

# Multibody Dynamics Module

User's Guide

# Multibody Dynamics Module User's Guide

© 1998–2020 COMSOL

Protected by patents listed on [www.comsol.com/patents](http://www.comsol.com/patents), and U.S. Patents 7,519,518; 7,596,474; 7,623,991; 8,457,932; 9,098,106; 9,146,652; 9,323,503; 9,372,673; 9,454,625; 10,019,544; 10,650,177; and 10,776,541. Patents pending.

This Documentation and the Programs described herein are furnished under the COMSOL Software License Agreement ([www.comsol.com/comsol-license-agreement](http://www.comsol.com/comsol-license-agreement)) and may be used or copied only under the terms of the license agreement.

COMSOL, the COMSOL logo, COMSOL Multiphysics, COMSOL Desktop, COMSOL Compiler, COMSOL Server, and LiveLink are either registered trademarks or trademarks of COMSOL AB. All other trademarks are the property of their respective owners, and COMSOL AB and its subsidiaries and products are not affiliated with, endorsed by, sponsored by, or supported by those trademark owners. For a list of such trademark owners, see [www.comsol.com/trademarks](http://www.comsol.com/trademarks).

Version: COMSOL 5.6

## Contact Information

Visit the Contact COMSOL page at [www.comsol.com/contact](http://www.comsol.com/contact) to submit general inquiries, contact Technical Support, or search for an address and phone number. You can also visit the Worldwide Sales Offices page at [www.comsol.com/contact/offices](http://www.comsol.com/contact/offices) for address and contact information.

If you need to contact Support, an online request form is located at the COMSOL Access page at [www.comsol.com/support/case](http://www.comsol.com/support/case). Other useful links include:

- Support Center: [www.comsol.com/support](http://www.comsol.com/support)
- Product Download: [www.comsol.com/product-download](http://www.comsol.com/product-download)
- Product Updates: [www.comsol.com/support/updates](http://www.comsol.com/support/updates)
- COMSOL Blog: [www.comsol.com/blogs](http://www.comsol.com/blogs)
- Discussion Forum: [www.comsol.com/community](http://www.comsol.com/community)
- Events: [www.comsol.com/events](http://www.comsol.com/events)
- COMSOL Video Gallery: [www.comsol.com/video](http://www.comsol.com/video)
- Support Knowledge Base: [www.comsol.com/support/knowledgebase](http://www.comsol.com/support/knowledgebase)

Part number: CM023801

## Contents

### Chapter 1: Introduction

<b>About the Multibody Dynamics Module</b>	<b>12</b>
What Can the Multibody Dynamics Module Do? . . . . .	12
Where Do I Access the Documentation and Application Libraries? . . . . .	13
<b>Overview of the User's Guide</b>	<b>18</b>

### Chapter 2: Multibody Dynamics Modeling

<b>The Multibody Dynamics Model</b>	<b>22</b>
Rigid and Flexible Parts . . . . .	22
Meshing . . . . .	22
Connecting Parts . . . . .	23
Gears . . . . .	23
Connecting to Other Physics Interfaces . . . . .	24
<b>Joints</b>	<b>25</b>
About Joints . . . . .	25
Prismatic Joint . . . . .	36
Hinge Joint . . . . .	37
Cylindrical Joint . . . . .	37
Screw Joint . . . . .	39
Planar Joint . . . . .	40
Ball Joint . . . . .	41
Slot Joint . . . . .	41
Reduced Slot Joint . . . . .	42
Clearance Joint . . . . .	43
Fixed Joint . . . . .	43
Distance Joint . . . . .	44
Universal Joint. . . . .	44

<b>Attachments</b>	<b>46</b>
<b>Cam-Follower</b>	<b>49</b>
<b>Springs and Dampers</b>	<b>50</b>
<b>Gears</b>	<b>51</b>
About Gears . . . . .	51
About Gear Pairs. . . . .	53
<b>Gear Geometries</b>	<b>57</b>
Gear Tooth Geometry (External: 2D) . . . . .	58
Gear Tooth Geometry (Internal: 2D) . . . . .	58
Rack Tooth Geometry (2D) . . . . .	59
Spur Gear Geometry (External: 2D) . . . . .	60
Spur Gear Geometry (Internal: 2D). . . . .	60
Spur Rack Geometry (2D) . . . . .	61
Spur Gear Geometry (External: 3D) . . . . .	62
Spur Gear Geometry (Internal: 3D). . . . .	62
Helical Gear Geometry (External: 3D). . . . .	63
Helical Gear Geometry (Internal: 3D) . . . . .	64
Bevel Gear Geometry (3D) . . . . .	65
Worm Gear Geometry (3D) . . . . .	66
Spur Rack Geometry (3D) . . . . .	67
Helical Rack Geometry (3D) . . . . .	68
<b>Chain Drive</b>	<b>70</b>
About Chain Drive . . . . .	70
<b>Chain Geometries</b>	<b>78</b>
Components of a Roller Chain Sprocket Assembly . . . . .	78
Geometries of Roller Chain Sprocket Assembly . . . . .	80
Roller Chain Geometry (2D) . . . . .	80
Sprocket Geometry (2D). . . . .	82
Roller Chain Sprocket Assembly Geometry (2D). . . . .	83
Roller Chain Geometry (3D) . . . . .	86
Sprocket Geometry (3D). . . . .	87
Roller Chain Sprocket Assembly Geometry (3D). . . . .	88

<b>Bearings</b>	<b>92</b>
Bearing Modeling. . . . .	92
<b>Initialization in Time-Dependent Studies</b>	<b>94</b>
<b>Geometric Nonlinearity</b>	<b>95</b>
<b>Multiphysics Couplings</b>	<b>96</b>
The Fluid-Multibody Interaction Interface . . . . .	97
The Fluid-Multibody Interaction, Assembly Interface . . . . .	99

## Chapter 3: Multibody Dynamics Theory

<b>Joint Theory</b>	<b>104</b>
Representation of Rotations . . . . .	104
Prismatic Joint Theory. . . . .	106
Hinge Joint Theory . . . . .	110
Cylindrical Joint Theory . . . . .	112
Screw Joint Theory . . . . .	113
Planar Joint Theory . . . . .	114
Ball Joint Theory . . . . .	115
Slot Joint Theory. . . . .	116
Reduced Slot Joint Theory . . . . .	118
Clearance Joint Theory . . . . .	119
Fixed Joint Theory . . . . .	121
Distance Joint Theory . . . . .	121
Universal Joint Theory. . . . .	122
Theory for Joint Constraints . . . . .	125
Theory for Joint Locking . . . . .	131
Theory for Springs and Dampers on Joints . . . . .	134
Theory for Prescribed Motion on Joints . . . . .	136
Theory for Applied Forces and Moments on Joints . . . . .	138
Theory for Friction in Joints. . . . .	139
Theory for Joint Forces and Moments Computation . . . . .	146

<b>Cam-Follower Theory</b>	<b>149</b>
Cam . . . . .	149
Follower . . . . .	149
Closest Point Determination . . . . .	149
Cam with Boundary Selection . . . . .	150
Cam with Edge Selection . . . . .	151
Cam-Follower Connection Force Computation . . . . .	152
<b>Gear Theory</b>	<b>154</b>
Spur Gear Theory . . . . .	154
Helical Gear Theory . . . . .	156
Bevel Gear Theory . . . . .	158
Worm Gear Theory . . . . .	159
Spur Rack Theory . . . . .	160
Helical Rack Theory. . . . .	162
<b>Gear Pair Theory</b>	<b>163</b>
Theory for Gear Pairs . . . . .	163
Theory for Worm and Wheel. . . . .	171
Theory for Rack and Pinion. . . . .	173
Theory for Gear Elasticity . . . . .	176
Theory for Transmission Error . . . . .	183
Theory for Backlash. . . . .	184
Theory for Gear Friction. . . . .	186
<b>Rigid Body Contact Theory</b>	<b>189</b>
Theory of Rigid Body Contact. . . . .	189
Spherical to Spherical Formulation . . . . .	189
Spherical to Arbitrary Formulation . . . . .	192
Theory of Friction in Rigid Body Contact. . . . .	194
 <b>Chapter 4: Multibody Dynamics</b>	
<b>The Multibody Dynamics Interface</b>	<b>198</b>
Domain, Boundary, Edge, Point, and Pair Nodes for the Multibody	
Dynamics Interface. . . . .	204

Initial Values . . . . .	208
Initially Rigid . . . . .	210
Attachment . . . . .	211
Rigid Domain . . . . .	213
Center of Rotation Nodes . . . . .	215
Initial Values (Rigid Domain) . . . . .	216
Prismatic Joint . . . . .	217
Hinge Joint . . . . .	219
Cylindrical Joint . . . . .	220
Screw Joint . . . . .	221
Planar Joint . . . . .	222
Ball Joint . . . . .	222
Slot Joint . . . . .	223
Reduced Slot Joint . . . . .	224
Clearance Joint . . . . .	225
Fixed Joint . . . . .	227
Distance Joint . . . . .	227
Universal Joint . . . . .	229
Joint Elasticity . . . . .	231
Constraints . . . . .	232
Locking . . . . .	235
Spring and Damper . . . . .	237
Prescribed Motion . . . . .	240
Applied Force and Moment . . . . .	242
Friction . . . . .	243
Contact Area . . . . .	246
Joint Axis . . . . .	246
Input Shaft Orientation . . . . .	246
Output Shaft Orientation . . . . .	247
Center of Joint Nodes . . . . .	247
Source Point Nodes . . . . .	247
Destination Point Nodes . . . . .	248
Source Filter . . . . .	249
Destination Filter . . . . .	249
Base Motion . . . . .	250
Rigid Body Contact . . . . .	251
Friction (Rigid Body Contact) . . . . .	254
Cam-Follower . . . . .	255

Spur Gear . . . . .	258
Helical Gear. . . . .	261
Bevel Gear . . . . .	262
Worm Gear . . . . .	263
Spur Rack . . . . .	264
Helical Rack. . . . .	265
Gear Axis . . . . .	266
Rack Axis. . . . .	266
Gear Pair. . . . .	267
Worm and Wheel . . . . .	269
Rack and Pinion . . . . .	270
Gear Elasticity . . . . .	271
Transmission Error . . . . .	273
Backlash . . . . .	275
Friction (Gear). . . . .	276
Chain Drive. . . . .	277
Sprocket Axis . . . . .	283
Radial Roller Bearing . . . . .	283
Flexible Foundation . . . . .	286
Moving Foundation . . . . .	286

## Chapter 5: Lumped Mechanical System

<b>The Lumped Mechanical System Interface</b>	<b>290</b>
Nodes for the Lumped Mechanical System Interface . . . . .	291
Mass . . . . .	292
Spring . . . . .	293
Damper . . . . .	294
Impedance . . . . .	295
Displacement Source . . . . .	296
Force Source . . . . .	297
External Source . . . . .	298
Subsystem Definition . . . . .	299
Subsystem Instance . . . . .	300
Fixed Node . . . . .	301
Free Node . . . . .	302

Displacement Node. . . . .	302
Velocity Node. . . . .	303
Acceleration Node . . . . .	304
Force Node. . . . .	305
Impedance Node. . . . .	306
Lumped-Structure Connection . . . . .	306
<b>Theory for the Lumped Mechanical System Interface</b>	<b>309</b>
Dependent Variables . . . . .	309
Theory for Two-Port Components. . . . .	309
Theory for Single-Port Components . . . . .	315
<b>Index</b>	<b>321</b>



# Introduction

This guide describes the Multibody Dynamics Module, an optional add-on package for the COMSOL Multiphysics® simulation software designed to perform static or dynamic analysis for a system of flexible and rigid components connected through a set of joints, springs, dampers, gears, chains, and cams.

This chapter introduces you to the capabilities of this module. A summary of the physics interfaces and where you can find documentation and model examples is also included. The last section is a brief overview with links to the chapters in this guide.

In this chapter:

- [About the Multibody Dynamics Module](#)
- [Overview of the User's Guide](#)

# About the Multibody Dynamics Module

In this section:

- [What Can the Multibody Dynamics Module Do?](#)
- [Where Do I Access the Documentation and Application Libraries?](#)

## *What Can the Multibody Dynamics Module Do?*

---

The Multibody Dynamics Module can be used to model the static and dynamic behavior of rigid or flexible components connected using joints with certain degrees of freedom. The components can then undergo complex combinations of translational and rotational motions with respect to one another. Some major application areas are for automotive applications, aerospace engineering, biomechanics, biomedical instruments, robotics, and general dynamic simulations.

The main components of this module are [The Multibody Dynamics Interface](#) and [The Lumped Mechanical System Interface](#).

The Multibody Dynamics interface is used to model assemblies of flexible components, rigid components, or the combination of both. Flexible components can be defined through solid, shell, or beam elements. They can also have geometric and material nonlinearity. With this physics interface, you can simultaneously perform a system analysis and a detailed component analysis of a mechanical system.

The features that are used to model connections are joints, attachments, rigid domains, springs and dampers. The joints that connect the different structural parts can have many different properties, including elastic connections, viscous damping, friction losses, stop conditions, and lock conditions.

There are several types of joints available for modeling different types of connections between components. Attachments are used by joints to establish a connection primarily between flexible components, but can also be used with rigid bodies. A rigid domain models a rigid component, and it is directly used in a joint to establish its connection with another rigid or a flexible component. Elastic springs and viscous dampers can also be used to connect rigid or flexible components.

This interface is also used to model geared systems for power transmission. There are several types of gears available to connect and transfer power from one shaft to another. The gear pairs that connect two gears can have many different properties, including mesh stiffness and damping, static transmission error, backlash, and friction losses. Similarly chain drives are also possible to model for power transmission. The roller chain and sprocket assembly is available to connect and transfer power from one shaft to another. The roller chain sprocket assembly can have different parts rigid or elastic and based on that different contact methods between chain and sprocket, roller center based and mesh based, are available.

The results obtained from the gear or chain dynamics can be further used for the fatigue analysis of the gear/sprocket or for the acoustic analysis to find out the noise emitted by the system.

This interface has a dedicated feature for modeling frictional contact between rigid bodies. The rigid body contact functionality is primarily for modeling the contact between standard rigid shaped bodies. It is much robust and faster compared to a standard mesh-based contact.

The Multibody Dynamics interface can be coupled to a heat transfer, acoustics, or fluid flow interface through multiphysics coupling nodes.

The Lumped Mechanical System interface is intended to model displacements and forces in a mechanical system using a circuit representation with lumped components such as masses, springs, and dampers. In addition to the displacement and force sources in the system, terminal nodes are used to define forces, displacements, velocities, accelerations, or impedances as boundary conditions.

---

In the *COMSOL Multiphysics Reference Manual*:

- [Studies and Solvers](#)
- [The Physics Interfaces](#)
- For a list of all the core physics interfaces included with a COMSOL Multiphysics license, see [Physics Interface Guide](#).

---

### *Where Do I Access the Documentation and Application Libraries?*

---

A number of internet resources have more information about COMSOL, including licensing and technical information. The electronic documentation, topic-based (or

context-based) help, and the application libraries are all accessed through the COMSOL Desktop.



If you are reading the documentation as a PDF file on your computer, the [blue links](#) do not work to open an application or content referenced in a different guide. However, if you are using the Help system in COMSOL Multiphysics, these links work to open other modules, application examples, and documentation sets.

### THE DOCUMENTATION AND ONLINE HELP

The *COMSOL Multiphysics Reference Manual* describes the core physics interfaces and functionality included with the COMSOL Multiphysics license. This book also has instructions about how to use COMSOL Multiphysics and how to access the electronic Documentation and Help content.

#### Opening Topic-Based Help

The Help window is useful as it is connected to the features in the COMSOL Desktop. To learn more about a node in the Model Builder, or a window on the Desktop, click to highlight a node or window, then press F1 to open the Help window, which then displays information about that feature (or click a node in the Model Builder followed by the **Help** button (  ). This is called *topic-based* (or *context*) *help*.

To open the **Help** window:

- In the **Model Builder**, **Application Builder**, or **Physics Builder** click a node or window and then press F1.
- On any toolbar (for example, **Home**, **Definitions**, or **Geometry**), hover the mouse over a button (for example, **Add Physics** or **Build All**) and then press F1.
- From the **File** menu, click **Help** (  ).
- In the upper-right corner of the COMSOL Desktop, click the **Help** (  ) button.



---

	To open the <b>Help</b> window:
	<ul style="list-style-type: none"><li>• In the <b>Model Builder</b> or <b>Physics Builder</b> click a node or window and then press F1.</li></ul>
	<ul style="list-style-type: none"><li>• On the main toolbar, click the <b>Help</b> (  ) button.</li><li>• From the main menu, select <b>Help&gt;Help</b>.</li></ul>

---

### *Opening the Documentation Window*

---

	To open the <b>Documentation</b> window:
	<ul style="list-style-type: none"><li>• Press Ctrl+F1.</li><li>• From the <b>File</b> menu select <b>Help&gt;Documentation</b> (  ).</li></ul>

---

---

	To open the <b>Documentation</b> window:
	<ul style="list-style-type: none"><li>• Press Ctrl+F1.</li></ul>
	<ul style="list-style-type: none"><li>• On the main toolbar, click the <b>Documentation</b> (  ) button.</li><li>• From the main menu, select <b>Help&gt;Documentation</b>.</li></ul>

---

### **THE APPLICATION LIBRARIES WINDOW**

Each model or application includes documentation with the theoretical background and step-by-step instructions to create a model or application. The models and applications are available in COMSOL Multiphysics as MPH files that you can open for further investigation. You can use the step-by-step instructions and the actual models as templates for your own modeling. In most models, SI units are used to describe the relevant properties, parameters, and dimensions, but other unit systems are available.

Once the Application Libraries window is opened, you can search by name or browse under a module folder name. Click to view a summary of the model or application and its properties, including options to open it or its associated PDF document.

---

	<a href="#">The Application Libraries Window</a> in the <i>COMSOL Multiphysics Reference Manual</i> .
---	---

---

## *Opening the Application Libraries Window*

To open the **Application Libraries** window (  ):

---

- From the **Home** toolbar, **Windows** menu, click (  ) **Applications Libraries**.

 Win

- From the **File** menu select **Application Libraries**.

To include the latest versions of model examples, from the **File>Help** menu, select (  ) **Update COMSOL Application Library**.

---

 Mac

Select **Application Libraries** from the main **File>** or **Windows>** menus.

 Linux

To include the latest versions of model examples, from the **Help** menu select (  ) **Update COMSOL Application Library**.

---

## **CONTACTING COMSOL BY EMAIL**

For general product information, contact COMSOL at [info@comsol.com](mailto:info@comsol.com).

## **COMSOL ACCESS AND TECHNICAL SUPPORT**

To receive technical support from COMSOL for the COMSOL products, please contact your local COMSOL representative or send your questions to [support@comsol.com](mailto:support@comsol.com). An automatic notification and a case number are sent to you by email. You can also access technical support, software updates, license information, and other resources by registering for a COMSOL Access account.

