where the results of the superelement in the use pass are expanded to the original group of elements so that their displacements, forces, strains, and stresses are recovered.

In the use pass, ANSYS allows the superelement to rotate with arbitrarily large rotations.

In the generation pass, you define master degrees of freedom (MDOFs). The MDOFs are the DOFs that the superelement uses to interface with, or connect to, the other bodies or joints.

Because the flexible body analysis occurs within a dynamic analysis, you must include the dynamic (mass) effects. Use component mode synthesis (CMS) to augment the superelement static stiffness with mode shapes that characterize the dynamic behavior, much as you would when performing a mode-superposition transient dynamic analysis.

CMS is a form of substructure analysis allowing you to derive the dynamic behavior of the entire assembly from its constituent components. For more information, see "Component Mode Synthesis" in the Advanced Analysis Techniques Guide.

### 5.4. Master Degrees of Freedom in a Substructured Multibody Simulation

The master degrees of freedom (MDOFs) are the degrees of freedom (DOFs) of the superelement which you intend to use to connect to the DOFs of the remaining bodies and joints. Because you almost always use all the DOFs of a node in the definition of the MDOFs, you can think in terms of master “nodes”; that is, the MDOFs are the nodes of the superelement that connect to the nodes of the remaining joints and bodies.

If the connection occurs at a joint at the center of a hole or slot, you must place a master node there. For more information, see Connecting Bodies to Joints (p. 28).

**Nonrotating Bodies**

For nonrotating bodies, master nodes are located at the points where the superelement connects with the other bodies and are typically located at the centers of bolts or other fasteners and bearings. Try to minimize the number of master nodes. Where appropriate, use the techniques presented in Connecting Bodies to Joints (p. 28) to create a single master node that connects to a number of nodes.
Rotating Bodies

For *rotating bodies*, the idea is to create a beam-like superelement, ideally with two master nodes (but never less than two). You can use more than two master nodes (for example, when modeling a lever or rocker plate), but ANSYS assumes that the rotation of the superelement is the average of the rotations of all master nodes.

All master nodes of a rotating body must have six active structural DOFs: UX, UY, UZ, ROTX, ROTY, and ROTZ. If the master node does not have six DOFs—for example, if it is the node of a 3-D solid element—create a six-DOF node at that location and tie it to the rest of the body appropriately. You can use either of the following techniques, both of which essentially place a six-DOF node connected to a patch of elements superimposed on the existing solid elements.

- **MPC Contact** -- Create a pilot node and link it to bonded contact elements overlaid on the patch. For more information, see Connecting Bodies to Joints (p. 28).

- **Beams** -- Overlay beam elements or MPC184 Rigid Beam elements in a “spider web” fashion. The beams should have high stiffness and no mass.

Following is an illustration of both methods:

When “rotating” the created node, the body rotates accordingly.

You can also define MDOFs where loads are to be applied as well as at any points where velocities or accelerations are of interest.

5.5. Steps for Performing a Substructured Multibody Simulation

The methodology for performing a substructured multibody simulation assumes that you have generated the entire finite element model of the multibodies including the joints--using ANSYS Workbench, for example--and want to take advantage of substructuring to reduce the solution time. ANSYS refers to this method as a top-down approach (as opposed to a bottom-up approach of defining the substructure first and then building the rest of the model around it).

Using substructures to represent some or all of the flexible bodies in a completely defined multibody model requires the following steps:

5.5.1. Step 1: Prepare the Full Model for a Substructured Multibody Analysis
5.5.2. Step 2: Create the Substructures (Generation Pass)
5.5.3. Step 3: Build the CMS-based Model (Use Pass)
5.5.4. Step 4: Run the Multibody Analysis
5.5.5. Step 5: Expand all Solutions (Expansion Pass)
5.5.6. Step 6: Create the Merged Results File
5.5.7. Step 7: Postprocess the Results
Before proceeding, prepare the full multibody model (as described in Steps 1 through 4 in Overview of the ANSYS Multibody Analysis Process (p. 2)). Verify that the bodies are connected to the joints as described in Connecting Bodies to Joints (p. 28).

The multiple passes used in substructuring require that the files created and used in the process are handled appropriately. To aid in file management when performing a substructured multibody simulation, use the /FILNAME command to modify the current jobname as needed.