## COMSOL ONLINE RESOURCES

COMSOL website	<a href="http://www.comsol.com">www.comsol.com</a>
Contact COMSOL	<a href="http://www.comsol.com/contact">www.comsol.com/contact</a>
COMSOL Access	<a href="http://www.comsol.com/access">www.comsol.com/access</a>
Support Center	<a href="http://www.comsol.com/support">www.comsol.com/support</a>
Product Download	<a href="http://www.comsol.com/product-download">www.comsol.com/product-download</a>
Product Updates	<a href="http://www.comsol.com/support/updates">www.comsol.com/support/updates</a>
COMSOL Blog	<a href="http://www.comsol.com/blogs">www.comsol.com/blogs</a>
Discussion Forum	<a href="http://www.comsol.com/community">www.comsol.com/community</a>
Events	<a href="http://www.comsol.com/events">www.comsol.com/events</a>
COMSOL Application Gallery	<a href="http://www.comsol.com/models">www.comsol.com/models</a>
COMSOL Video Gallery	<a href="http://www.comsol.com/video">www.comsol.com/video</a>
Support Knowledge Base	<a href="http://www.comsol.com/support/knowledgebase">www.comsol.com/support/knowledgebase</a>

# Overview of the User's Guide

The *Multibody Dynamics Module User's Guide* gets you started with modeling using COMSOL Multiphysics. The information in this guide is specific to this module. Instructions how to use COMSOL Multiphysics in general are included with the *COMSOL Multiphysics Reference Manual*.



As detailed in the section [Where Do I Access the Documentation and Application Libraries?](#) this information can also be searched from the COMSOL Multiphysics software **Help** menu.

## TABLE OF CONTENTS AND INDEX

To help you navigate through this guide, see the [Contents](#) and [Index](#).

## MULTIBODY DYNAMICS MODELING

The [Multibody Dynamics Modeling](#) chapter outlines the modeling guidelines for the Multibody Dynamics interface. The [Multibody Dynamics Model](#) section describes various elements of a multibody system. The [Joints](#) section describes the types of joints and the additional functionality available. The [Attachments](#) section describes the use and formulation of attachment features. The [Cam-Follower](#) section describes the formulation of cam-follower connection. The [Springs and Dampers](#) section describes the formulation of an elastic spring and a viscous damper. The [Gears](#) section describes the type of gears, gear pairs, and the additional functionality available for modeling of gears. The [Gear Geometries](#) section describes the parameterized gear geometries available. The [Chain Drive](#) section describes the modeling of roller chain sprocket assembly. The [Chain Geometries](#) section describes the parameterized chain-sprocket geometries available. The section on [Initialization in Time-Dependent Studies](#) describes the consistent initialization in a multibody system. The [Geometric Nonlinearity](#) section talks about the cases when a geometric nonlinear analysis is required. In the end, [Multiphysics Couplings](#) section describes all the built-in multiphysics coupling features available for multibody dynamics.

## THE THEORY OF THE MULTIBODY DYNAMICS INTERFACE

The [Multibody Dynamics Theory](#) chapter outlines the theory for the Multibody Dynamics interface. The [Joint Theory](#) section describes the types of joints and the additional functionality available. The [Cam-Follower Theory](#) section describes the

theory for the cam and follower connection. The [Gear Theory](#) and [Gear Pair Theory](#) sections describe the theory of all the gear types and gear pair types respectively. In the end, the [Rigid Body Contact Theory](#) section describes the theory of modeling rigid body contact and friction between standard shaped rigid bodies.

### **THE MULTIBODY DYNAMICS INTERFACE**

The [Multibody Dynamics](#) chapter describes all the physics nodes available in [The Multibody Dynamics Interface](#). Joints, spring-dampers, cam-followers, attachments, rigid domains, base motions, initial values, gears, gear pairs, chain drives, and rigid body contacts are described in detail. The rest of the physics nodes are described in the *Structural Mechanics Module User's Guide*.

### **LUMPED MECHANICAL SYSTEM**

The [Lumped Mechanical System](#) chapter describes all the physics nodes available in [The Lumped Mechanical System Interface](#). The corresponding theory for all the physics nodes is described in the [Theory for the Lumped Mechanical System Interface](#) section.



# Multibody Dynamics Modeling

This chapter introduces you to the theory for the Multibody Dynamics interface features.

In this chapter:

- [The Multibody Dynamics Model](#)
- [Joints](#)
- [Attachments](#)
- [Cam-Follower](#)
- [Springs and Dampers](#)
- [Gears](#)
- [Gear Geometries](#)
- [Chain Drive](#)
- [Chain Geometries](#)
- [Bearings](#)
- [Initialization in Time-Dependent Studies](#)
- [Geometric Nonlinearity](#)
- [Multiphysics Couplings](#)

# The Multibody Dynamics Model

## *Rigid and Flexible Parts*

---

One of the big advantages of the Multibody Dynamics Module is the ease by which you can mix rigid and flexible parts. Usually, all or the majority of the parts in a multibody simulation are rigid and are thus represented by only the degrees of freedom of a rigid body. The material model for a rigid part is **Rigid Domain**.

Sometimes, however, you may want to include the flexibility of one or several domains. To do that, use a **Linear Elastic Material**. This is the default material model in the Multibody Dynamics interface, so the workflow is to add as many **Rigid Domain** nodes as needed and assign domains to them. The remaining domains will then be linear elastic and thus flexible. By adding a Solid Mechanics interface, you can get access to many other material models for the flexible domains.

## *Meshing*

---

A rigid domain is only represented by a set of ODEs, representing the equations of motion for a rigid body. The mesh on a rigid domain is only used for integrating properties like mass, center of gravity, and moments of inertia. This means that the mesh can be much coarser than what is usually the case in a finite element model. Since there is still a computational and memory consumption cost involved with each element, you should strive for a very coarse mesh in rigid domains.

When meshing a flexible body, you have to make the same considerations as when meshing any other finite element model.



In the Multibody Dynamics interface, the default shape function order for the flexible parts is linear, as opposed to the Solid Mechanics interface where it is quadratic. The purpose is to keep down the number of degrees of freedom in those cases where the stiffness is more important than stress evaluation.

If you need more accurate stress results, you can either use a finer mesh, or change the discretization to **Quadratic serendipity** or **Quadratic** in the **Discretization** section settings of the interface.

## Connecting Parts

---

The most important feature for connecting part is a *joint*. A joint will in general allow relative movement between two parts in one or more degrees of freedom. This is described in detail in the [Joints](#) section.

It is also possible to connect parts by springs and dampers, as described in the [Springs and Dampers](#) section.

A part can be connected to several joints, springs, or dampers. The way to attach them differs between a rigid and a flexible object.

- A flexible body must be connected through an *attachment*, as described in the [Attachments](#) section. Add an **Attachment** node to the model and select the attached boundaries. The attachment is then referenced in, for example, a joint.
- A rigid domain can be connected directly, by just specifying the location of the connection. The connection point does not even have to physically inside the rigid domain since the domain can be considered as an abstract object.
- A rigid domain can also be connected by an attachment. This is convenient if you consider switching between a rigid and a flexible formulation for this domain.



A Rigid Domain can consist of a selection of several geometrical domains. These domains will act as a single rigid object, irrespective of whether they are geometrically connected or not.

Adjacent domains selected in different Rigid Domains nodes are independent objects, and can even penetrate each other.

## Gears

---

A gear is defined as a rigid body. A finite stiffness for the gear mesh (or gear tooth) can be specified in the gear pair node while connecting to other gears. Similar to the rigid domain, the gear is a material model, which is mutually exclusive to all other material models. The only material property needed is the mass density.

## *Connecting to Other Physics Interfaces*

---

When modeling flexible parts, it is possible to add other structural mechanics interfaces to get additional functionality:

- If you have the Structural Mechanics Module, the Solid Mechanics interface has several features which are not available directly from the Multibody Dynamics interface.
- If you have the Nonlinear Structural Mechanics Module or the Geomechanics Module, then you can get access to the whole range of material model by adding a Solid Mechanics interface. In this way, you can for example include a hyperelastic or elastoplastic domain in your multibody dynamics simulation.
- If you have the Structural Mechanics Module, you can add a Shell or Beam interface to model thin or slender flexible objects.

There are two ways in which you can connect to objects from the other structural mechanics interfaces:

- You can connect through joints or springs. In the Solid Mechanics, Shell, or Beam interfaces, you can add **Attachment** nodes. Such attachments will then appear in the lists of attachments in various features in the Multibody Dynamics interface.
- You can also connect to objects in the Shell and Beam interfaces through the **Solid-Shell Connection** and **Solid-Beam Connection** multiphysics couplings.

Using the Lumped Mechanical System interface, you can model circuit representation of mechanical systems of masses, springs, and dampers. With the **Lumped-Structure Connection** multiphysics coupling, you can connect the Lumped Mechanical System Interface to the Multibody Dynamics interface.

# Joins

## *About Joints*

---

A joint is used to allow a certain type of relative motion between the two components it connects. A joint is a global feature that needs two *attachments*, one from each component. One attachment acts as a *source* and the other as a *destination*.



The two attachments belonging to a joint can be [Attachment](#) features, [Rigid Domain](#) features, or a combination of the two. The source attachment can also be set to be either *fixed* or to have a given **Base motion**. In the case of a fixed attachment, it cannot translate or rotate. This avoids the modeling of extra fixed rigid domains when a structure is “grounded”. The base motion is a generalization of the fixed attachment, where the environment to which it is attached can have a prescribed displacement, velocity, or acceleration.

The source attachment moves independently in space, whereas the destination attachment is constrained to follow the source attachment allowing for some relative degrees of freedom in the form of translation or rotation. The relative degrees of freedom are determined by the type of joint. In each joint, the relative degrees of freedom are represented by ODE variables.

The degrees of freedom that are constrained in a joint can be rigidly constrained or contain some relative elastic displacement.

The rest of this section has these topics:

- Joint Types and Degrees of Freedom
- Selecting Attachments
- Joint Axis and Local Coordinate System
- Positioning the Joint Center
- Constraints
- Locking
- Springs and Dampers on Joints
- Elasticity in Joints
- Prescribed Motion
- Applied Forces and Moments on Joints
- Friction
- Computing Joint Forces and Moments
- Initial Values of Joint Degrees of Freedom
- Creating Joints Automatically

### JOINT TYPES AND DEGREES OF FREEDOM

Table 2-1 lists the joint types for 3D models based on the possible relative degrees of freedom.

TABLE 2-1: JOINT TYPES AND DEGREES OF FREEDOM IN 3D

JOINT TYPE	TOTAL DEGREES OF FREEDOM	TRANSLATIONAL DEGREES OF FREEDOM	ROTATIONAL DEGREES OF FREEDOM
Prismatic Joint	1	1	0
Hinge Joint	1	0	1
Cylindrical Joint	2	1	1
Screw Joint*	1*	1	1
Planar Joint	3	2	1
Ball Joint	3	0	3
Slot Joint	4	1	3
Reduced Slot Joint	2	1	1
Clearance Joint**	3	0**	3
Fixed Joint	0	0	0
Distance Joint***	5	2***	3

TABLE 2-1: JOINT TYPES AND DEGREES OF FREEDOM IN 3D

JOINT TYPE	TOTAL DEGREES OF FREEDOM	TRANSLATIONAL DEGREES OF FREEDOM	ROTATIONAL DEGREES OF FREEDOM
Universal Joint	2	0	2

\* For a Screw Joint, the two degrees of freedom are linearly dependent.

\*\* For a Clearance Joint, all three translational degrees of freedom are not completely restrained but allowed with a limiting value.

\*\* For a Distance Joint, all three translational degrees for freedom are not independent.

Table 2-2 lists the available joint features for 2D models:

TABLE 2-2: JOINT TYPES AND DEGREES OF FREEDOM IN 2D

JOINT TYPE	TOTAL DEGREES OF FREEDOM	TRANSLATIONAL DEGREES OF FREEDOM	ROTATIONAL DEGREES OF FREEDOM
Prismatic Joint	1	1	0
Hinge Joint	1	0	1
Reduced Slot Joint	2	1	1
Clearance Joint*	1	0*	1
Fixed Joint	0	0	0
Distance Joint**	2	1**	1

\* For a Clearance Joint, both translational degrees of freedom are not completely restrained but allowed with a limiting value.

\*\* For a Distance Joint, both translational degrees for freedom are not independent.

### SELECTING ATTACHMENTS

There are no rules for how to select the source or destination attachment for a joint, but one selection can be more convenient than the other. The only difference is that the interpretation of certain quantities changes if the source and destination are swapped. Some of the important quantities and the relationship with the source and destination attachments are:

- The joint coordinate system follows the source attachment.
- All the relative degrees of freedom are created on the destination attachment.
- The joint moments are computed around the center of joint attached to the source attachment.

- In the [Ball Joint](#) and [Slot Joint](#), the inclination angle is the angle between the reference axis and the destination axis. The axial rotation occurs about the destination axis. This reference axis is defined on the source attachment.
- In the [Planar Joint](#), all axes in its subnodes follow the source attachment.

### POSITIONING THE JOINT CENTER

You can position the center of a joint in different ways:

- At the centroid of the source attachment.
- At the centroid of the destination attachment.
- At the centroid of some selected entities (boundaries, edges, or points). As a special case a single point can be used.
- Explicitly given coordinates.



Attachments must be used for flexible components, and are optional for rigid components. The first two options are only available if an attachment is selected as source or destination.

### JOINT AXIS AND LOCAL COORDINATE SYSTEM

The local coordinate system of a joint, which can be seen as rigidly connected to the source attachment, consists of three axes: joint axis ( $e_1$ ), second axis ( $e_2$ ), and third axis ( $e_3$ ). You specify the initial direction joint axis ( $e_{10}$ ), using one of the following methods:

- **Method 1:** Give the components explicitly in a selected coordinate system.
- **Method 2:** Select one of the axes of a selected coordinate system.
- **Method 3:** Select an edge that is parallel to the joint axis.

For all methods, the axis is normalized to give a unit vector.

### CONSTRAINTS

[Constraints](#) are used to restrict the relative motion between the two components sharing a joint. You can specify the upper and lower bound of the relative motion. Applying constraints on the relative motion can be interpreted as placing an imaginary

stopper that restricts the motion of the destination attachment with respect to the source attachment in a prescribed direction.

---



### Theory for Joint Constraints

---

#### LOCKING

The **Locking** feature is used to lock the relative motion between the two components connected to a joint. You specify limits on the relative motion in the form of an upper and lower bound. This can be interpreted as an imaginary object (like a snap hook) that locks one of the free relative degrees of freedom of the destination attachment when it reaches the limiting value.

The Locking feature differs from **Constraints** in that the value of the particular joint degree of freedom becomes constant for the rest of the analysis after it attains its prescribed limiting value.

---



### Theory for Joint Locking

---

#### SPRINGS AND DAMPERS ON JOINTS

You can use the **Spring and Damper** feature to apply a spring or a dashpot to act on the relative motion in a joint. A dashpot can be used to account for various kinds of losses at the joint. The spring can be given a predeformation so that the spring equilibrium position is not the same as the initial state.

---



### Theory for Springs and Dampers on Joints

---

#### ELASTICITY IN JOINTS

In the directions in which relative motion is not allowed in a joint, the default case is that the source and destination attachments are rigidly connected. It is possible to instead insert an elastic connection by setting **Joint Elasticity** to **Elastic joint** in the settings for the joint. There are several occasions when you may want to use this option:

- There is a physical flexibility in the joint, like a bushing.

- The mechanical system is overconstrained by the chosen set of joints. You can then release some degrees of freedom by making them elastic. This should typically be done in directions where no significant forces act.
- You have a mechanism which is not possible to model by the built-in joints. You can then add degrees of freedom to the built-in joint by making them elastic, and not supply any stiffness.
- The joint gets deactivated and the two bodies are no longer connected after certain time in a transient simulation. You then can use the elastic joints for this purpose by making the stiffness and damping properties a function of time.

In addition to the elasticity, you can also add viscous damping to a degree of freedom selected as being elastic.

---

The theory for joint elasticity is described separately for each joint type:

- [Joint Elasticity in Prismatic Joint](#)
- [Joint Elasticity in Hinge Joint](#)
- [Joint Elasticity in Cylindrical Joint](#)
- [Joint Elasticity in Planar Joint](#)
- [Joint Elasticity in Ball Joint](#)
- [Joint Elasticity in Slot Joint](#)
- [Joint Elasticity in Reduced Slot Joint](#)
- [Joint Elasticity in Fixed Joint](#)
- [Joint Elasticity in Universal Joint](#)

---

#### **P R E S C R I B E D   M O T I O N**

With [Prescribed Motion](#) you can control the relative motion in a joint, either by prescribing the displacement or the velocity. This is useful when the degrees of



freedom in the joint are not free but are a known function of time. This is common in the field of robotics, for example.

---

Evaluating reaction force for the prescribed motion by selecting **Evaluate reaction forces** in the **Reaction force settings** section of the settings for **Prescribed Motion** has several benefits over evaluating it using joint forces and moments computed using weak constraints:

- It includes the contributions of springs, dampers, and friction applied on the joint.
- It works even if there are redundant constraints in the system, especially in a rigid body system, because joint constraints cannot be applied in the weak form in such a system.
- It is computationally more efficient as it adds fewer degrees of freedom.

---



Theory for Prescribed Motion on Joints

---

#### APPLIED FORCES AND MOMENTS ON JOINTS

The **Applied Force and Moment** feature is used to apply forces and moments on all types of joints. It can be applied at the source or destination joint attachment, or directly to the joint degrees of freedom.

---



Theory for Applied Forces and Moments on Joints

---

#### FRICITION

The **Friction** feature is used to add frictional loss to a joint. It is available for **Prismatic Joint**, **Hinge Joint**, **Cylindrical Joint**, **Screw Joint**, **Planar Joint**, and **Ball Joint**.

The friction force is modeled using a continuous friction law, which is capable of modeling sliding-sticking phenomena. A strict application of Coulomb's law involves discrete transition from sticking to sliding and vice versa, as dictated by a vanishing relative velocity. These discrete transitions cause numerical difficulties and to avoid them, the friction force is approximated with a continuous friction law:

$$F_f = -\mu N \frac{v}{|v|} \left( 1 - e^{-\frac{|v|}{v_0}} \right)$$

where  $F_f$  is the friction force,  $\mu$  is the frictional coefficient,  $N$  is the normal force,  $v$  is the slip velocity, and  $v_0$  is the characteristic slip velocity. The term

$$\left( 1 - e^{-\frac{|v|}{v_0}} \right)$$

is called the *regularization factor*.

The regularization factor is used to smoothen the friction force discontinuity. The characteristic slip velocity should be chosen to be small in comparison to the characteristic relative velocities encountered during the simulation. The continuous friction law describes both sliding and sticking behavior; that is, it completely replaces Coulomb's law. Sticking is replaced by creeping between the contacting bodies with a small relative velocity.



## Theory for Friction in Joints

### COMPUTING JOINT FORCES AND MOMENTS

It is often of interest to evaluate the internal forces (the reaction forces) in a joint. There are three ways to compute forces and moments — [Summing Reaction Forces Over the Boundaries](#), or [Using Weak Constraints](#), or [Using Penalty Formulation](#).

The joint forces and moments are evaluated in the global spatial coordinate system as well as in a joint local coordinate system. The joint forces and moments are referred to the center of joint in the current position. For the joints that have translational degrees of freedom, the center of joint should be interpreted as fixed to the source side.

When a joint connects two rigid bodies, the default is not to compute the joint forces. The reason is that it is common that rigid body systems are overconstrained, and then numerical problems can occur.

#### *Summing Reaction Forces Over the Boundaries*

This is the default method for computing the joint forces and moments for elastic objects. The computation can be done on either the source or destination attachment. The convention is that the force and moment are the ones acting on the attachment

from the joint. The sign changes if you switch between using the source or destination attachment for the evaluation.



To use this method at least one of the components connected to the joint must be flexible.

#### *Using Weak Constraints*

When using weak constraints, the joint constraints are applied in a weak form, and the values of the Lagrange multipliers give the joint forces and moments. This works for both flexible and rigid components. The use of Lagrange multipliers can, however, have an effect on the structure of the equation system, which limits the solvers that can be used.

The sign convention in this case is that the forces and moments should be interpreted as acting on the destination attachment.



Using the weak constraint option in an overconstrained rigid body system often leads to numerical difficulties.

#### *Using Penalty Formulation*

When using the penalty formulation, the constraints in the joint are implemented as stiff springs. You have to provide a *penalty factor*, which essentially is the spring constant. The penalty method will thus not fulfill the constraints exactly, but it has the advantage of making the model less sensitive to overconstraints.



#### **Theory for Joint Forces and Moments Computation**

##### **INITIAL VALUES OF JOINT DEGREES OF FREEDOM**

You do not need to supply separate initial values for joint degrees of freedom. They are consistently initialized from the degrees of freedom for a material model, either a

displacement field in a flexible component or the ODE degrees of freedom in a rigid component.



In cases where locking and constraint conditions are present on a joint, the initial values of the joint degrees of freedom should be within the specified minimum and maximum limits.

### CREATING JOINTS AUTOMATICALLY

The **Create Joints** option automatically creates **Joint** nodes from the geometry sequence, based on the shape of the boundaries. This option requires **Identity Boundary Pair** nodes created under **Definitions**. If the **Source Boundaries** and **Destination Boundaries** are of same standard shape, this option creates one **Joint** node for each active **Identity Boundary Pair**.

In 3D, planar, cylindrical, and spherical are the allowed boundary shapes, for which automatic joint creation is possible. In 2D, joints can be formed for straight and circular boundaries. The possible shapes of boundaries for automatic joint creation are listed under **Joint types**.

For 3D geometries, if both **Source Boundaries** and **Destination Boundaries** of the **Identity Boundary Pair** are of planar shape, a **Prismatic Joint** or a **Planar Joint** can be selected under **Planar boundaries**. For **Cylindrical boundaries**, the joint can be a **Hinge Joint**, a **Cylindrical Joint**, or a **Prismatic Joint**. Using **Spherical boundaries**, a **Ball Joint** or a **Hinge Joint** can be automatically created between spherically shaped boundaries of an **Identity Boundary Pair**.

For 2D geometries, a **Prismatic Joint** can be formed between the **Straight boundaries**. For **Circular boundaries**, a **Hinge Joint** can be automatically created between them.

For the above cases, two additional options **Fixed Joint** and **None** are available. If you want to create a **Fixed Joint** between similar shaped boundaries, use the **Fixed Joint** option. Similarly, if you do not want to create any **Joint** node between the boundaries

of a particular shape, choose the **None** option in the corresponding menu, under **Joint types**.

---

To automatically create **Identity Boundary Pair** nodes between geometrically disconnected objects, set the geometry sequence finalization step to **Form Assembly** from the default **Form Union**. If you want to automatically generate a joint node between any boundaries which are not part of the existing identity boundary pairs, create a new **Identity Boundary Pair** node with desired boundaries selected.

If the selection of **Source Boundaries** and **Destination Boundaries** contain mixed shape boundaries, it is not possible to automatically create any types of joints between them.

The **Create Joints** functionality only works for models with a geometry sequence. For models containing only a mesh, this functionality does not automatically create joints.

If you do not want to create joints between boundaries of a few specified **Identity Boundary Pair** nodes, temporarily disable them before clicking the **Create Joints** button. This will omit these boundaries from automatic joint creation.

---

After setting the joint type for each of the boundary shape, click the **Create Joints** button. This will create two **Attachments** and a corresponding **Joint** node.

- The selection of the two **Attachment** nodes is same as the selection of the **Source Boundaries** and **Destination Boundaries** respectively.
- The attachment nodes are set as **Source** and **Destination** of the Joint node.
- If there is an existing **Attachment** node on any of these boundaries, no new **Attachment** node is created. Instead, the selection of the existing **Attachment** is modified to include the selection of the **Source Boundaries** or **Destination Boundaries**.
- If the selection of **Source Boundaries** or **Destination Boundaries** is part of any **Rigid Domain** node selection, then no new **Attachment** node is created on these boundaries. Instead, the corresponding **Rigid Domain** is set as the **Source** or **Destination** of the Joint node.

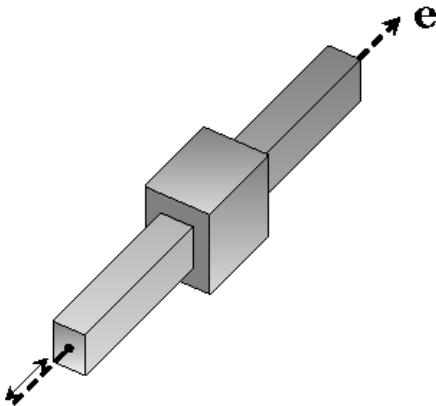
For the joints created between cylindrical boundaries, the center and axis are calculated from the geometry and entered as the **Center of Joint** and **Axis of Joint** inputs in the

settings. Similarly, for the joints created between spherical boundaries, the center is calculated from the geometry and entered as the **Center of Joint** input.

### *Prismatic Joint*

---

The **Prismatic Joint**, also known as a translational or sliding joint, has one translational degree of freedom between the two connected components. The components are free to translate, or slide, relative to each other along the joint axis.



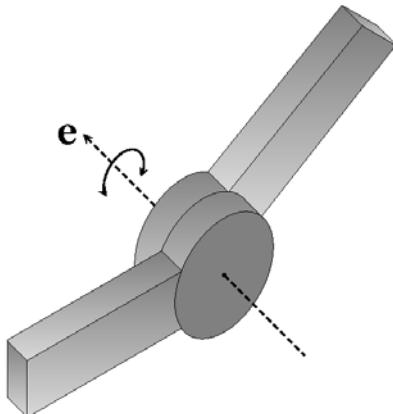
[Prismatic Joint Theory](#)

---

## *Hinge Joint*

---

The [Hinge Joint](#), also known as a revolute joint, has one rotational degree of freedom between the two components. The two components are free to rotate relative to each other about the axis of the joint.



### [Hinge Joint Theory](#)

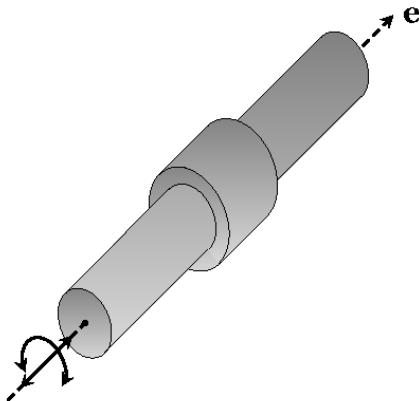
---

## *Cylindrical Joint*

---

The [Cylindrical Joint](#) has one translational and one rotational degree of freedom between the two connected components. The components are free to slide and rotate

relative to each other along the axis of joint. A cylindrical joint can be thought of as a combination of a prismatic joint and a hinge joint.



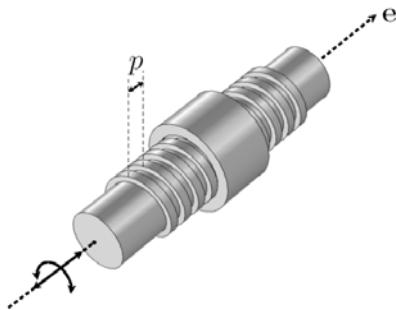
### Cylindrical Joint Theory

---

## *Screw Joint*

---

The **Screw Joint** is similar to the cylindrical joint except that the relative displacement and the relative rotation, instead of being independent, are linearly related to each other. Hence, this joint effectively has only one degree of freedom.



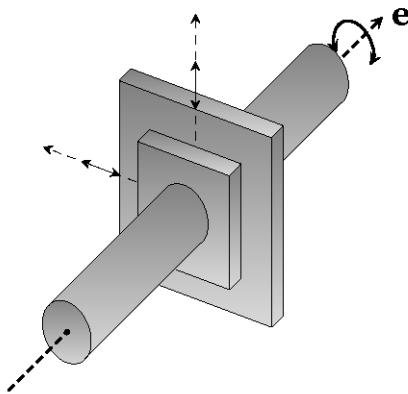
[Screw Joint Theory](#)

---

## *Planar Joint*

---

The **Planar Joint** has two translational and one rotational degrees of freedom between the two components. Both components are free to translate relative to each other in a plane perpendicular to the axis of joint, and they are also free to rotate about this axis.



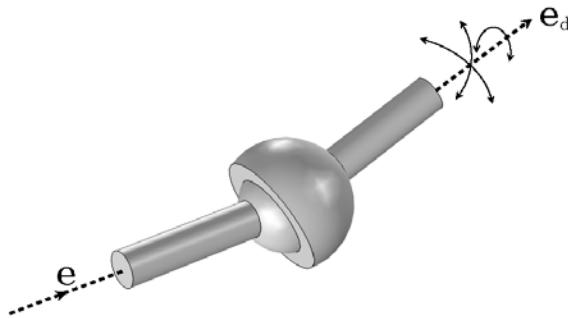
[Planar Joint Theory](#)

---

## *Ball Joint*

---

The [Ball Joint](#), also known as a spherical joint, has three rotational degrees of freedom between the two attached components. The two components are free to rotate relative to each other along all three axes.



## [Ball Joint Theory](#)

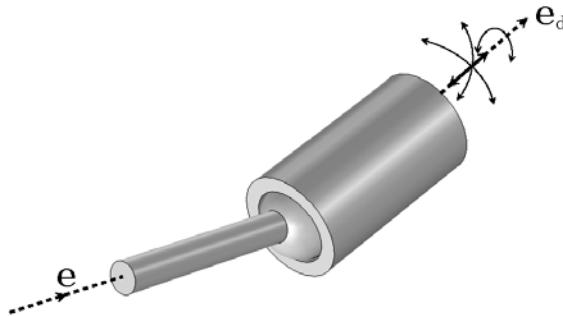
---

## *Slot Joint*

---

The [Slot Joint](#) has three rotational and one translational degrees of freedom between the two attached components. The components are free to rotate relative to each other

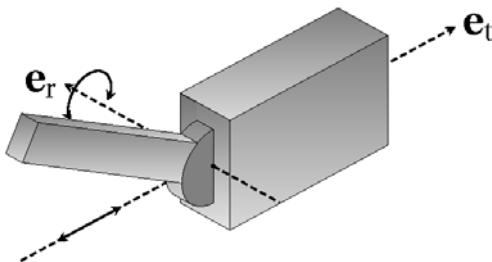
about all three axes and also free to translate along the joint translation axis. The slot joint can be seen as a combination of the prismatic joint and the ball joint.



#### Slot Joint Theory

#### *Reduced Slot Joint*

The **Reduced Slot Joint** has one rotational and one translational degree of freedom between the two attached components. Both components of the system are free to rotate about the joint rotational axis and free to translate along the joint translation axis. The reduced slot joint can be thought of as a combination of the prismatic joint and the hinge joint, with different axes for the allowed degrees of freedom. The translational axes and the rotational axis are orthogonal to each other.



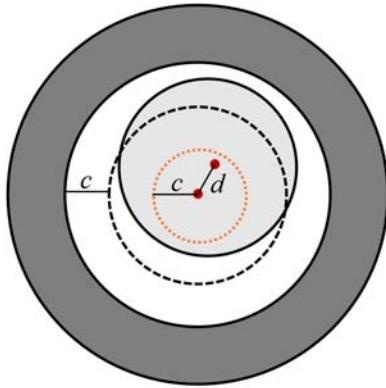
#### Reduced Slot Joint Theory

The reduced slot joint in 2D is similar to the reduced slot joint in 3D. The only difference is the initial joint rotational axis ( $\mathbf{e}_{\text{rot}0}$ ), which does not change in 2D and is always the out-of-plane axis ( $\mathbf{e}_z$ ).

### *Clearance Joint*

---

The **Clearance Joint** allows two points that are connected to a pair of attachments or rigid domains to move within a specified clearance distance. The components connected through clearance joint are free to rotate relative to each other about all three axes and free to translate relative to each other within the specified clearance limit.



#### [Clearance Joint Theory](#)

---

### *Fixed Joint*

---

The **Fixed Joint** rigidly connects the two components and does not allow any relative motion between them. It does not add any degrees of freedom.

---



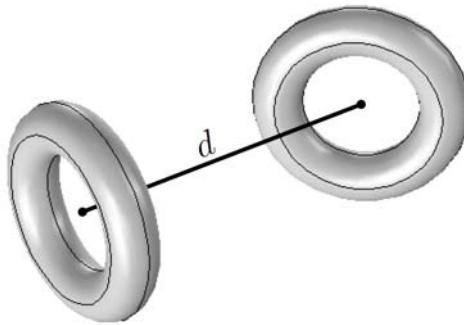
#### [Fixed Joint Theory](#)

---

## *Distance Joint*

---

The **Distance Joint** maintains a specified distance between two points that are connected to a pair of attachments or rigid domains. As a default, the initial distance is maintained, so that the joint can be considered as a rigid link with ball joints at both ends. It is also possible to vary the distance so that the joint acts as a telescopic link.



### [Distance Joint Theory](#)

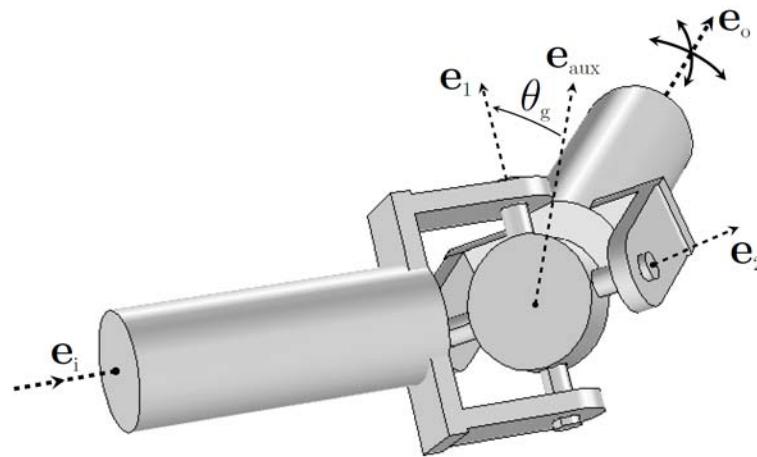
---

## *Universal Joint*

---

An **Universal Joint**, also known as Hooke's joint or Cardan joint, is commonly used to transmit rotational motion between shafts that have different orientations. Physically this joint is made up of a pair of hinge joints, oriented perpendicular to each other, connected by a cross shaft or gimbal. This joint is a kind of abstract joint where the gimbal is not modeled and both the shafts are directly connected to each other. In general, this joint is used to transfer mechanical power from one shaft to another shaft.

The added degrees of freedom in this joint are the relative rotations about the two legs of the gimbal that form the joint axis and second axis.



Universal Joint Theory

---

# Attachments

An [Attachment](#) is a set of boundaries, edges, or points on a flexible or rigid component used to connect it to another flexible component or a rigid component through a joint or spring. An attachment can be defined on the boundaries of a solid element, on the edges of a shell element, and on the points of a beam element.

When an attachment is connected to a flexible body, you can use two different formulations: rigid and flexible.

In the rigid attachment formulation, all selected boundaries or edges behave as if they were connected to a common rigid body. This may cause an unrealistic stiffening and local stress concentrations.

In the flexible version, the boundaries are allowed to deform, and the rigid body constraints are enforced only in an average sense.

The attachment formulation is similar to that of a rigid connector. In the rigid case, the only degrees of freedom needed to represent this assembly are the ones needed to describe the movement of a rigid body. In 2D this is just two in-plane translations, and the rotation around the  $z$ -axis.

In 3D the situation is more complex. Six degrees of freedom are necessary, usually selected as three translations and three parameters for the rotation. For finite rotations any choice of three rotation parameters is singular at some specific set of angles. For this reason, a four-parameter quaternion representation is used for the rotations.

Some extra degrees of freedom are added for each attachment where the flexible formulation is used.

When an attachment is defined on a rigid component, it does not create any degrees of freedom of its own and directly picks the degrees of freedom of the rigid component.



---

The formulation of the attachment feature is same as the rigid connector formulation, which is discussed in [Rigid Connector Theory](#) in the *Structural Mechanics Module User's Guide*.

---



The links to sections described in external guides, such as the *Structural Mechanics Module User's Guide*, do not work in the PDF, only from the on-line help in COMSOL Multiphysics.

Some useful information about the attachment feature:

- The attachment can be defined either on a flexible or on a rigid component. It is not possible to select the boundaries from both types of components in a single attachment. The reason is the different formulation of attachment for flexible or rigid components.
- The attachment center of rotation is the centroid of its selected boundaries, edges, or points. In a joint, it is possible to select the attachment center of rotation as the center of the joint.
- If the attachment is connected to a flexible domain, the forces and moments on an attachment with a boundary selection are computed by summing the reaction forces on the selected boundaries. There is an option to use these forces and moments to evaluate the joint forces.
- For an attachment in the Shell interface, the selection consists of edges. The forces and moments on an attachment are then computed by summing the reaction forces and reaction moments on the selected edges. There is an option to use these forces and moments to evaluate the joint forces.
- For an attachment in the Beam interface, the selection consists of points. The forces and moments on an attachment are then computed by summing the reaction forces and reaction moments on the selected points. There is an option to use these forces and moments to evaluate the joint forces.
- When an attachment is defined in the Shell or Beam interface, the normal force cannot be defined through **Attachment reaction forces** in the **Friction** subnode of a joint.
- In a joint you select attachments to establish its connection with flexible, or optionally, rigid bodies. This is accomplished by setting up a relation between the source attachment and destination attachment degrees of freedom. Thus, the motion of an attachment is governed by the joints where this attachment is used.
- An attachment which is not referenced by any joint acts as an unconstrained rigid connector.