### 5.5.1. Step 1: Prepare the Full Model for a Substructured Multibody Analysis

Prepare the full model for a substructured multibody analysis, as follows:

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
<th>Comments</th>
<th>Command(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Specify the full jobname.</td>
<td>Example: /FILNAME,FULL</td>
<td>/FILNAME</td>
</tr>
<tr>
<td>1.2</td>
<td>Resume (or build) the full model.</td>
<td>See Overview of the ANSYS Multibody Analysis Process and Connecting Bodies to Joints.</td>
<td>RESUME</td>
</tr>
<tr>
<td>1.3</td>
<td>Make components of the flexible body. (Repeat for each flexible body.)</td>
<td>Create an element component of the elements of the body, including any contact elements used to connect the body to a joint. (Do not include the joint elements.)</td>
<td>ESEL CM,Ename,ELEM</td>
</tr>
<tr>
<td>1.4</td>
<td>Select the entire model.</td>
<td>---</td>
<td>ALLSEL</td>
</tr>
<tr>
<td>1.5</td>
<td>Save the model.</td>
<td>---</td>
<td>SAVE</td>
</tr>
</tbody>
</table>

### 5.5.2. Step 2: Create the Substructures (Generation Pass)

Perform the generation pass to create the CMS substructure (in the matrix .SUB file) characterizing the dynamic flexibility of the body.

You must decide how many modes to include in the CMS substructure. The number you determine depends on several factors including:

- The driving frequency.
- The frequencies to be excited (such as flexural, axial, torsional, etc.).
- Whether displacements are of primary interest, or whether stresses/strains (or fatigue) are of primary interest. (The latter require more modes to accurately capture their response.)
- Whether impact (contact) is included. (Impact tends to excite higher frequencies.)
- Whether acoustic frequencies are desired.

For most analyses, and particularly for rotating bodies, the fixed-interface method (CMSOPT,FIX) is sufficient. For analyses where higher frequencies are of interest (for example, those involving acoustics or high-speed equipment), the residual-flexible free-interface method (CMSOPT,RFFB) provides more accuracy. For more information, see CMS Methods Supported in the Advanced Analysis Techniques Guide.

For nonrotating bodies, you can apply constraints (D) in the generation pass to the degrees of freedom (DOFs), but not the master degree of freedom (MDOF). Set KEYOPT(4) = 1 for these superelements in the use pass; otherwise, your analysis will have convergence problems. For rotating bodies, do not apply constraints in the generation pass because the superelement must have six rigid body modes; you can, however, apply constraints to its MDOF in the use pass.
Loading Considerations

When applying loads, be aware that:

- The loads rotate with the rotating substructure by default. This behavior is valid for most load types (especially pressure loads). In the use pass, however, you can specify that the load vector *not* rotate with the substructure; disabling load rotation is useful in some cases, such as those involving nodal forces where you want to maintain their original direction.

- When to apply gravity and other acceleration loads (such as those applied via `ACEL` and `OMEGA` commands) depends on whether the body is rotating or not. For a rotating body, apply the loads in the use pass. For a nonrotating body, you can apply the loads in this step and use it in the use pass; however, be careful not to specify it twice (for example, by issuing an `ACEL` command in the use pass). Issue the `CMACEL` command to apply the acceleration to the nonstructured elements only.

- By applying a unit load in this step, you can easily scale it in the use pass and make use of tabular loads to apply a complex load-versus-time history in a single load step. ANSYS recommends this approach as it allows for straightforward creation of the full model results file.

Creating the Superelements

Follow these steps to create the superelements for a substructured multibody analysis:

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
<th>Comments</th>
<th>Command(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Clear the database.</td>
<td>Required only if performing this step in the same session as the prior step.</td>
<td><code>/CLEAR</code></td>
</tr>
<tr>
<td>2.2</td>
<td>Specify the <em>generation pass</em> job-name.</td>
<td>Example: <code>/FILNAME,BODY1</code></td>
<td><code>/FILNAME</code></td>
</tr>
<tr>
<td>2.3</td>
<td>Resume the full model.</td>
<td>Example: <code>RESUME,FULL.DB</code></td>
<td><code>RESUME</code></td>
</tr>
<tr>
<td>2.4</td>
<td>Define the analysis type.</td>
<td>The analysis type is <em>substructure</em>.</td>
<td><code>/SOLANTYPE,SUBSTR</code></td>
</tr>
<tr>
<td>2.5</td>
<td>Define substructure options.</td>
<td>Substructure name, and generate stiffness and mass, as in this example: <code>SEOPT,BODY1SE,2</code></td>
<td><code>SEOPT,Sename,2</code></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CMS options, including the number of modes.</td>
<td><code>CMSOPT,FIX,NMODE</code></td>
</tr>
<tr>
<td>2.6</td>
<td>Select the substructure nodes and elements.</td>
<td>Select the elements defined in Step 1.3.</td>
<td><code>CMSEL,S,ELEM</code></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Select the interface nodes defined in Step 1.3</td>
<td><code>CMSEL,S,NODE</code></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Create master degrees of freedom (MDOFs) at all selected nodes.</td>
<td><code>M,ALL,ALL</code></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Select the nodes attached to the elements.</td>
<td><code>NSLE</code></td>
</tr>
<tr>
<td>2.7</td>
<td>Apply loads, if any.</td>
<td>These are loads typically interior to the body (that is, not applied to a MDOF).</td>
<td><code>F SF SF ACEL</code></td>
</tr>
</tbody>
</table>

Release 12.1 - © 2009 SASIP, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.
Repeat the steps above for each flexible body you wish to replace with CMS substructures. Use unique job-names and substructure names for each flexible body.

Residual-Flexible Free-Interface CMS Method

If you are using the residual-flexible free interface method, use \texttt{CMSOPT,RFFB,NMODE} (rather than \texttt{CMSOPT,FIX,NMODE}) in Step 2.5. You must also define pseudo-constraints (D,,SUPPORT).

For further information, see The CMS Generation Pass: Creating the Superelement in the Advanced Analysis Techniques Guide.

5.5.3. Step 3: Build the CMS-based Model (Use Pass)

Replace the flexible bodies with their corresponding CMS substructures.

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
<th>Comments</th>
<th>Command(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Clear the database.</td>
<td>Required only if performing this step in the same session as the prior step.</td>
<td>/CLEAR</td>
</tr>
<tr>
<td>3.2</td>
<td>Specify the use pass jobname.</td>
<td>Example: /FILNAME,USE</td>
<td>/FILNAME</td>
</tr>
<tr>
<td>3.3</td>
<td>Resume the full model.</td>
<td>Example: RESUME,FULL.DB</td>
<td>RESUME</td>
</tr>
<tr>
<td>3.4</td>
<td>Replace the flexible bodies.</td>
<td>Deselect the flexible elements.</td>
<td>/PREP7 CMSEL,U,Ename ET,ITYPE,50</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Define the substructure element type using an available type number (ITYPE).</td>
<td>KEYOPT,ITYPE,3,1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If any loads were applied in Step 2 and you do not want them to rotate with the substructure, set the appropriate key option.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>For nonrotating substructures that have constraints applied in the generation pass, set the appropriate key option.</td>
<td>KEYOPT,ITYPE,4,1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Define the substructure.</td>
<td>TYPE,ITYPE SE,Sename</td>
</tr>
</tbody>
</table>

Repeat the \texttt{CMSEL,U} and \texttt{SE} commands for each flexible body.

\textbf{Caution}

Be careful not to select all elements (for example, via an \texttt{ALLSEL} command) before initiating the solution (\texttt{SOLVE}) in the next step. If you do so, ANSYS solves for both sets of elements.
5.5.4. Step 4: Run the Multibody Analysis

Set up the multibody analysis and run it.

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
<th>Comments</th>
<th>Command(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Specify the analysis type.</td>
<td>Large deflection, transient analysis (multibody analysis).</td>
<td>/SOLU ANTYPE,TRANS</td>
</tr>
<tr>
<td>4.2</td>
<td>Specify the transient analysis options.</td>
<td>HHT method with 0.1 numerical damping.</td>
<td>TRNOPT,FULL,,,,,HHT TINTP,0.1</td>
</tr>
<tr>
<td>4.3</td>
<td>Specify boundary conditions.</td>
<td>Constraints on motion and initial conditions</td>
<td>D DJ IC …</td>
</tr>
<tr>
<td></td>
<td>Applied loads, including applying loads from the generation pass Step 2.7 (SFE„,SELV)</td>
<td></td>
<td>F FJ ACEL SF SFE …</td>
</tr>
<tr>
<td>4.4</td>
<td>Specify load step options and solve.</td>
<td>Ending time and time step sizes.</td>
<td>TIME DELTIM or NSUBST</td>
</tr>
<tr>
<td></td>
<td>Results file output controls.</td>
<td></td>
<td>OUTRES</td>
</tr>
<tr>
<td></td>
<td>Run the analysis.</td>
<td></td>
<td>SOLVE</td>
</tr>
</tbody>
</table>