- The Multibody Dynamics Interface
- The Solid Mechanics Interface, The Shell and Plate Interfaces, and The Beam Interface in the *Structural Mechanics Module User's Guide*.

---

# Cam-Follower

A cam is defined through a set of boundaries or edges that is being followed by a point called “follower”. The cam selection can be defined on rigid as well as on elastic bodies. Thus, cam boundaries or edges can undergo any kind of rigid body motion as well as deformation. When defining a cam selection, it should also be possible to use boundaries or edges that are defined outside the physics interface.

A follower is defined through a point which follows the cam boundaries or edges with certain optional offset. The follower point can be a part of rigid or elastic bodies.

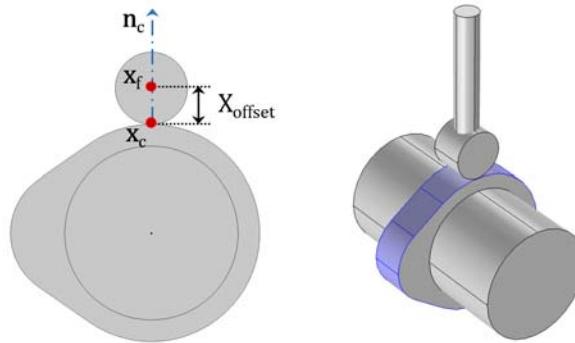


Figure 2-1: Example of a cam-follower connection.

The following settings help in avoiding convergence issues and getting more accurate results:

- The geometry shape order should be quadratic or higher. By default, it is the same as the shape order of the physics dependent variable. Select quadratic as the discretization for the physics dependent variable.
- The finite element mesh should be sufficiently fine on the cam selection in order to get accurate spatial position, spatial normal, and spatial tangent of the closest point.
- The solver time steps should be sufficiently small such that the bodies do not rotate or move by a large amount in single time step.



# Springs and Dampers

You can use a [Spring-Damper](#) to connect two points by an elastic spring, a viscous damper, or both. There are two variants of spring — *directional* and *matrix*. The two connection points are called the *source* and *destination* points, respectively.

- A source point can be placed on a flexible body, a rigid domain, an attachment, a fixed position, or a moving base.
- A destination point can be placed on a flexible body, an attachment, a rigid domain, or a prescribed displacement.

When a flexible body is defined within the Multibody Dynamics interface, a connection point can be directly selected on the body. In the case when a flexible body is defined in another interface (Solid Mechanics, Shell, or Beam interface) an attachment is needed to define a connection point.

In directional spring, the two points can move relative to each other in an arbitrary way as long as they do not coincide. The spring and damping forces act along the line between the current positions of the two connection points.

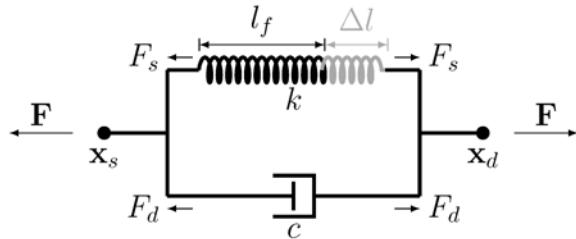


Figure 2-2: Conceptual sketch of a Spring-Damper.

You can specify that the Spring-Damper should be deactivated under certain conditions. It can for example be active only in tension, or break at a certain elongation.



[Spring-Damper Theory](#)

# Gears

## About Gears

A gear is defined as a rigid body. A finite stiffness for the gear mesh (or gear tooth) can be specified in the gear pair node while connecting to other gears. Similar to the rigid domain, the gear is a material model, which is mutually exclusive to all other material models. The only material property needed is the mass density.



For details about the theory of a rigid body see [Rigid Domain Model](#) in the *Structural Mechanics Module User's Guide*.

The rest of this section has these topics:

- [Gear Types](#)
- [Connection with Other Gears](#)
- [Center of Rotation](#)
- [Gear Axis](#)
- [Mounting Methods](#)
- [Connecting to Other Physics Interfaces](#)

### GEAR TYPES

There are six type of gears available:

- [Spur Gear](#)
- [Helical Gear](#)
- [Bevel Gear](#)
- [Worm Gear](#)
- [Spur Rack](#)
- [Helical Rack](#)

The bevel and worm gears are available only in 3D.

### CONNECTION WITH OTHER GEARS

A gear can be connected to other gears through following nodes:

- [Gear Pair](#)
- [Worm and Wheel](#)
- [Rack and Pinion](#)

A single gear can be connected to one or more gears.

#### **CENTER OF ROTATION**

Gears are defined through rigid body degrees of freedom. These degrees of freedom are created at the center of rotation and the rotation is interpreted about this point. By default, it is set to the center of mass, but there are other ways also to define it explicitly. This is the point where the forces and moment which act on the gear due to meshing with other gears are interpreted.

#### **GEAR AXIS**

The gear axis is the axis of rotation of the gear. It passes through the center of rotation. This axis is used to create the gear local coordinate system in the gear pair node. The gear rotation, a degree of freedom in the gear pair node, is also interpreted about this axis.

#### **MOUNTING METHODS**

The motion of a gear can be constrained directly using its subnode. Alternatively, it can be mounted on a shaft. There are several ways through which a gear can be mounted on a shaft. Some of the commonly used methods are as follows:

- Directly mounted on a flexible shaft
- Mounted on a rigid or flexible shaft through a rigid or elastic fixed joint
- Mounted on a rigid or flexible shaft through a rigid or elastic hinge joint
- Mounted on a ground (a fixed shaft) through a rigid or elastic hinge joint

#### **CONNECTING TO OTHER PHYSICS INTERFACES**

The multibody dynamics analysis of a geared system provides the contact force variation on the gear tooth and the vibration in various parts of the structure. These results can be further used for subsequent analysis.

- A stress analysis of a gear tooth or a full gear can be performed using the contact force obtained from the multibody dynamics analysis. This analysis can be performed within the Multibody Dynamics interface.
- A fatigue analysis of a gear tooth can be performed using the contact force variation over a mesh cycle or full revolution. This analysis can be performed using the Fatigue interface.
- A set of gears or a gear train operating at a high speed gives rise to high frequency vibration which results in noise. An acoustic analysis of a geared system can be performed using the normal acceleration of the vibrating surfaces. This can be

performed using the Pressure Acoustics, Transient or Pressure Acoustics, Frequency Domain interfaces.

## *About Gear Pairs*

---

This section has these topics:

- [Gear Pair](#)
- [Worm and Wheel](#)
- [Rack and Pinion](#)
- [Gear Pair Compatibility Criteria](#)
- [Gear Elasticity](#)
- [Transmission Error](#)
- [Backlash](#)
- [Friction \(Gear\)](#)

### **GEAR PAIR**

The gear pair node connects two spur gears, helical gears, or bevel gears in such a way that at the contact point they have no relative motion along the line of action. The remaining displacements and rotations of both the gears are independent of each other.

For a line contact model, one additional constraint is added to restrict the relative rotation about a line joining the two gear centers. When friction is included on a gear pair, frictional forces in the plane perpendicular to the line of action are added on both the gears.

The degrees of freedom at the gear pair are  $\theta_{wh}$  and  $\theta_{pn}$ . They are defined as the rotation of the wheel and pinion about the first axis of their respective local coordinate system.



[Theory for Gear Pairs](#)

---

### **WORM AND WHEEL**

The worm and wheel node connects a worm gear with a spur or helical gear so that at the contact point they have no relative motion along the line of action. The remaining displacements and rotations of both the gears are independent of each other.

When friction is included on a worm and wheel node, frictional forces in the plane perpendicular to the line of action are added on both the gears.

The degrees of freedom at the worm and wheel are  $\theta_{wm}$  and  $\theta_{wh}$ . They are defined as the rotation of the worm and wheel about the first axis of their respective local coordinate system.

---



## Theory for Worm and Wheel

### RACK AND PINION

The rack and pinion node connects a spur or helical rack with a spur or helical gear in so that at the contact point they have no relative motion along the line of action. The remaining displacements and rotations of the rack and gear are independent of each other.

For a line contact model, one additional constraint is added to restrict the relative rotation about a line joining the rack and gear centers. If friction is included on a rack and pinion node, frictional forces in the plane perpendicular to the line of action are added on the rack and pinion.

The degrees of freedom at the rack and pinion are  $u_{rk}$  and  $\theta_{pn}$ . They are defined as the displacement of rack along the third axis and the rotation of pinion about the first axis of their respective local coordinate system.

---



## Theory for Rack and Pinion

### GEAR PAIR COMPATIBILITY CRITERIA

In a gear pair node, you can select any two gears defined for the model. For the correct tooth meshing, a set of gears must however fulfill the following compatibility criteria:

#### *Normal Module*

The normal module of the both the gears must be same.

$$m_{n, wh} = m_{n, pn}$$

#### *Pressure Angle*

The pressure angle of the both the gears must be same.

$$\alpha_{wh} = \alpha_{pn}$$

### *Configuration*

For parallel axis helical gears, the sum of the helix angles must be zero.

$$\beta_{wh} + \beta_{pn} = 0$$

For crossed axis helical gears, the sum of the helix angles must be nonzero.

$$\beta_{wh} + \beta_{pn} \neq 0$$

### **GEAR ELASTICITY**

Gear elasticity can be added to gear pairs. In this node, elastic properties of the gear mesh such as mesh stiffness, mesh damping, and contact ratio can be specified. By default, the gear mesh is assumed to be rigid.



[Theory for Gear Elasticity](#)

### **TRANSMISSION ERROR**

A transmission error can be added to gear pairs. In this node, the static transmission error can be specified. It is added in the rotation constraint of the gear pair. By default, there is no transmission error for a gear pair and it is assumed to be an ideal pair.

The static transmission error could be due to geometrical errors and geometrical modifications such as gear runouts, misalignments, tooth tip and root relief.



[Theory for Transmission Error](#)

### **BACKLASH**

A backlash can be added to gear pairs. The transmission error due to backlash is added to the rotation constraint of the gear pair. By default, there is no backlash for a gear pair.

The backlash on a gear pair plays an important role for unloaded gears. For loaded gears, it does not affect the dynamics of gears to a large extent.



[Theory for Backlash](#)

### FRICITION (GEAR)

Friction can be added to gear pairs. When friction is included for a gear pair, frictional forces in the plane perpendicular to the line of action are added on both the gears.

By default, there is no friction on [Gear Pair](#) and [Rack and Pinion](#). It is, however, included for [Worm and Wheel](#). The worm and wheel pair has always neither parallel nor intersecting configuration and in this configuration, there is always a slip even under perfect conditions.



Theory for Gear Friction

---

# Gear Geometries

The Multibody Dynamics Module has a built-in geometry library for creating different types of gears. The gear geometries are available for 3D and 2D models. You can use them to build a gear tooth, a single gear, a pair of gears, or a gear train.

All the gear geometries are parameterized. You can vary input parameters to customize the gear tooth or gear blank shape.

To avoid building an invalid geometry, there are several checks on the input parameters. The set of input parameters must result in a valid gear geometry. In case of invalid set of parameters, an appropriate error message is displayed to adjust the specific parameter.

The geometries of the gear tooth cross sections are as follows:

- [Gear Tooth Geometry \(External: 2D\)](#)
- [Gear Tooth Geometry \(Internal: 2D\)](#)
- [Rack Tooth Geometry \(2D\)](#)

The gear geometries in 2D are as follows:

- [Spur Gear Geometry \(External: 2D\)](#)
- [Spur Gear Geometry \(Internal: 2D\)](#)
- [Spur Rack Geometry \(2D\)](#)

The gear geometries in 3D are as follows:

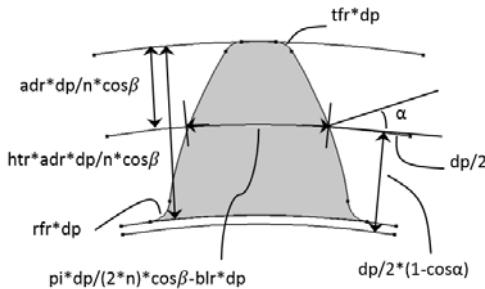
- [Spur Gear Geometry \(External: 3D\)](#)
- [Spur Gear Geometry \(Internal: 3D\)](#)
- [Helical Gear Geometry \(External: 3D\)](#)
- [Helical Gear Geometry \(Internal: 3D\)](#)
- [Bevel Gear Geometry \(3D\)](#)
- [Worm Gear Geometry \(3D\)](#)
- [Spur Rack Geometry \(3D\)](#)
- [Helical Rack Geometry \(3D\)](#)

## *Gear Tooth Geometry (External: 2D)*

---

The part name for the tooth cross section of an external gear is `gear_tooth_2d`. This cross section of the gear tooth is in the normal plane of the gear.

The following sketch shows all the tooth parameters that can be varied to customize the tooth shape.



*Figure 2-3: The sketch of the tooth cross section of an external gear.*

### **OPTIONAL FEATURES**

The following geometry operations are optional in this part:

- Fillet at the tooth tip
- Fillet at the tooth root

## *Gear Tooth Geometry (Internal: 2D)*

---

The part name for the tooth cross section of an internal gear is `internal_gear_tooth_2d`. This cross section of the gear tooth is in the normal plane of the gear.

The following sketch shows all the tooth parameters that can be varied to customize the tooth shape.

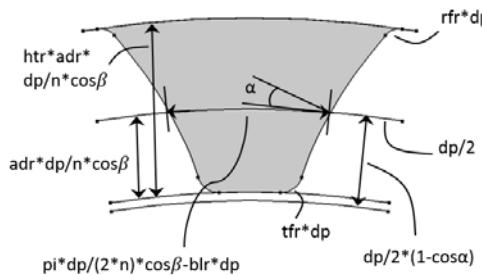


Figure 2-4: The sketch of the tooth cross-section of an internal gear.

#### OPTIONAL FEATURES

The following geometry operations are optional in this part:

- Fillet at the tooth tip
- Fillet at the tooth root

#### Rack Tooth Geometry (2D)

---

The part name for the tooth cross section of a rack is `rack_tooth_2d`. This cross section of the gear tooth is in the normal plane of the gear.

The following sketch shows all the tooth parameters that can be varied to customize the tooth shape.

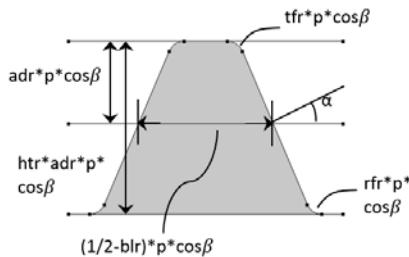


Figure 2-5: The sketch of the tooth cross section of a rack.

## OPTIONAL FEATURES

The following geometry operations are optional in this part:

- Fillet at the tooth tip
- Fillet at the tooth root

## *Spur Gear Geometry (External: 2D)*

---

The part name for the external spur gear geometry in 2D is `spur_gear_2d`.

The following sketch shows all the tooth and gear parameters that can be varied to customize the available geometry.

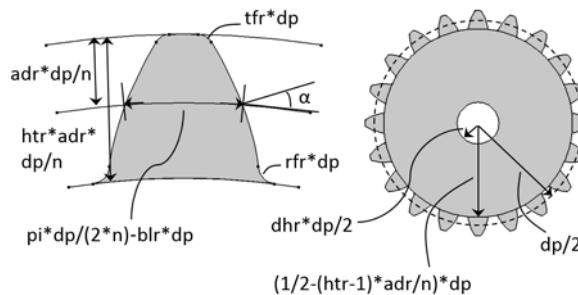


Figure 2-6: The sketch of the external spur gear geometry in 2D.

## OPTIONAL FEATURES

The following geometry operations are optional in this part:

- Fillet at the tooth tip
- Fillet at the tooth root
- Hole at the gear center

## *Spur Gear Geometry (Internal: 2D)*

---

The part name for the internal spur gear geometry in 2D is `internal_spur_gear_2d`.

The following sketch shows all the tooth and gear parameters that can be varied to customize the available geometry.

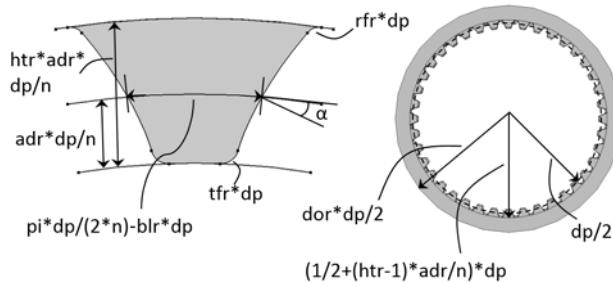


Figure 2-7: The sketch of the internal spur gear geometry in 2D.

#### OPTIONAL FEATURES

The following geometry operations are optional in this part:

- Fillet at the tooth tip
- Fillet at the tooth root

#### Spur Rack Geometry (2D)

The part name for the spur rack geometry in 2D is `spur_rack_2d`.

The following sketch shows all the tooth and rack parameters that can be varied to customize the available geometry.

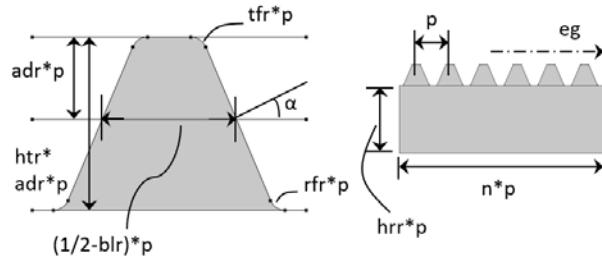


Figure 2-8: The sketch of the spur rack geometry in 2D.

## OPTIONAL FEATURES

The following geometry operations are optional in this part:

- Fillet at the tooth tip
- Fillet at the tooth root

## Spur Gear Geometry (External: 3D)

---

The part name for the external spur gear geometry in 3D is `spur_gear`.

The following sketch shows all the tooth and gear parameters that can be varied to customize the available geometry.

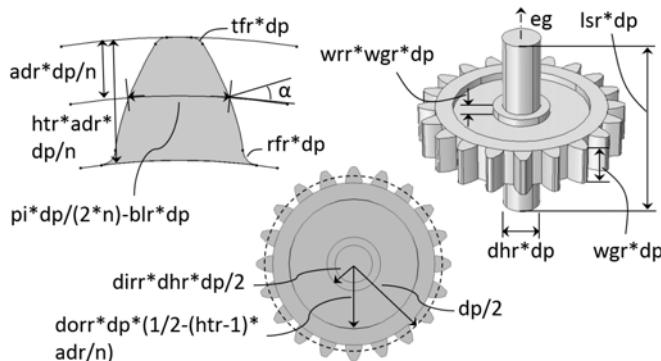


Figure 2-9: The sketch of the external spur gear geometry in 3D.

## OPTIONAL FEATURES

The following geometry operations are optional in this part:

- Fillet at the tooth tip
- Fillet at the tooth root
- Addition of a shaft
- Hole at the gear center
- Material removal from the gear blank

## Spur Gear Geometry (Internal: 3D)

---

The part name for the internal spur gear geometry in 3D is `internal_spur_gear`.

The following sketch shows all the tooth and gear parameters that can be varied to customize the available geometry.

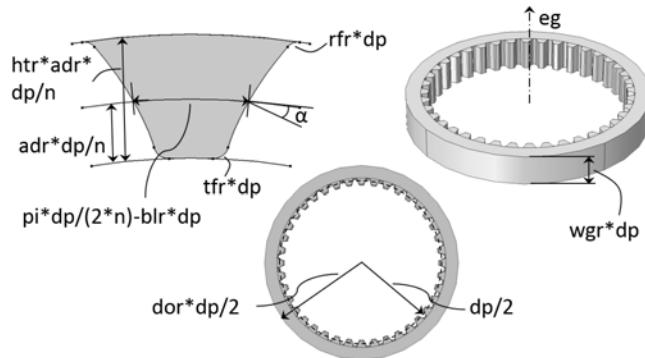


Figure 2-10: The sketch of the internal spur gear geometry in 3D.

#### OPTIONAL FEATURES

The following geometry operations are optional in this part:

- Fillet at the tooth tip
- Fillet at the tooth root

#### *Helical Gear Geometry (External: 3D)*

---

The part name for the external helical gear geometry in 3D is `helical_gear`.

The following sketch shows all the tooth and gear parameters that can be varied to customize the available geometry.

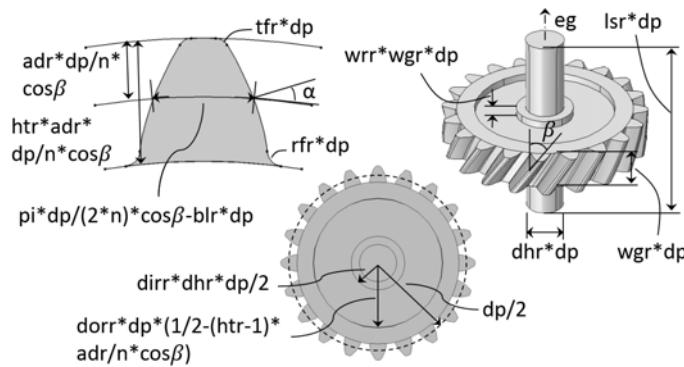


Figure 2-11: The sketch of the external helical gear geometry in 3D.

#### OPTIONAL FEATURES

The following geometry operations are optional in this part:

- Fillet at the tooth tip
- Fillet at the tooth root
- Addition of a shaft
- Hole at the gear center
- Material removal from the gear blank

#### *Helical Gear Geometry (Internal: 3D)*

---

The part name for the internal helical gear geometry in 3D is `internal_helical_gear`.

The following sketch shows all the tooth and gear parameters that can be varied to customize the available geometry.

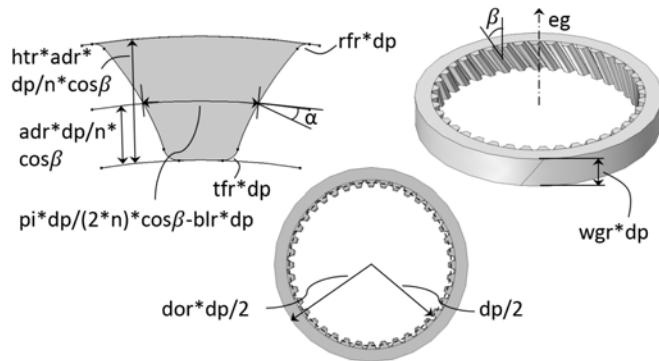


Figure 2-12: The sketch of the internal helical gear geometry in 3D.

#### OPTIONAL FEATURES

The following geometry operations are optional in this part:

- Fillet at the tooth tip
- Fillet at the tooth root

#### Bevel Gear Geometry (3D)

The bevel gear geometry is available only in 3D. The part name for this is `bevel_gear`.

The following sketch shows all the tooth and gear parameters that can be varied to customize the available geometry.

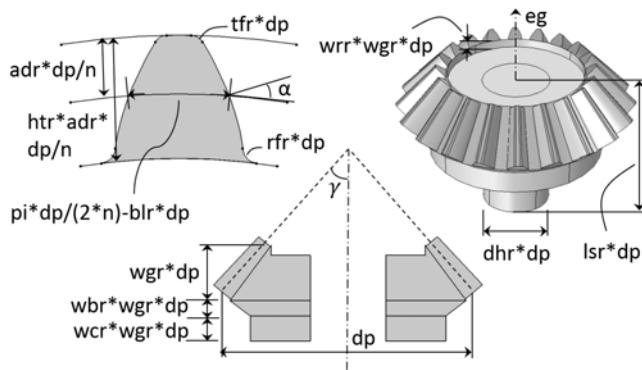


Figure 2-13: The sketch of the bevel gear geometry in 3D.

#### OPTIONAL FEATURES

The following geometry operations are optional in this part:

- Fillet at the tooth tip
- Fillet at the tooth root
- Addition of a shaft
- Hole at the gear center
- Material removal from the gear blank

#### Worm Gear Geometry (3D)

The worm gear geometry is available only in 3D. The part name for this is `worm_gear`.

The following sketch shows all the tooth and gear parameters that can be varied to customize the available geometry.

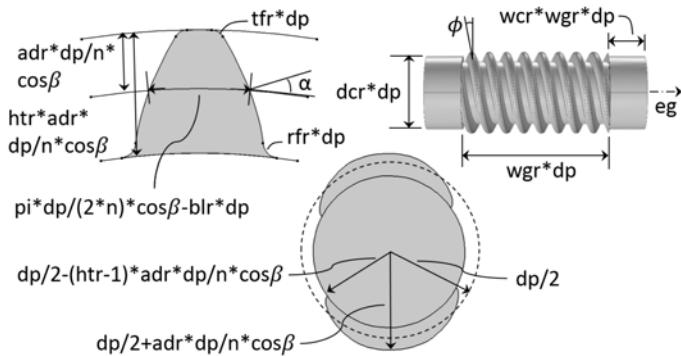


Figure 2-14: The sketch of the worm gear geometry in 3D.

#### OPTIONAL FEATURES

The following geometry operations are optional in this part:

- Fillet at the tooth tip
- Fillet at the tooth root
- Addition of collars

#### Spur Rack Geometry (3D)

The part name for the spur rack geometry in 3D is `spur_rack`.

The following sketch shows all the tooth and rack parameters that can be varied to customize the available geometry.

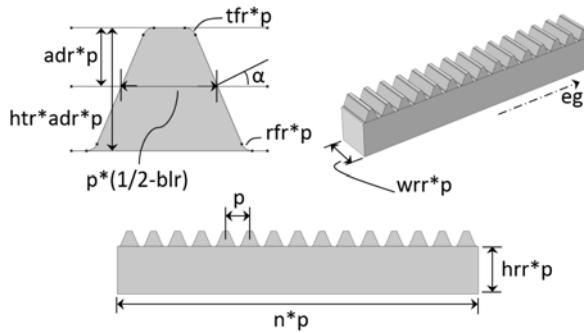


Figure 2-15: The sketch of the spur rack geometry in 3D.

#### OPTIONAL FEATURES

The following geometry operations are optional in this part:

- Fillet at the tooth tip
- Fillet at the tooth root

#### Helical Rack Geometry (3D)

The part name for the helical rack geometry in 3D is `helical_rack`.

The following sketch shows all the tooth and rack parameters that can be varied to customize the available geometry.

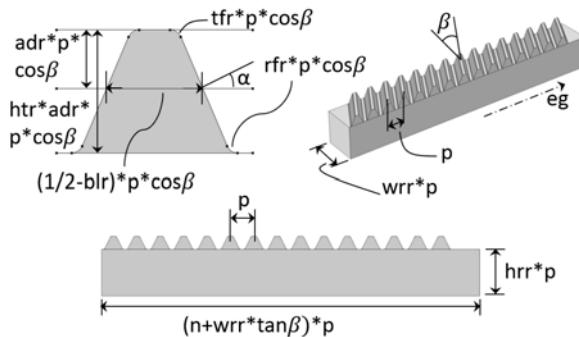


Figure 2-16: The sketch of the helical rack geometry in 3D.

### **OPTIONAL FEATURES**

The following geometry operations are optional in this part:

- Fillet at the tooth tip
- Fillet at the tooth root

# Chain Drive

## *About Chain Drive*

---

Transmission systems are devices used for transferring power from one place to another. A chain drive is an example of such transmission device that is widely used for various industrial applications. A typical chain drive is an assembly of two or more sprockets with a chain wrapped around them. Like gears, chain drives also transfer mechanical power from one shaft to another, however, they are more useful when the two shafts are placed at a distance from each other.

Using the [Chain Drive](#) node in the Multibody Dynamics interface, you can model a roller chain sprocket assembly in 2D or 3D. The Chain Drive node determines the interaction of the chain drive assembly, and automatically generates a set of physics nodes that are used to describe its behavior.

This section includes the following topics:

- [Geometry Settings](#)
- [Selection Settings](#)
- [Chain and Sprocket Types](#)
- [Contact Method](#)
- [Joint Settings](#)
- [Create Links and Joints Button](#)

### **GEOMETRY SETTINGS**

The Chain Drive node automatically creates a set of physics nodes in the Multibody Dynamics interface. This requires a geometry of the chain drive systems and a set of appropriate domain and boundary selections. With the COMSOL Part Library, you can either import a complete roller chain sprocket assembly, or create your own assembly from individual chain links and sprockets. Geometry parts are available in both 3D and 2D. The selections required to set up the physics of the chain drive system are predefined in the built-in geometry parts.



See the [Chain Geometries](#) section for details about various built-in geometries available in the COMSOL Part Library.

You can also create or import your own geometries, however, then you need to make sure that the final geometry is in the assembly state. Furthermore, you also need to create all selections required to set up the physics of the chain drive system manually.

## SELECTION SETTINGS

For a selected geometry part, the Chain Drive node automatically creates a set of physics nodes such as Rigid Domain, Attachment, Hinge Joint, and Contact. These physics nodes are used for modeling the chain drive system. Creation of these physics nodes is based on a set of domain and boundary selections on the selected geometry.

There are several selection inputs required by the Chain Drive node. If the geometry of your chain drive system is built using the COMSOL Part Library, the Chain Drive node automatically selects proper domain and boundary selections from the geometry part.



See the [Chain Geometries](#) section for more details about the selections available for different geometry parts.

When using the built-in parts, you do not need to give any additional selection inputs. However, if you are using your own geometry, you need to input proper domain and boundary selections for each of the selection inputs as described below:

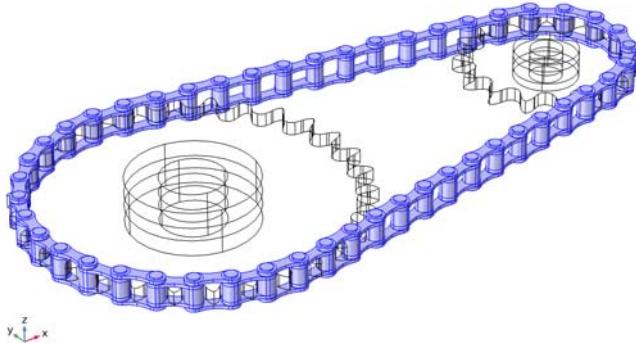
- [Domain Selection, Link](#)
- [Domain Selection, Sprocket](#)
- [Domain Selection, Bushing](#)
- [Boundary Selection, Pin](#)
- [Boundary Selection, Roller Inner](#)
- [Boundary Selection, Roller Outer](#)
- [Boundary Selection, Sprocket Outer](#)
- [Boundary selection, Sprocket Inner](#)

All the above selection inputs may not always be available. Depending on the type of modeling and other input parameters, some selection inputs may not be required. Selection inputs that are not required are hidden in the settings window of the Chain Drive node. For example, if you choose not to model the elastic bushings in a chain link, the [Domain Selection, Bushing](#) selection input does not appear in the Chain Drive node. Details about each of these selection inputs are given below.

### *Domain Selection, Link*

This is a domain selection input used to create a Rigid Domain node on each link plate of the chain. If you use a geometry from the COMSOL Part Library, a built-in domain selection named `Links` is automatically selected. If you are using your own geometry,

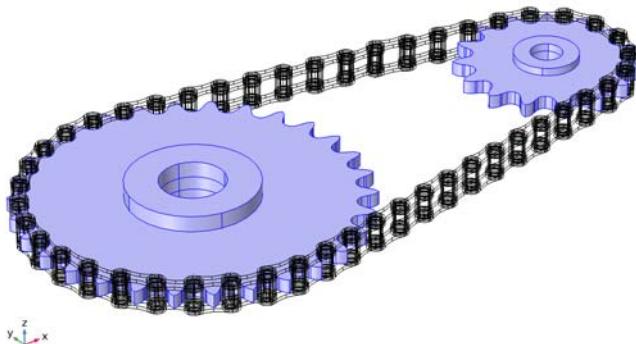
you need to input a domain selection containing all link plates. This selection input is available only when the chain links are modeled as rigid bodies.



*Figure 2-17: Selection of all the link plates used for creating Rigid Domain nodes.*

#### *Domain Selection, Sprocket*

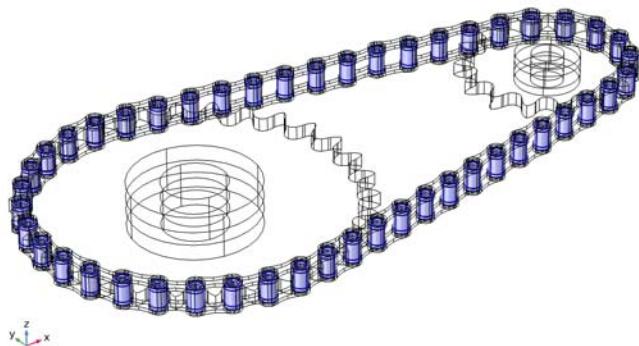
This is a domain selection input used to create a Rigid Domain node on each sprocket. If you use a geometry from the COMSOL Part Library, a built-in domain selection named **Sprockets** is automatically selected. If you are using your own geometry, you need to input a domain selection containing both sprockets domains. This selection input is available only when the sprockets are modeled as rigid bodies.



*Figure 2-18: Selection of both sprockets used for creating Rigid Domain nodes.*

### *Domain Selection, Bushing*

If you choose to model the chain links as rigid bodies with elastic bushings inside, this selection input is required. The input needed here is a domain selection containing all bushing domains. Using this input, the Chain Drive node automatically creates a Linear Elastic Material node on all bushing domains. If you use a geometry from the COMSOL Part Library, a built-in domain selection named **Bushings** is automatically selected. If you are using your own geometry, you need to input a domain selection containing all bushings domains. This selection input is available only when the chain links are modeled as rigid bodies with elastic bushings.



*Figure 2-19: Selection of bushing domains used for creating Linear Elastic Material node.*

### *Boundary Selection, Pin*

This input is a boundary selection used to create Attachment nodes on each pin plate of the chain. For 3D models created using the COMSOL Part Library, a built-in boundary selection named **Pin Outer Boundaries** is automatically selected. If you use your own geometry, you need to input a boundary selection containing the outer cylindrical surfaces of all pin plates.

For 2D models created from the COMSOL Part Library, a selection named **Pin Inner Boundaries** is automatically selected. If you use your own geometry, you need to input a boundary selection containing the inner boundaries of all pin plates.

### *Boundary Selection, Roller Inner*

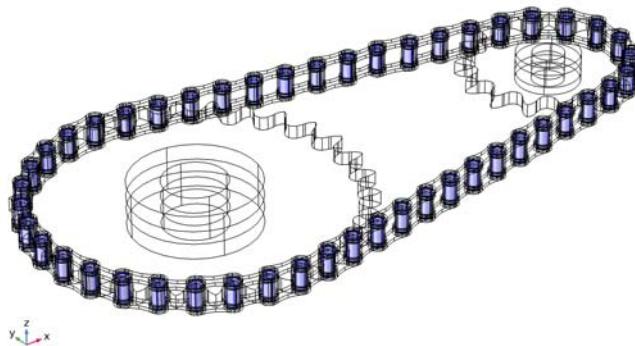
This input is a boundary selection used to create Attachment nodes on each roller plate of the chain. For models built using a geometry from the COMSOL Part Library, a built-in boundary selection named **Roller Inner Boundaries** is automatically

selected. If you use your own geometry, you need to input a boundary selection containing the inner cylindrical surfaces of all roller plates.



Note that this selection is at the same geometrical position as the **Pin Outer Boundaries** selection.

Attachments created on the pin and roller plates having the same locations are used as the source and destination for hinge joints between them.



*Figure 2-20: Selection on pin plate boundaries used for creating Attachment nodes. The roller inner boundaries selection is also at the same geometrical position.*

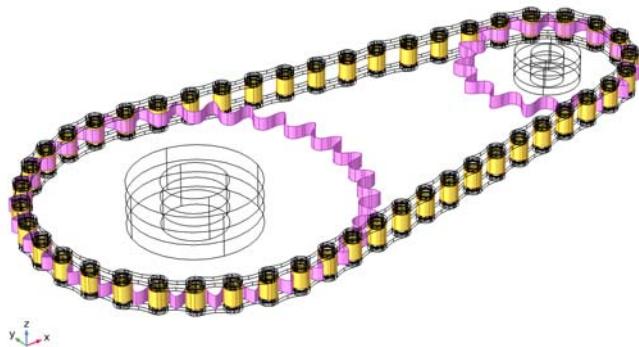
#### *Boundary Selection, Roller Outer*

This input is a boundary selection used to model contact between the chain and the outer boundaries of the sprockets. If the contact method is mesh based, this selection is used for creating a Contact Pair between the roller plates and the sprockets. For models built using a geometry from the COMSOL Part Library, a built-in boundary selection named **Roller Outer Boundaries** is automatically selected. If you use your own geometry, you need to input a boundary selection containing the outer cylindrical surfaces of all roller plates.