To dampen out excessive solution noise, particularly in the velocities and accelerations, you typically use numerical damping. For more information, see Damping (p. 37).

In Step 4.3, use tabular loads to specify complex load-versus-time histories. By default, loads are simply ramped (or step-applied [KBC]) over the time interval from one load step to the next. Tabular loads, however, allow a general load curve. To use multiple load steps to define the loading, repeat Steps 4.3 and 4.4 for each load configuration.

For more information about setting up and performing a multibody analysis, see Chapter 3, Performing a Multibody Analysis (p. 33).

5.5.5. Step 5: Expand all Solutions (Expansion Pass)

Using the solutions from the prior step (displacements at the MDOFs at each time point), obtain the displacements and stresses (if desired) for all nodes and elements of the flexible bodies.

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
<th>Comments</th>
<th>Command(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1</td>
<td>Clear the database.</td>
<td>Required only if performing this step in the same session as the prior step.</td>
<td>/CLEAR</td>
</tr>
<tr>
<td>5.2</td>
<td>Specify the generation pass job-name from Step 2.</td>
<td>Example: /FILNAME,BODY1</td>
<td>/FILNAME</td>
</tr>
<tr>
<td>5.3</td>
<td>Resume that jobname's database.</td>
<td>No file name required.</td>
<td>RESUME</td>
</tr>
<tr>
<td>5.4</td>
<td>Specify an expansion pass.</td>
<td>---</td>
<td>/SOLU</td>
</tr>
</tbody>
</table>
Repeat all steps for each substructured body (including clearing the database [/CLEAR]).

5.5.6. Step 6: Create the Merged Results File

Merge all results files (one from the use pass and one from each of the expanded substructures) to create a results file with the full model data. After completing this part of the process, you can perform postprocessing as though you had run the full model in the multibody simulation.

Understanding the example commands in this step:

- **NSUBSTEPS** is the total number of substeps (time points) in the results files.
- In the example commands, the jobname from the use pass (Step 3) is USE; therefore, its results file is named USE.RST. Likewise, the jobname from the expansion pass (Step 5) is BODY1; therefore, its results file is named BODY1.RST. Adjust the command arguments accordingly to accommodate your own jobnames.
5.5.7. Step 7: Postprocess the Results

Postprocess the full model as though you had run a nonsubstructured analysis.

Use the POST1 postprocessor (/POST1) to review the results over the entire model. Use the POST26 postprocessor (/POST26) to obtain time-history listings and plots. For more information, see Chapter 4, Reviewing Multibody Analysis Results (p. 39) for specific multibody postprocessing.

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
<th>Comments</th>
<th>Command(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1</td>
<td>Specify the full model jobname from Step 1.</td>
<td>Example: /FILNAME,FULL</td>
<td>/FILNAME</td>
</tr>
<tr>
<td>7.2</td>
<td>Resume that jobname's database.</td>
<td>No file name required.</td>
<td>RESUME</td>
</tr>
<tr>
<td>7.3</td>
<td>Review results at a specific point in time</td>
<td>---</td>
<td>/POST1 SET,</td>
</tr>
<tr>
<td>7.4</td>
<td>Select the entire model.</td>
<td>---</td>
<td>/POST26</td>
</tr>
</tbody>
</table>

Nodal velocity and acceleration nodal results are not available for the substructure interior nodes (non-MDOFs).
Chapter 6: Example Multibody Analysis: Crank Slot Mechanism

The example crank slot analysis in this section introduces you to the ANSYS program’s multibody analysis capabilities. To facilitate modeling and simulation in a multibody analysis, ANSYS, Inc. suggests using the ANSYS Workbench product along with the ANSYS program to develop your analysis. The input files used to run the crank slot analysis in the ANSYS program were generated by ANSYS Workbench.