#### *Boundary Selection, Sprocket Outer*

This boundary selection input is used to model contact between the chain and the outer boundaries of the sprockets. If the contact method is mesh based, this selection is used for creating a Contact Pair between the roller plates and the sprockets. For models built using a geometry from the COMSOL Part Library, a built-in boundary selection named **Sprocket Outer Boundaries** is automatically selected. If you use

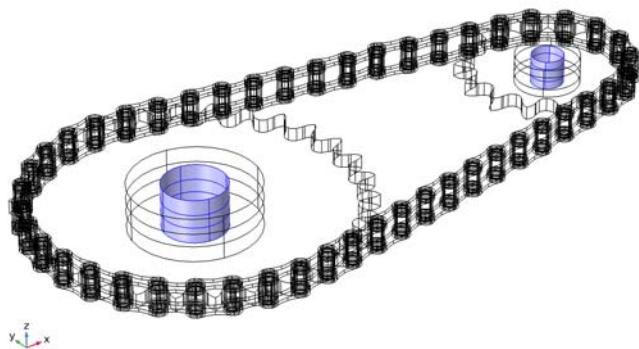
your own geometry, you need to input a boundary selection containing the outer surfaces of the sprockets.



*Figure 2-21: Contact Pair between the outer boundaries of the roller and the outer boundaries of the sprockets.*

#### *Boundary selection, Sprocket Inner*

This boundary selection input is used to create an Attachment and a Hinge Joint on each sprocket. These can be used to model the mounting the chain drive system to external components such as shafts. For models built using a geometry from the COMSOL Part Library, a built-in boundary selection named **Sprocket Inner Boundaries** is automatically selected. If you use your own geometry, you need to input a boundary selection containing the inner surfaces of both sprockets.



*Figure 2-22: Selection of inner boundaries of the sprocket used for creating Attachment and Hinge Joint nodes.*

## CHAIN AND SPROCKET TYPES

Using **Chain Drive** node, you can either model the chain links as rigid or elastic bodies. If the link plates are assumed rigid, the **Chain Drive** node automatically creates a **Rigid Domain** node for each link plate, see [Domain Selection, Link](#). It is also possible to model a chain with elastic bushings present between rigid link plates, see [Domain Selection, Bushing](#).

The sprockets can also either be modeled as rigid or elastic bodies. If the sprockets are assumed rigid, the **Chain Drive** node automatically creates a **Rigid Domain** node for each sprocket, see [Domain Selection, Sprocket](#)

## CONTACT METHOD

The contact condition between the chain links and outer boundaries of the sprockets can be modeled with two different methods:

- Mesh based
- Roller center based

For the mesh-based contact method, the **Chain Drive** node automatically creates a contact pair between the outer boundaries of the rollers and the outer boundaries of the sprockets. A **Contact** node with penalty formulation is also added to the physics, which uses the created Contact Pair for computing the contact forces.

When the roller plates are assumed to be rigid, the roller center-based contact method can be used. From the center of each roller cylinder, the closest point on the outer boundary of the sprocket is determined using a general extrusion operator. The gap distance is defined as the distance between the outer boundary of the roller and its closest point on the sprocket in the direction of the spatial normal. This method is a penalty-based formulation, where the contact force is applied if the gap between a roller and the sprocket is negative. The input penalty factor determines the stiffness of the spring preventing penetration of the contacting bodies.

The mesh-based method is the most accurate of the two methods. It is, however, computationally expensive and only available when at least one of the two contacting bodies is assumed to be elastic. On the other hand, the roller center-based method is faster, but only available if the roller plates are assumed rigid.

## JOINT SETTINGS

The **Chain Drive** node automatically generates a **Hinge Joint** node between each roller and pin plate. The axis of the **Hinge Joints** is the same as the sprocket axis. If you use a geometry from the COMSOL Part Library, the sprocket axis is automatically taken

from the geometry part. It is also possible to change the axis of the sprocket, either by specifying a direction, or by selecting an edge parallel to the sprocket axis.

The **Attachment** nodes created on the roller and pin plates having same geometrical positions are used as the source and destination for the corresponding **Hinge Joint**. These attachments can either be rigid or flexible, depending on the setting in the parent **Chain Drive** node.

By default, the joints are assumed rigid, and the source and destination attachments are then rigidly connected in the directions in which the relative motion is restricted. However, it is possible to insert an elastic connection between attachment surfaces by setting **Joint Type** to **Elastic** in the **Chain Drive** node. This automatically sets all **Hinge Joint** nodes to be elastic. From the **Chain Drive** node, you can also control the viscous damping properties of each **Hinge Joint** in order to model losses, by activating rotational damping and setting a value for the damping coefficient  $c$ .

In most applications, the chain drive system is mounted some external component, such as shafts. Including the mounting is automated in the **Chain Drive** node, and you can select a check box to automatically create an **Attachment** node and a **Hinge Joint** node for each sprocket.

#### **CREATE LINKS AND JOINTS BUTTON**

The **Creates Link and Joints Button** is used to generate all physics nodes required to model the roller chain sprocket assembly. Keep the following points in mind when using this button:

- Only click the **Create Links and Joints** button after setting appropriate values for all parameters in the **Chain Drive** node.
- If you change one or more parameters in the **Chain Drive** node after the automatic creation of physics nodes, the settings of the associated physics nodes also require updates. Click the **Create Links and Joints** button again in order to update the settings of the existing physics nodes. A warning message is added under the **Chain Drive** node to notify when such an update is required.
- If any selections or related parameters in the **Chain Drive** node are modified after the automatic creation of physics nodes, all associated physics nodes needs to be recreated. Click the **Create Links and Joints** button again in order to do this. Since all physics nodes are deleted and created again during this operation, the update may take a while. A warning message is added under the **Chain Drive** node to notify when such an update is required.

# Chain Geometries

The Multibody Dynamics Module has a built-in geometry library for creating roller chains, sprockets, and roller chain sprocket assemblies. All geometry parts are available for 3D and 2D models and can be used to build a chain drive system.

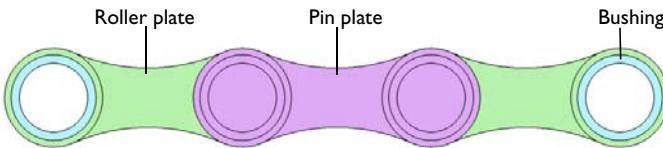
All chain and sprocket geometry parts are fully parameterized. You can modify the input parameters to customize the shape of any chain or sprocket, as well as any roller chain sprocket assembly.

The set of input parameters for a geometry part must result in a valid roller chain, sprocket or roller chain sprocket assembly geometry. Several parameter checks are done when building the geometry to avoid creating invalid geometry parts. If an invalid parameter definition is found, an appropriate error message is displayed that includes information on how to correct the definition.

## *Components of a Roller Chain Sprocket Assembly*

---

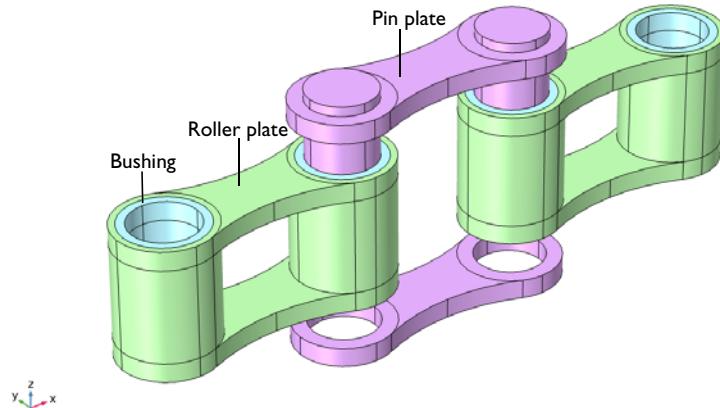
A roller chain sprocket assembly consists of two sprockets with a closed chain wrapped around them. The chain modeled here is of a roller type, and is constructed by joining a number of link plates together. A typical roller chain has two types of link plates, namely roller plates and pin plates. These plates are connected in such a way that the relative rotation between them is unrestricted. Often, elastic bushings are also present between the roller plates and the pin plates. The sketch in [Figure 2-23](#) shows the arrangement of the roller plates and the pin plates in a 2D roller chain unit.



*Figure 2-23: Components of a roller chain unit in 2D.*

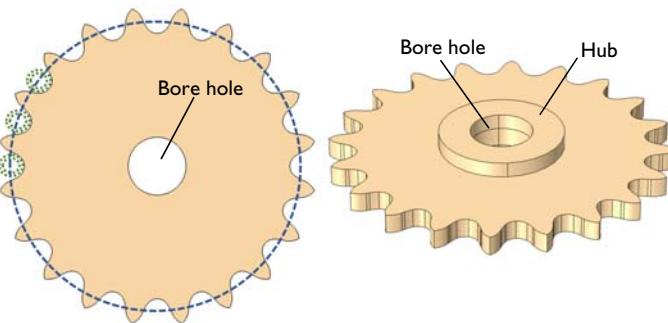
For a 3D roller chain, the constituent components are 3D link plates. Here, the roller plate is made of two hollow cylinders joined by two side plates. Similarly, the pin plate is a union of two solid cylinders joined by two side plates. A chain is formed by inserting the solid pin plates into the hollow cylindrical legs of the adjacent roller plates, producing a press-fit connection. This type of connection allows for relative

rotation between the links, thereby helping the transmission of motion from one part of the system to another part. Optionally, elastic bushings between the roller and pin plates can be included. The sketch in [Figure 2-24](#) shows the arrangement of the roller plates and the pin plates in a 3D chain unit.



*Figure 2-24: Exploded view of a roller chain unit in 3D.*

In 2D, a sprocket is a circular object with a number of teeth in which the rollers of the chain continuously engage and disengage while moving. A bore hole can optionally be created, which helps when mounting the system on external components, such as shafts. In 3D, you can also create a hub on both the top and bottom sides of the sprocket.



*Figure 2-25: Sprocket geometry in 2D and 3D.*

## *Geometries of Roller Chain Sprocket Assembly*

---

The geometries of the roller chain, sprocket, and roller chain sprocket assembly are as follows:

Geometries in 2D:

- [Roller Chain Geometry \(2D\)](#)
- [Sprocket Geometry \(2D\)](#)
- [Roller Chain Sprocket Assembly Geometry \(2D\)](#)

Geometries in 3D:

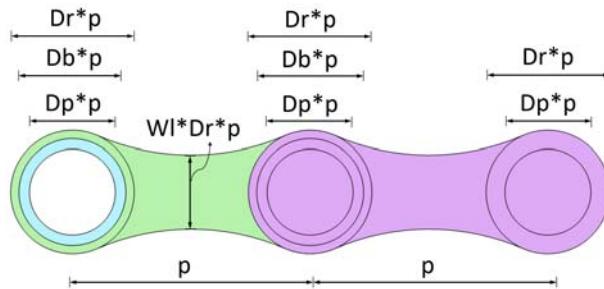
- [Roller Chain Geometry \(3D\)](#)
- [Sprocket Geometry \(3D\)](#)
- [Roller Chain Sprocket Assembly Geometry \(3D\)](#)

### *Roller Chain Geometry (2D)*

---

In 2D, the part name for the roller chain geometry is `roller_chain_2d`.

The sketch in [Figure 2-26](#) shows all input parameters of the chain that can be used to customize the geometry part.



*Figure 2-26: Sketch of the roller chain geometry in 2D.*

## PARAMETERS DESCRIPTION

Table 2-3 shows the description for all important input parameters of the `roller_chain_2d` geometry part.

TABLE 2-3: ROLLER CHAIN (2D) PARAMETERS

NAME	DESCRIPTION
P	Pitch
n	Number of chain links
Dr	Roller diameter to pitch ratio
Dp	Pin diameter to pitch ratio
Db	Bushing diameter to pitch ratio
WI	Minimum link plate width to roller diameter ratio

## OPTIONAL FEATURES

The following geometry operation is optional for this part:

- Bushing between the roller and pin plate. Set Db to zero to remove the bushings.

## SELECTIONS

Table 2-4 describes the selections available for this part

TABLE 2-4: ROLLER CHAIN (2D) SELECTIONS

NAME	TYPE OF SELECTION	DESCRIPTION
Roller Plates (Object)	Object Selection	All roller plate objects
Pin Plates (Object)	Object Selection	All pin plate objects
Links	Domain Selection	All domains
Roller Plates	Domain Selection	Roller plate domains
Bushings	Domain Selection	Bushing domains
Pin Plates	Domain Selection	Pin plates domains
Roller Outer Boundaries	Boundary Selection	Outer boundaries of roller plates
Roller Pin Outer Boundaries	Boundary Selection	Union of outer boundaries of roller plates and pin plates
Roller Inner Boundaries	Boundary Selection	Inner boundaries of roller plates
Pin Inner Boundaries	Boundary Selection	Inner boundaries of pin plates

## Sprocket Geometry (2D)

---

In 2D, the part name for the sprocket geometry is `sprocket_2d`.

The sketch [Figure 2-27](#) shows all input parameters of the sprocket that can be used to customize the geometry part.

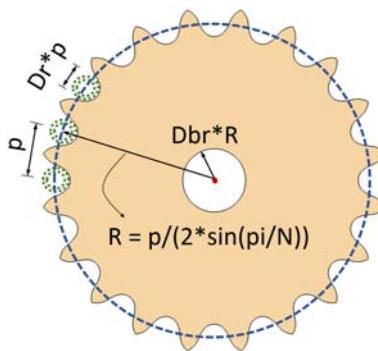


Figure 2-27: Sketch of the sprocket geometry in 2D.

### PARAMETERS DESCRIPTION

[Table 2-5](#) shows the description for all important input parameters of the `sprocket_2d` geometry part

TABLE 2-5: SPROCKET (2D) PARAMETERS

NAME	DESCRIPTION
P	Pitch
N	Number of teeth
Dr	Roller diameter to pitch ratio
Dbr	Bore diameter to pitch diameter ratio

### OPTIONAL FEATURES

The following geometry operation is optional in this part:

- Bore hole in the sprocket. Set `Dbr` to zero to remove the bore hole.

### SELECTIONS

[Table 2-5](#) describes the selections available for this part.

TABLE 2-6: SPROCKET (2D) SELECTIONS

NAME	TYPE OF SELECTION	DESCRIPTION
Sprocket (Object)	Object Selection	Sprocket object
Sprocket	Domain Selection	Sprocket domain
Sprocket Outer Boundaries	Boundary Selection	Outer boundaries of sprocket
Sprocket Inner Boundaries	Boundary Selection	Inner boundaries of sprocket

### *Roller Chain Sprocket Assembly Geometry (2D)*

in 2D, the part name for roller chain sprocket assembly geometry is `roller_chain_sprocket_assembly_2d`.

The sketch in Figure 2-28 shows all input parameters of the roller chain sprocket assembly that can be used to customize the geometry part.

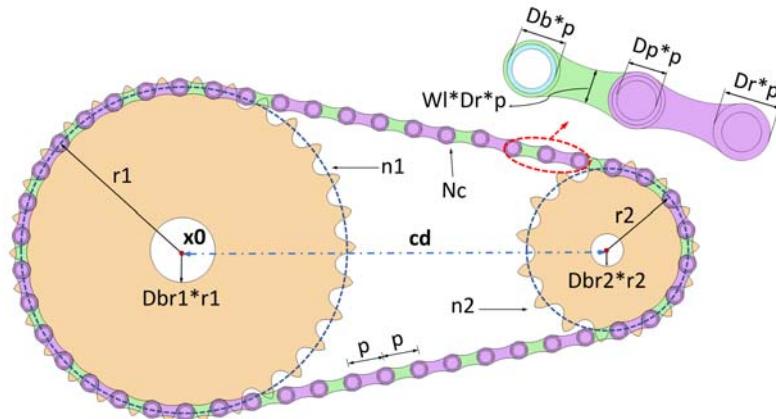


Figure 2-28: Sketch of the roller chain sprocket assembly geometry in 2D.

## PARAMETERS DESCRIPTION

[Table 2-7](#) shows the description of important input parameters for the assembly defined by `roller_chain_sprocket_assembly_2d`. For the input parameters related to the chain and the sprocket, see [Table 2-3](#) and [Table 2-5](#).

TABLE 2-7: ROLLER CHAIN SPROCKET ASSEMBLY (2D) PARAMETERS

NAME	DESCRIPTION
<code>n1</code>	Number of teeth, first sprocket
<code>n2</code>	Number of teeth, second sprocket
<code>cd</code>	Sprocket center distance
<code>Nc</code>	Number of chain links
<code>(x0, y0)</code>	First sprocket center coordinates

## PART VARIANTS

There are two variants available for the `roller_chain_sprocket_assembly_2d` geometry part:

- Specify sprocket center distance
- Specify number of chain links

If you want to create a roller chain sprocket assembly by specifying the center distance between two sprockets, use the first variant of the part. In this case, COMSOL builds the geometry by calculating the minimum even number of links required to wrap the two sprockets placed at the given distance.

If you input instead want the provide the number of chain links, use the second variant of the part. For this variant, COMSOL adjusts the position of the second sprocket, such that the given number of links perfectly wraps around the two sprockets.

Note that in both these cases, the computation of the sprocket center distance and the number of chain links are based on nonlinear equations. Hence, there is a chance that the links may not tightly wrap around the sprockets. If you want to make minor adjustments in center distance between the sprockets, set the center distance correction parameter `ccorr` to one, and tune the value of center distance correction factor `cdelta` accordingly.

## SELECTIONS

[Table 2-8](#) describes the selections available for this part.

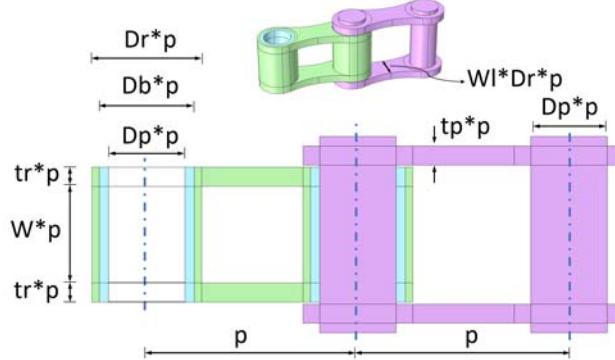
TABLE 2-8: ROLLER CHAIN SPROCKET ASSEMBLY (2D) SELECTIONS

NAME	TYPE OF SELECTION	DESCRIPTION
Roller Plates (Object)	Object Selection	All roller plate objects
Pin Plates (Object)	Object Selection	All pin plate objects
First Sprocket (Object)	Object Selection	First sprocket object
Second Sprocket (Object)	Object Selection	Second sprocket objects
Links	Domain Selection	All domains
Roller Plates	Domain Selection	Roller plate domains
Bushings	Domain Selection	Bushing domains
Pin Plates	Domain Selection	Pin plates domains
First Sprocket	Domain Selection	First sprocket domain
Second Sprocket	Domain Selection	Second sprocket domain
Sprockets	Domain Selection	Union of first and second sprocket domains
Roller Outer Boundaries	Boundary Selection	Outer boundaries of roller plates
Roller Pin Outer Boundaries	Boundary Selection	Union of outer boundaries of roller plates and pin plates
Roller Inner Boundaries	Boundary Selection	Inner boundaries of roller plates
Pin Inner Boundaries	Boundary Selection	Inner boundaries of pin plates
First Sprocket Outer Boundaries	Boundary Selection	Outer boundaries of first sprocket
First Sprocket Inner Boundaries	Boundary Selection	Inner boundaries of first sprocket
Second Sprocket Outer Boundaries	Boundary Selection	Outer boundaries of second sprocket
Sprocket Inner Boundaries	Boundary Selection	Inner boundaries of second sprocket
Sprocket Outer Boundaries	Boundary Selection	Outer boundaries of both sprockets
Sprocket Inner Boundaries	Boundary Selection	Inner boundaries of both sprockets

## Roller Chain Geometry (3D)

In 3D, the part name for the roller chain geometry is `roller_chain`.

The sketch in [Figure 2-29](#) shows all input parameters of the chain that can be used to customize the geometry part.



*Figure 2-29: Sketch of the roller chain geometry in 3D and its cross-section.*

### PARAMETERS DESCRIPTION

[Table 2-9](#) shows the descriptions for all important input parameters of the `roller_chain` geometry part.

TABLE 2-9: ROLLER CHAIN PARAMETERS

NAME	DESCRIPTION
$p$	Pitch
$n$	Number of chain links
$W$	Sprocket width to pitch ratio
$Dr$	Roller diameter to pitch ratio
$Dp$	Pin diameter to pitch ratio
$Db$	Bushing diameter to pitch ratio
$Wl$	Minimum link plate width to roller diameter ratio
$tr$	Thickness to pitch ratio, roller plate
$tp$	Thickness to pitch ratio, pin plate

### OPTIONAL FEATURES

The following geometry operation is optional in this part:

- Bushing between roller and pin plate. Set  $Db$  to zero to remove the bushings.

## SELECTIONS

Table 2-10 describes the selections available for this part.

TABLE 2-10: ROLLER CHAIN SELECTIONS

NAME	TYPE OF SELECTION	DESCRIPTION
Roller Plates (Object)	Object Selection	All roller plate objects
Pin Plates (Object)	Object Selection	All pin plate objects
Links	Domain Selection	All domains
Roller Plates	Domain Selection	Roller plate domains
Bushings	Domain Selection	Bushing domains
Pin Plates	Domain Selection	Pin plates domains
Roller Outer Boundaries	Boundary Selection	Outer cylindrical surfaces of roller plates
Roller Inner Boundaries	Boundary Selection	Inner cylindrical surfaces of roller plates
Pin Outer Boundaries	Boundary Selection	Outer cylindrical surfaces of pin plates

## Sprocket Geometry (3D)

In 3D, the part name for the sprocket geometry is *sprocket*.

The sketch in Figure 2-30 shows all input parameters of the sprocket that can be used to customize the available geometry.

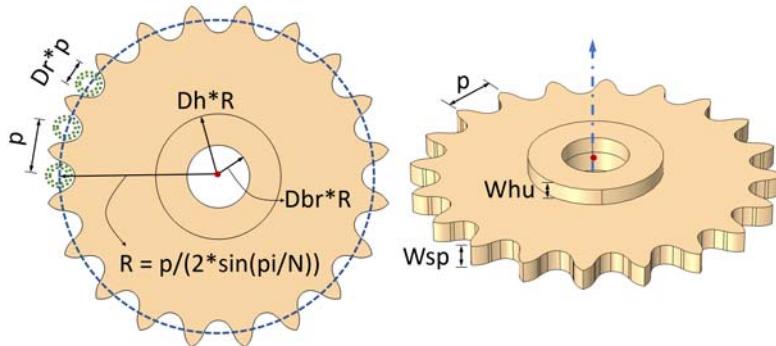


Figure 2-30: Sketch of the sprocket geometry in 3D.

### PARAMETERS DESCRIPTION

Table 2-11 shows the description for all important input parameters of the sprocket geometry part.

TABLE 2-11: SPROCKET PARAMETERS

NAME	DESCRIPTION
P	Pitch
N	Number of teeth
Wsp	Sprocket width to pitch ratio
Dr	Roller diameter to pitch ratio
Dbr	Bore diameter to pitch diameter ratio
Dh	Hub diameter to pitch diameter ratio
Whu	Hub width to pitch ratio, upside
Whd	Hub width to pitch ratio, downside

### OPTIONAL FEATURES

The following geometry operations are optional in this part:

- Bore hole in the sprocket. Set Dbr to zero to remove the bore hole.
- Hub on the upside and downside of the sprocket. Set Dh to zero to remove the hubs.

### SELECTIONS

Table 2-12 describes the selections available for this part.

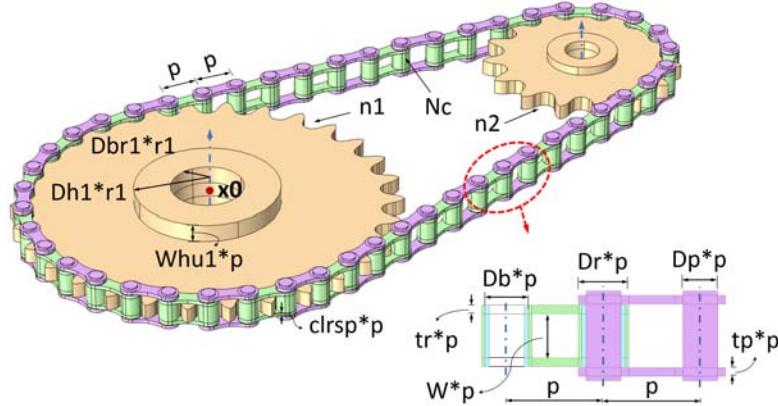
TABLE 2-12: SPROCKET SELECTIONS

NAME	TYPE OF SELECTION	DESCRIPTION
Sprocket (Object)	Object Selection	Sprocket object
Sprocket	Domain Selection	Sprocket domain
Sprocket Outer Boundaries	Boundary Selection	Outer side surfaces of sprocket
Sprocket Inner Boundaries	Boundary Selection	Inner surfaces of sprocket

### *Roller Chain Sprocket Assembly Geometry (3D)*

In 3D, the part name for roller chain sprocket assembly geometry is `roller_chain_sprocket_assembly`.

The sketch in [Figure 2-31](#) shows all input parameters of the roller chain sprocket assembly that can be used to customize the geometry part.



*Figure 2-31: Sketch of the roller chain sprocket assembly geometry in 3D.*

#### PARAMETERS DESCRIPTION

[Table 2-13](#) shows the description of all important input parameters of the `roller_chain_sprocket_assembly` geometry part. For the input parameters related to the chain and the sprocket, see [Table 2-9](#) and [Table 2-11](#).

**TABLE 2-13: ROLLER CHAIN SPROCKET ASSEMBLY PARAMETERS**

NAME	DESCRIPTION
<code>n1</code>	Number of teeth, first sprocket
<code>n2</code>	Number of teeth, second sprocket
<code>cd</code>	Sprocket center distance
<code>Nc</code>	Number of chain links
<code>(x0, y0, z0)</code>	First sprocket center coordinates
<code>(esx, esy, esz)</code>	Sprocket axis direction
<code>clrsp</code>	Sprocket clearance

#### PART VARIANTS

There are two variants available for `roller_chain_sprocket_assembly` geometry part:

- Specify sprocket center distance
- Specify number of chain links

If you want to create a chain sprocket assembly by specifying the center distance between two sprockets, use the first variant of the part. In this case, COMSOL builds the geometry by calculating the minimum even number of links required to wrap the two sprockets placed at the given distance.

If you input instead want the provide the number of chain links, use the second variant of the part. For this variant, COMSOL adjusts the position of the second sprocket, such that the given number of links perfectly wraps around the two sprockets.

Note that in both these cases, the computation of the sprocket center distance and the number of chain links are based on nonlinear equations. Hence, there is a chance that the links may not tightly wrap around the sprockets. If you want to make minor adjustments in center distance between the sprockets, set the center distance correction parameter `ccorr` to one, and tune the value of center distance correction factor `cdelta` accordingly.

## SELECTIONS

Table 2-14 describes the selections available for this part.

TABLE 2-14: ROLLER CHAIN SPROCKET ASSEMBLY SELECTIONS

NAME	TYPE OF SELECTION	DESCRIPTION
Roller Plates (Object)	Object Selection	All roller plate objects
Pin Plates (Object)	Object Selection	All pin plate objects
First Sprocket (Object)	Object Selection	First sprocket object
Second Sprocket (Object)	Object Selection	Second sprocket objects
Links	Domain Selection	All domains
Roller Plates	Domain Selection	Roller plate domains
Bushings	Domain Selection	Bushing domains
Pin Plates	Domain Selection	Pin plates domains
First Sprocket	Domain Selection	First sprocket domain
Second Sprocket	Domain Selection	Second sprocket domain
Sprockets	Domain Selection	Union of first and second sprocket domains
Roller Outer Boundaries	Boundary Selection	Outer cylindrical surfaces of roller plates
Roller Inner Boundaries	Boundary Selection	Inner cylindrical surfaces of roller plates
Pin Outer Boundaries	Boundary Selection	Outer cylindrical surfaces of pin plates

TABLE 2-14: ROLLER CHAIN SPROCKET ASSEMBLY SELECTIONS

NAME	TYPE OF SELECTION	DESCRIPTION
First Sprocket Outer Boundaries	Boundary Selection	Outer boundaries of first sprocket
First Sprocket Inner Boundaries	Boundary Selection	Inner boundaries of first sprocket
Second Sprocket Outer Boundaries	Boundary Selection	Outer boundaries of second sprocket
Sprocket Inner Boundaries	Boundary Selection	Inner boundaries of second sprocket
Sprocket Outer Boundaries	Boundary Selection	Outer boundaries of both sprockets
Sprocket Inner Boundaries	Boundary Selection	Inner boundaries of both sprockets

# Bearings

Bearings are common in multibody systems, since they are necessary to support the relative rotation between two components. Depending on the relative rotational speed and magnitude of the load transfer between the components, different types of bearings are used, for example, rolling element bearings and fluid film bearings.

Roller bearings are often used for low speed applications where noise is not an important consideration. However, roller bearings perform poorly for impact loads. These bearings have a limited life, but can be easily replaced due to their low cost. In particular, misalignment can decrease their life expectancy significantly. Due to poor damping characteristics, these bearings are sometimes used with squeeze film dampers.

Fluid film bearings overcome all these problems, but they are expensive to maintain. A complex lubrication system is needed to maintain the film thickness in the bearings and to avoid them from starving. Fluid film bearings also provide a good stability to unbalance excitation. However, fluid film bearings cannot be used for low speed applications. Other limitations of the fluid film bearings include requirement of a higher starting torque, high power loss, and temperature sensitivity.

One of the key criteria for selecting bearings is the combination of speed and load carrying capacity. The load carrying capacity of a fluid film bearing increases with the speed, while the opposite is true for rolling element bearings. Therefore, for a high load at high speeds, fluid film bearings are preferred.

## *Bearing Modeling*

---

A number of approaches exist for modeling the bearings in an assembly. The simplest approach is to use equivalent stiffness and damping constants for the bearings in the system model. In this case, dynamic characteristics of the bearings are obtained either by experiments or by simulations. These characteristics then can directly be used in a multibody system model. In the Multibody Dynamics interface, you can use an elastic hinge joint and provide the bearing stiffness as an input on the **Joint Elasticity** node to model the bearings using this approach.

A more complex approach is to combine a bearing simulation together with the multibody simulation. It is possible to perform such simulations in Multibody Dynamics Module by using functionality from the Rotordynamics Module. The

following sections describe the modeling of radial roller bearings and hydrodynamic bearings together in a multibody dynamics context.

### **RADIAL ROLLER BEARING**

A **Radial Roller Bearing** node can be added in the Multibody Dynamics interface when you have a license also for the Rotordynamics Module. You select boundaries in the multibody system where the inner race of the bearing is connected. The following bearing types are available:

- Deep Groove Ball Bearing
- Angular Contact Ball Bearing
- Self-Aligning Ball Bearing
- Spherical Roller Bearing
- Cylindrical Roller Bearing
- Tapered Roller Bearing

Depending on the bearing type, you need to input the respective geometric and materials properties of the bearing components.

### **HYDRODYNAMIC BEARING**

Modeling of hydrodynamic bearings in multibody requires a coupling to the Hydrodynamic Bearing interface in the Rotordynamics Module. When both a Multibody Dynamics interface and a Hydrodynamic Bearing interface are present in the model, a multiphysics coupling named **Solid-Bearing Coupling** can be added under the **Multiphysics** node in the Model Builder. Use this feature to combine the multibody and fluid film simulations. Add a **Hydrodynamic Journal Bearing** node in the Hydrodynamic Bearing interface to simulate different types of Journal Bearings. The following bearing types are available:

- Plain Journal Bearing
- Elliptic Journal Bearing
- Offset Halves Journal Bearing
- Multilobe Journal Bearing
- Tilted Pad Journal Bearing

You can also model a floating ring journal bearing using the **Floating Ring Bearing** node in the Hydrodynamic Bearing interface.

# Initialization in Time-Dependent Studies

A time-dependent analysis must have appropriate initial conditions. If the analysis starts with nonzero initial conditions, you would in principle need to derive and input all components of the initial values for all parts of the structure. In a complex assembly, this may require some effort.

In a time-dependent analysis, there is a step where consistent initial values can be computed. This step ensures, by modifying the given initial conditions, that the initial values actually used are matching everywhere. The new initial values can be seen as a type of average of the ones you entered, if they were not consistent.

In a multibody dynamics analysis, it is common that you know the initial conditions for a certain component, and want those initial values to be exactly fulfilled. The initial values for other parts should then be computed so that they match the ones prescribed. The initial value settings in the Multibody Dynamics Module gives you an option for doing so. The computation of consistent initial values is still performed, but the values you have marked by setting the control **Consistent initialization** to **Force initial values** are respected.

If this option is used, the degrees of freedom for which **Force initial values** is selected are treated as having prescribed displacements or rotations during the initialization step.

Even if the initial values of certain components are forced in a multibody system containing flexible bodies, it is possible that the system is not initialized with the known (forced) initial values because of the deformation of the elastic components during the consistent initialization step. To initialize the system with the known initial values like a rigid body system, elastic bodies can be assumed rigid during the consistent initialization step. You can do that by adding an **Initially rigid** node, where you select the flexible domains which should be treated as rigid during the initialization.

# Geometric Nonlinearity

In most cases, the parts in a multibody analysis will translate and rotate significantly during the analysis. This means that the analysis must be considered as geometrically nonlinear.

You enable a geometrically nonlinear analysis for a certain study step by selecting the **Include geometric nonlinearity** check box in the **Study Settings** section of a study step.

If any active feature in the model requires the analysis to be geometrically nonlinear, the **Include geometric nonlinearity** check box is selected automatically, and it cannot be cleared. For **Stationary** or **Time Dependent** study steps, the following physics features will force the use of geometric nonlinearity:

- When any **Joint** is used.
- When a nonzero rotation is prescribed in a **Prescribed Displacement/Rotation** subnode of **Rigid Domain**.
- When a nonzero rotation is prescribed in a **Prescribed Displacement/Rotation** subnode of any **Gear**.
- When a nonzero rotation is prescribed for a **Rigid Connector**.
- When a **Cam-Follower** node is used
- When a **Rigid Body Contact** node is used.
- When a **Contact** node is used.

Note that **Contact** node will force the use of geometric nonlinearity for all study types.



---

- [Modeling Geometric Nonlinearity](#)
- [Joints](#)
- [Contact Modeling](#)

For other study step types, you can choose whether to include the effect of geometric nonlinearity. For example, when computing the eigenfrequencies of a structure using an **Eigenfrequency** study step, the problem is geometrically linear, but in the case of a pre-stressed eigenfrequency analysis, effects of geometric nonlinearity have to be included.

# Multiphysics Couplings

It is possible to couple the Multibody Dynamics interface to several other physics interfaces through predefined multiphysics couplings. This section describes the available couplings.

When you add a physics interface for which a predefined coupling is available to the component, a **Multiphysics Couplings** node appears in the model tree. Under the **Multiphysics Couplings** node, you can add the required couplings.

TABLE 2-15: AVAILABLE MULTIPHYSICS COUPLINGS FOR THE MULTIBODY DYNAMICS INTERFACE

COUPLING	INTERFACES	COMMENT
Thermal Expansion	Heat Transfer Heat Transfer in Solids Heat Transfer in Porous Media Bioheat Transfer	One-way domain coupling
Acoustic-Structure Boundary	Pressure Acoustics, Frequency Domain Pressure Acoustics, Time Domain	Two-way boundary coupling
Thermoviscous Acoustics-Structure Boundary	Thermoviscous Acoustics, Frequency Domain	Two-way boundary coupling
Lumped-Structure Connection	Lumped Mechanical System	Two-way point coupling
Fluid-Structure Interaction	Any single-phase flow interface	One-way or two-way boundary coupling
Fluid-Structure Interaction Pair	Any single-phase flow interface	Two-way boundary coupling
Solid-Bearing Coupling	Hydrodynamic Bearing	Two-way boundary coupling

There are two multiphysics interfaces which are specific to the Multibody Dynamics Module:

- [The Fluid-Multibody Interaction Interface](#)
- [The Fluid-Multibody Interaction, Assembly Interface](#)

---

The other couplings are analogous to those available for the Solid Mechanics interface. See also:

- [Defining Multiphysics Models](#) in the *Structural Mechanics Module User's Guide*
- [Thermally Coupled Problems](#) in the *Structural Mechanics Module User's Guide*
- [Fluid-Structure Interaction](#) in the *Structural Mechanics Module User's Guide*
- More details about Solid-Bearing Coupling in the *Rotordynamics Module User's Guide*



### *The Fluid-Multibody Interaction Interface*

---

Using the **Fluid-Multibody Interaction** (fluid and solid icon) interface, you can model phenomena where a fluid and a rigid or deformable solid affect each other. The interface includes a predefined condition for the interaction at the fluid-solid boundaries. [Arbitrary Lagrangian-Eulerian Formulation \(ALE\)](#) is used for incorporating the geometrical changes of the fluid domain.

Use this interface if the geometry sequence is set up to form a union. In multibody modeling, it is more common that the assembly mode is used, in which case you should use [The Fluid-Multibody Interaction, Assembly Interface](#).

Using a stationary or a time-dependent study, the **Fluid-Multibody Interaction** interface models two-way coupling between solids and fluids.

You add a predefined **Fluid-Multibody Interaction** interface from the **Fluid-Structure Interaction** (fluid and solid icon) group in the **Fluid Flow** branch (fluid icon) of the **Model Wizard** or **Add Physics** windows. **Laminar Flow** and **Multibody Dynamics** interfaces are then added to the Model Builder.