The following topics are available for this example multibody analysis of a crank slot mechanism:
- 6.1. Problem Description
- 6.2. Problem Specifications
- 6.3. Defining Joints
- 6.4. Performing the Rigid Body Analysis
- 6.5. Performing the Flexible Body Analysis
- 6.6. Using Component Mode Synthesis in the Multibody Analysis
- 6.7. Using Joint Probes
- 6.8. Comparing Processing Times
- 6.9. Input Files Used in This Analysis

6.1. Problem Description

The crank slot model consists of several parts connected by joint elements. Perform a simulation using multibody dynamics to study the motion of the crank mechanism and the joint forces when starting the mechanism at one of the joints from rest with a rotational acceleration of 25 rad/sec^2. In this problem, it is also important to examine the transient stress results in one of the slider rods.

6.2. Problem Specifications

The geometry for the crank slot model consists of a base and two rods. The two rods are attached to each other and the base with three bolts. The material used for all components is structural steel.
The material properties for this analysis are as follows:

- Young's modulus (E) = 2e+005 MPa
- Poisson's ratio (ν) = 0.3
- Density = 7.85e-006 kg/m

6.3. Defining Joints

Define the joints that connect the parts of the crank slot model. Revolute, slot, and cylindrical joints form the moving joints. The base of the model is fixed to the ground via a fixed joint.

The following figure shows the parts of the model, with the joints listed to the right:
All joints are available via the MPC184 element’s KEYOPT(1) setting and, in some cases, the KEYOPT(4) setting. For more information, see *Connecting Multibody Components with Joint Elements* (p. 14).

6.4. Performing the Rigid Body Analysis

Run the crank slot analysis using a rigid body specification. Specifying a body as rigid in ANSYS models it as a combination of:

- A MASS21 element at the center of gravity (CG) of the parts, and
- MPC184 elements for the joints connected to each other via rigid body nodes.

For more information, see *Modeling Rigid Bodies in a Multibody Analysis* (p. 7). The input file CrankSlot_Rigid.inp (available on the ANSYS distribution media) is used to perform the rigid body portion of the analysis.

The following figures show the finite element (FE) representation of the model and the time-history plot of the total displacement of the rigid Rod2 part:
6.5. Performing the Flexible Body Analysis

Run the crank slot analysis using a flexible approximation for the Rod2 part. After defining Rod2 as a flexible body, mesh it using ANSYS 3-D SOLID186 elements. (At this stage, the remaining parts are still considered to be rigid.)

For more information, see *Modeling Flexible Bodies in a Multibody Analysis* (p. 5). The input file Crank-Slot_Flexible.inp (available on the ANSYS distribution media) is used to perform the flexible body portion of the analysis.

The following figures show the FE representation of the flexible Rod2 part and a representation of the entire model:
6.5. Performing the Flexible Body Analysis
6.6. Using Component Mode Synthesis in the Multibody Analysis

**CMS a Powerful Tool**

Using CMS for static and transient nonlinear analysis reduces problem size and minimizes CPU-resource requirements. You can convert parts of a model which exhibit linear behavior (such as Rod2 in this case) to a superelement using CMS with large rotation. You can restrict all geometric, contact, and material nonlinearity to those parts of the model which require nonlinear behavior.

For more information, see *Chapter 5, Using Component Mode Synthesis Superelements in a Multibody Analysis* (p. 43) and "Component Mode Synthesis" in the *Advanced Analysis Techniques Guide*.

Using the flexible body created previously, create a component mode synthesis (CMS) model with large rotation. Using CMS for the multibody analysis consists of:

1. Creating a superelement of the flexible body (generation pass).
2. Using the superelement in the transient analysis (use pass).
3. Recovering stress and displacement results for the entire model (expansion pass).

The results are similar to those of the flexible model, as shown:

To leverage the advantage of a CMS analysis for large rotation, define another part of the model, Rod1, as a flexible body. Define the other flexible part, Rod2, as a CMS part. The input file `CrankSlot_FlexibleCMS.inp` (available on the ANSYS distribution media) is used to perform the CMS portion of the analysis.
The CMS part Rod2 assumes linear behavior with large rotations, whereas the flexible part Rod1 retains all geometric and material nonlinearity in the model, as shown:

### 6.7. Using Joint Probes

In addition to information about the displacement and stress in the structure, you can use the joint probes to obtain specific results information about the various joints in the model. Here the total force at a single joint is plotted as a function of time:
6.8. Comparing Processing Times

Comparison of the CPU times shows the advantage of using CMS even for a simple model such as the crank slot. The benefits of CMS for large-rotation and nonlinear analyses can multiply in cases involving larger and more complex models, especially those exhibiting more nonlinear behavior.

6.9. Input Files Used in This Analysis

The following ANSYS input files (available on the ANSYS distribution media) are used in the example analysis of the crank slot mechanism described in this section. The files were generated by the ANSYS Workbench product.

- CrankSlot_Rigid.inp
- CrankSlot_Flexible.inp
- CrankSlot_FlexibleCMS.inp
Chapter 7: Troubleshooting a Flexible Multibody Analysis

A successful flexible multibody simulation involves proper element selection, appropriate material behavior, and proper application of load and boundary conditions. To troubleshoot problems, debugging must occur at all levels of the analysis. Typical questions requiring answers include:

- Is the choice of elements appropriate for this analysis? (For more information, see *Element Choices for Flexible Bodies* (p. 6), *Defining a Rigid Body* (p. 7), and *Connecting Multibody Components with Joint Elements* (p. 14).)
- Does the chosen material model correctly represent the actual material behavior?
- Are the loading and boundary conditions appropriately modeled?
- Are overconstraint conditions causing convergence problems?
- Do the problem's physics indicate global or local buckling issues that must be addressed?

Although other topics in this document provide guidelines for element selection, modeling, and solver options while setting up your multibody analysis, the following troubleshooting topics are available to help you achieve a successful multibody simulation:

7.1. Addressing Overconstraint Issues During Modeling

7.1.1. Overconstraints in Rigid Bodies

Overconstraints may arise when rigid bodies are joined together using multiple joint elements. The overconstraints can occur due to redundant joints performing the same function or contradictory motion resulting from improper use of joints connecting different bodies.
The following examples illustrate scenarios in which overconstraint conditions can occur.

### 7.1.1.1. Standard Four-Bar Mechanism

In this scenario, all components are rigid. The example shows how overconstraint can occur even in simple models.

Consider the standard 3-D four-bar mechanism shown here. (See Geradin and Cardona in *Learning More About Multibody Dynamics* (p. 3).) The mechanism consists of four rigid links and four revolute joints.

**Figure 7.1: Overconstrained System: Standard 3-D Four-Bar Mechanism**

![Diagram of a standard 3-D four-bar mechanism with revolute joints and a pilot node.](image)

**Solution:** Replace three of the revolute joints with spherical joints.

With six DOFs available for each rigid body, the four rigid bodies yield a total of $6 \times 4 = 24$ DOFs. A revolute joint has only one free DOF and five constraints. Thus, the four revolute joints impose a total of $5 \times 4 = 20$ constraints. If one of the rigid links is fixed in space, then an additional six constraints are imposed. If a rotation is applied at one of the revolute joints (thereby adding one more constraint), the number of overconstraints is $24 - (20 + 6 + 1) = -3$. As modeled, therefore, this mechanism is overconstrained.

In this case, you can resolve the overconstraints by replacing three of the revolute joints with spherical joints. Each spherical joint imposes only three constraints; after replacing the joint type, a DOF count indicates that the system is no longer overconstrained. While the overconstraint in this model can be resolved fairly easily, this is not a typical case. It is therefore vital that you exercise care when setting up your model. For more information, see *Resolving Overconstraint Problems* (p. 64).

### 7.1.1.2. Redundant Rigid Bodies

This simple example illustrates overconstraints caused by redundant rigid components.
The figure shows a plate modeled with shell elements. A portion of the plate is made rigid by adding MPC184 Rigid Beam elements (represented by the thick lines in the figure). The addition of rigid beams AB and BC is redundant and leads to an overconstrained model.

In ANSYS, if the MPC184 Rigid Beam elements with direct elimination option are used to model this type of problem, the redundant constraints are eliminated automatically. However, if MPC184 Rigid Beam with the Lagrange multiplier option is used, the solution may not converge.

### 7.1.1.3. Redundant Boundary Conditions

Redundant boundary conditions can lead to overconstraint. In some cases, the multibody mechanism may actually end up as a “structure” with zero mobility if improper boundary conditions are applied.

In some cases involving MPC184 Rigid Beam elements with the direct elimination option (which is based on all DOFs at a node), redundant boundary conditions can result in an overconstrained system.

Consider a cylindrical tube with one end fixed and subjected to a bending moment at the other end. A quarter of the cylinder is modeled with appropriate symmetry and antisymmetry boundary conditions as shown in the following figure. MPC184 Rigid Beam elements with the direct elimination option connect all the nodes of the tube to a center point, and a moment is applied at the center node.
7.1.2. Overconstraints Caused by User-Defined Constraint Equations

User-defined constraint equations (created via the CE and CP commands) can conflict with the internal constraint equations generated for the rigid bodies using the contact MPC capability or the joint elements. ANSYS recommends avoiding user-defined CEs and/or CPs while performing a flexible multibody simulation.

7.2. Resolving Overconstraint Problems

Overconstraint problems frequently arise in multibody system models containing rigid bodies. Overconstraints in the model can result in nonconvergence, slow convergence, solver small pivot messages, and in some cases an incorrect solution. Often, overconstraint problems are not readily identifiable. For example, even adding flexibility to the model may not completely resolve an overconstraint problem. It is therefore vital that you address overconstraint issues during the modeling phase if possible instead of trying to resolve overconstraint problems afterwards.

ANSYS does not resolve overconstraints automatically. To check for overconstraints, model the multibody mechanism as a rigid mechanism using a rigid body solver.

Following are some hints to help you resolve overconstraint problems:

- Perform a DOF count in the mechanism. Various methods are available for evaluating the number of free DOFs in a given rigid body mechanism. See Learning More About Multibody Dynamics (p. 3).
- Know the number of constraints for each joint element. In some cases, replacing one type of joint with another may resolve an overconstraint issue. Check the number of constraints for a given joint and replace it with a simpler one if possible. For example, a revolute joint (which imposes five constraints) can possibly be replaced by a cylindrical joint (which imposes only four constraints). For more information, see Joint Element Types (p. 15).
- A translational joint fixes five DOFs while allowing motion in only one direction. You may be able to replace it with a slot joint which allows more free relative DOFs.
7.2. Resolving Overconstraint Problems

- The local axes specified at the joint element nodes must be defined properly. Improper definitions result in unanticipated motion or constraints. For example, if you define the four-bar mechanism in Figure 7.1: Overconstrained System: Standard 3-D Four-Bar Mechanism (p. 62) in a plane other than one of the global Cartesian planes, verify that the joint coordinate systems for each joint align.

- Perform a modal analysis to ensure that appropriate modes are present in the idealized model of the mechanism. Overconstraints can lead to modes that are not usually present in the actual system.

- Use more flexible components in the model. Avoid models with only rigid bodies, which can lead to solver difficulties.

- Avoid external (user-defined) constraint equations (CE and CP). They may conflict with those generated internally by ANSYS for contact with MPC and the joint elements.

- Check the model for redundant boundary conditions.

- Do not mix MPC184 Rigid Beam/Link and MPC184 Joint elements implemented using the Lagrange multiplier method with those implemented using the direct elimination method.
Index

M
multibody analysis
  additional sources of information, 3
  ANSYS-ADAMS interface, 3
  boundary conditions for rigid bodies, 12
  complex model representation using rigid bodies, 13
  connecting bodies to joints, 28
  connecting flexible and/or rigid components, 14
  connecting joint elements to rigid bodies, 13
  convergence criteria, 33
  damping methods, 37
  defining a rigid body, 7
  definition, 33
  element choices for flexible bodies, 6
  energy output, 42
  example analysis: crank slot mechanism, 53
  finite element method benefits, 1
  flexible body modeling, 5
  initial conditions, 34
  introduction, 1
  joint element types, 15
  kinematic constraints, 33
  material behavior in joint elements, 22
  modeling contact with rigid bodies, 14
  modeling criteria, 5
  overconstraint problems, 61
  POST1 results, 39
  POST26 results, 40
  process overview, 2
  results viewing, 39
  rigid body DOFs, 11
  rigid body modeling, 7
  SMISC quantities for joint elements, 41
  solver options, 38
  time stepping, 38
  troubleshooting, 61
  using CMS superelements, 43