In addition, the **Multiphysics Couplings** node is added, which automatically includes the multiphysics coupling feature **Fluid-Structure Interaction**.

At the same time, a **Moving Mesh** node with a **Deforming Domain** subnode is created under **Definitions**. For more information, see [Deforming Fluid Domains](#).

#### *On the Constituent Physics Interfaces*

- The Laminar Flow interface is described in [The Single-Phase Flow, Laminar Flow Interface](#). When the CFD Module is available, turbulent flow is supported through the inclusion of a number of turbulence models.
- The Multibody Dynamics interface is described in [The Multibody Dynamics Interface](#).
- The Deforming Domain feature is described in the [Moving Mesh Features](#).

#### **SETTINGS FOR PHYSICS INTERFACES AND COUPLING FEATURES**

When physics interfaces are added using the predefined couplings, all specific settings are included with the physics interfaces and the coupling features.

However, if physics interfaces are added one at a time, followed by the coupling features, these modified settings are not automatically included.

For example, if single Laminar Flow and Multibody Dynamics interfaces are added, COMSOL Multiphysics adds an empty **Multiphysics** node. You can choose from the available coupling features, but the modified settings are not included.

Another difference is that the Laminar Flow interface added by the predefined coupling employs pseudo-time stepping per default for stationary models, while the standalone version uses an automatic Newton damping. The latter can sometimes give shorter solution times, while the pseudo-time stepping often gives a more robust solution procedure. You can disable the pseudo-time stepping in **Advanced Settings** section for the fluid interface.



Coupling features are available from the context menu (right-click the **Multiphysics** node) or from the **Physics** toolbar, **Multiphysics Couplings** menu.

#### **PHYSICS INTERFACES AND COUPLING FEATURES**



Use the online help in COMSOL Multiphysics to locate and search all the documentation. All these links also work directly in COMSOL Multiphysics when using the Help system.

## Coupling Features

The **Fluid-Structure Interaction** multiphysics coupling is used to couple the interfaces.

## Physics Interface Features

Physics nodes are available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or right-click to access the context menu (all users).



In general, to add a node, go to the **Physics** toolbar, no matter what operating system you are using. Subnodes are available by clicking the parent node and selecting it from the **Attributes** menu.

- The available physics features for **The Single-Phase Flow**, **Laminar Flow Interface** are listed in the section **Domain, Boundary, Pair, and Point Nodes for Single-Phase Flow** in the *COMSOL Multiphysics Reference Manual*.
- The available physics features for **The Multibody Dynamics Interface** are listed in the **Domain, Boundary, Edge, Point, and Pair Nodes for the Multibody Dynamics Interface** section.

## *The Fluid-Multibody Interaction, Assembly Interface*

Using the **Fluid-Multibody Interaction, Assembly** ( ) interface, you can model phenomena where a fluid and a rigid or deformable solid affect each other. The interface includes a predefined condition for the interaction at the fluid-solid boundaries. This interface is used when the geometry sequence is set up to form an assembly. **Arbitrary Lagrangian-Eulerian Formulation (ALE)** is used for incorporating the geometrical changes of the fluid domain.

Using a stationary or a time-dependent study, the **Fluid-Solid Interaction, Assembly** interface models two-way coupling between solids and fluids.

You add a predefined **Fluid-Multibody Interaction, Assembly** interface from the **Fluid-Structure Interaction** ( ) group in the **Fluid Flow** branch ( ) of the **Model Wizard** or **Add Physics** windows. **Laminar Flow** and **Multibody Dynamics** interfaces are then added to the Model Builder.

In addition, the **Multiphysics Couplings** node is added, which automatically includes the multiphysics coupling feature **Fluid-Structure Interaction, Pair**.

At the same time, a **Moving Mesh** node with a **Deforming Domain** subnode is created under **Definitions**. For more information, see **Deforming Fluid Domains**.

### *On the Constituent Physics Interfaces*

- The Laminar Flow interface is described in [The Single-Phase Flow, Laminar Flow Interface](#). When the CFD Module is available, turbulent flow is supported through the inclusion of a number of turbulence models.
- The Multibody Dynamics interface is described in [The Multibody Dynamics Interface](#).
- The Deforming Domain feature is described in the [Moving Mesh Features](#).

### **SETTINGS FOR PHYSICS INTERFACES AND COUPLING FEATURES**

When physics interfaces are added using the predefined couplings, all specific settings are included with the physics interfaces and the coupling features.

However, if physics interfaces are added one at a time, followed by the coupling features, these modified settings are not automatically included.

For example, if single Laminar Flow and Multibody Dynamics interfaces are added, COMSOL Multiphysics adds an empty **Multiphysics** node. You can choose from the available coupling features, but the modified settings are not included.

Another difference is that the Laminar Flow interface added by the predefined coupling employs pseudo-time stepping per default for stationary models, while the standalone version uses an automatic Newton damping. The latter can sometimes give shorter solution times, while the pseudo-time stepping often gives a more robust solution procedure. You can disable the pseudo-time stepping in **Advanced Settings** section for the fluid interface.



Coupling features are available from the context menu (right-click the **Multiphysics** node) or from the **Physics** toolbar, **Multiphysics Couplings** menu.

---

### **PHYSICS INTERFACES AND COUPLING FEATURES**

---



Use the online help in COMSOL Multiphysics to locate and search all the documentation. All these links also work directly in COMSOL Multiphysics when using the Help system.

#### *Coupling Features*

The [Fluid-Structure Interaction, Pair](#) multiphysics coupling is used to couple the interfaces.

### Physics Interface Features

Physics nodes are available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or right-click to access the context menu (all users).



In general, to add a node, go to the **Physics** toolbar, no matter what operating system you are using. Subnodes are available by clicking the parent node and selecting it from the **Attributes** menu.

- The available physics features for [The Single-Phase Flow](#), [Laminar Flow Interface](#) are listed in the section [Domain, Boundary, Pair, and Point Nodes for Single-Phase Flow](#) in the *COMSOL Multiphysics Reference Manual*.
- The available physics features for [The Multibody Dynamics Interface](#) are listed in the [Domain, Boundary, Edge, Point, and Pair Nodes for the Multibody Dynamics Interface](#) section.



## Multibody Dynamics Theory

This chapter introduces you to the theory which is specific for the Multibody Dynamics interface features.

In this chapter:

- [Joint Theory](#)
- [Cam-Follower Theory](#)
- [Gear Theory](#)
- [Gear Pair Theory](#)
- [Rigid Body Contact Theory](#)

# Joint Theory

In this section:

- Representation of Rotations
- Prismatic Joint Theory
- Hinge Joint Theory
- Cylindrical Joint Theory
- Screw Joint Theory
- Planar Joint Theory
- Ball Joint Theory
- Slot Joint Theory
- Reduced Slot Joint Theory
- Clearance Joint Theory
- Fixed Joint Theory
- Distance Joint Theory
- Universal Joint Theory
- Theory for Joint Constraints
- Theory for Joint Locking
- Theory for Springs and Dampers on Joints
- Theory for Prescribed Motion on Joints
- Theory for Applied Forces and Moments on Joints
- Theory for Friction in Joints
- Theory for Joint Forces and Moments Computation

## *Representation of Rotations*

---

In 2D models, the rotation can only occur around the  $z$ -axis. It is represented by a scalar angle,  $\theta$ .

In 3D the situation is more complex. There are three rotational degrees of freedom. For finite rotations, however, any choice of three rotation parameters is singular at some specific set of angles. For this reason, a four-parameter *quaternion* representation is used for the rotations in the joints. The quaternion parameters are called  $a$ ,  $b$ ,  $c$ , and  $d$ . These four parameters are not independent, so an extra equation stating that the following relation is added:

$$a^2 + b^2 + c^2 + d^2 = 1$$

The four quaternion parameters are often represented as one scalar and one vector,

$$\mathbf{q} = \begin{bmatrix} a \\ \mathbf{b} \end{bmatrix}$$

where

$$\mathbf{b} = \begin{bmatrix} b \\ c \\ d \end{bmatrix}$$

The connection between the quaternion parameters and a rotation matrix  $R$  is

$$R = \begin{bmatrix} a^2 + b^2 - c^2 - d^2 & 2bc - 2ad & 2ac + 2bd \\ 2ad + 2bc & a^2 - b^2 + c^2 - d^2 & 2cd - 2ab \\ 2bd - 2ac & 2ab + 2cd & a^2 - b^2 - c^2 + d^2 \end{bmatrix}$$

The rotation can be also represented as a rotation vector with direction  $\mathbf{e}_\Theta$  and a magnitude,  $\Theta$ .

$$\Theta = \Theta \mathbf{e}_\Theta$$

The magnitude of the rotation is

$$\Theta = 2 \arccos(a)$$

and the direction is

$$\mathbf{e}_\Theta = \frac{\mathbf{b}}{\|\mathbf{b}\|}$$

The parameter  $a$  can thus be considered as measuring the rotation, while  $b$ ,  $c$ , and  $d$  can be interpreted as the orientation of the rotation vector. For small rotations, this relation simplifies to

$$\Theta = 2 \begin{bmatrix} b \\ c \\ d \end{bmatrix}$$

Successive rotations are represented by a quaternion multiplication

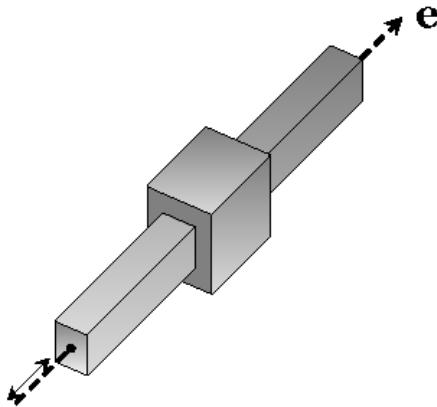
$$\begin{bmatrix} a_{\text{tot}} \\ \mathbf{b}_{\text{tot}} \end{bmatrix} = \begin{bmatrix} a_1 \\ \mathbf{b}_1 \end{bmatrix} \otimes \begin{bmatrix} a_2 \\ \mathbf{b}_2 \end{bmatrix} = \begin{bmatrix} a_1 a_2 - \mathbf{b}_1 \cdot \mathbf{b}_2 \\ a_1 \mathbf{b}_2 + a_2 \mathbf{b}_1 + \mathbf{b}_1 \times \mathbf{b}_2 \end{bmatrix}$$

Just as finite rotations, quaternion multiplications are not commutative.

## *Prismatic Joint Theory*

---

The **Prismatic Joint**, also known as a translational or sliding joint, has one translational degree of freedom between the two connected components. The components are free to translate, or slide, relative to each other along the joint axis.



### **JOINT AXIS AND ITS LOCAL COORDINATE SYSTEM**

The local coordinate system of a joint, which can be seen as rigidly connected to the source attachment, consists of three axes: joint axis ( $\mathbf{e}_1$ ), second axis ( $\mathbf{e}_2$ ), and third axis ( $\mathbf{e}_3$ ). You specify the initial direction joint axis ( $\mathbf{e}_{10}$ ), using one of the following methods:

- **Method 1:** Give the components explicitly in a selected coordinate system.
- **Method 2:** Select one of the axes of a selected coordinate system.
- **Method 3:** Select an edge that is parallel to the joint axis.



For all methods, the axis is normalized to give a unit vector.

---

In Method 3, the direction that has a positive projection on the global  $x$ -axis is selected. If the projection on the  $x$ -axis is very small (the edge is almost perpendicular to the  $x$ -axis), then the  $y$ -axis is used in the projection criterion. If the edge is perpendicular also to the  $y$ -axis, then the  $z$ -axis is used as the joint axis. You can also choose to reverse the direction of initial joint axis.

If the directions of the second and third axis are important (for example, when interpreting joint force results in the joint local coordinate system, or when there is elasticity in the joint), then Method 2 is preferred because this gives full control over all three axes. When an axis in the selected coordinate system is chosen to specify the joint axis, the other two axes are determined through cyclic permutation.

For Method 2 and Method 3, the initial second axis and initial third axis are computed internally as described below.

The initial second axis ( $\mathbf{e}_{20}$ ) (perpendicular to the initial joint axis) is computed by making an *auxiliary axis* ( $\mathbf{e}_{\text{aux}}$ ) (an axis not parallel to the initial joint axis) and then taking a cross product of the initial joint axis and this auxiliary axis:

$$\mathbf{e}_{20} = \mathbf{e}_{10} \times \mathbf{e}_{\text{aux}}$$

The initial third axis ( $\mathbf{e}_{30}$ ) is computed by taking a cross product of the initial joint axis and the initial second axis:

$$\mathbf{e}_{30} = \mathbf{e}_{10} \times \mathbf{e}_{20}$$

Finally, the rotated axes in the deformed configuration are formed by rotating the individual axes using the rotation matrix of the source attachment ( $R_{\text{src}}$ ):

$$\begin{aligned}\mathbf{e}_1 &= R_{\text{src}} \cdot \mathbf{e}_{10} \\ \mathbf{e}_2 &= R_{\text{src}} \cdot \mathbf{e}_{20} \\ \mathbf{e}_3 &= R_{\text{src}} \cdot \mathbf{e}_{30}\end{aligned}$$

For 2D models, the generation of the coordinate system is significantly simplified. The initial third axis ( $\mathbf{e}_{30}$ ) is taken as the out-of-plane axis ( $\mathbf{e}_z$ ). The initial second axis ( $\mathbf{e}_{20}$ ) is orthogonal to ( $\mathbf{e}_{10}$ ) and ( $\mathbf{e}_{30}$ ):

$$\begin{aligned}\mathbf{e}_{20} &= \mathbf{e}_z \times \mathbf{e}_{10} \\ \mathbf{e}_{30} &= \mathbf{e}_z\end{aligned}$$

#### PRISMATIC JOINT FORMULATION

For a prismatic joint, the destination attachment is free to translate relative to the source attachment along the joint axis. The added degree of freedom is thus the relative displacement along the joint axis ( $u$ ).

The translational motion of the source and destination attachments at the center of the joint can be computed as:

$$\begin{aligned}\mathbf{u}_{c, \text{src}} &= (\mathbf{R}_{\text{src}} - \mathbf{I}) \cdot (\mathbf{X}_c - \mathbf{X}_{c, \text{src}}) + \mathbf{u}_{\text{src}} \\ \mathbf{u}_{c, \text{dst}} &= (\mathbf{R}_{\text{dst}} - \mathbf{I}) \cdot (\mathbf{X}_c - \mathbf{X}_{c, \text{dst}}) + \mathbf{u}_{\text{dst}}\end{aligned}$$

where, for the source and destination attachments, respectively:

- $\mathbf{u}_{c, \text{src}}$  and  $\mathbf{u}_{c, \text{dst}}$  are the displacement vectors for the attachments at the center of joint.
- $\mathbf{R}_{\text{src}}$  and  $\mathbf{R}_{\text{dst}}$  are the rotation matrices describing the rotation of each attachment.
- $\mathbf{X}_{c, \text{src}}$  and  $\mathbf{X}_{c, \text{dst}}$  are the positions of the centroids of the attachments.
- $\mathbf{u}_{\text{src}}$  and  $\mathbf{u}_{\text{dst}}$  are the displacements at the centroids of the attachments.
- $\mathbf{X}_c$  is the joint center.

To formulate this kind of connection for a rigid joint, the motion of the destination attachment is prescribed in terms of the motion of the source attachment:

$$\begin{aligned}\mathbf{u}_{c, \text{dst}} &= \mathbf{u}_{c, \text{src}} + \mathbf{u}_r \\ \mathbf{b}_{\text{dst}} &= \mathbf{b}_{\text{src}}\end{aligned}$$

The displacement vector ( $\mathbf{u}$ ) and quaternion vector ( $\mathbf{b}$ ) for the source and destination attachments are defined as:

$$\mathbf{u} = \begin{bmatrix} u \\ v \\ w \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} b \\ c \\ d \end{bmatrix}$$

The relative displacement vector in the global spatial coordinate system ( $\mathbf{u}_r$ ) is defined as:

$$\mathbf{u}_r = u \mathbf{e}_1$$

For 2D, the motion of the destination attachment is prescribed in terms of the motion of source attachment:

$$\begin{aligned}\mathbf{u}_{c, \text{dst}} &= \mathbf{u}_{c, \text{src}} + \mathbf{u}_r \\ \phi_{\text{dest}} &= \phi_{\text{src}}\end{aligned}$$

where  $\phi$  is the rotation about  $z$ -axis (out-of-plane axis).

### *Joint Elasticity in Prismatic Joint*

In the case that the joint is elastic, the elastic displacements and rotations are added to the source motion, so that the displacements and rotations at the center of joint is no longer equal for the source and destination attachments.

$$\begin{aligned}\mathbf{u}_{c, \text{dst}} &= \mathbf{u}_{c, \text{src}} + \mathbf{u}_{\text{el}} + \mathbf{u}_r \\ \mathbf{b}_{\text{dst}} &= \tilde{\mathbf{b}}\end{aligned}$$

$$\begin{bmatrix} \tilde{\mathbf{a}} \\ \tilde{\mathbf{b}} \end{bmatrix} = \begin{bmatrix} \mathbf{a}_{\text{el}} \\ \mathbf{b}_{\text{el}} \end{bmatrix} \otimes \begin{bmatrix} \mathbf{a}_{\text{src}} \\ \mathbf{b}_{\text{src}} \end{bmatrix}$$

The elastic translation and rotation vectors are expressed in terms of the joint axes as

$$\begin{aligned}\mathbf{u}_{\text{el}} &= u_2 \mathbf{e}_2 + u_3 \mathbf{e}_3 \\ \theta_{\text{el}} &= \theta_1 \mathbf{e}_1 + \theta_2 \mathbf{e}_2 + \theta_3 \mathbf{e}_3\end{aligned}$$

Here,  $u_2, u_3, \theta_1, \theta_2$ , and  $\theta_3$  are the default elastic degrees of freedom available for this joint. Optionally, only a subset of them can be chosen to be elastic.

The quaternion representing the elastic rotation is

$$\begin{bmatrix} \mathbf{a}_{\text{el}} \\ \mathbf{b}_{\text{el}} \end{bmatrix}$$

and can be computed from the elastic rotation vector  $\theta_{\text{el}}$ .

The spring and damping forces and moments are proportional to the elastic displacements and velocities.

$$\begin{bmatrix} \mathbf{F}_{s, \text{el}} \\ \mathbf{M}_{s, \text{el}} \end{bmatrix} = \begin{bmatrix} \mathbf{k}_u & \mathbf{k}_{u\theta} \\ \mathbf{k}_{\theta u} & \mathbf{k}_\theta \end{bmatrix} \begin{bmatrix} \mathbf{u}_{\text{el}} \\ \theta_{\text{el}} \end{bmatrix}$$

$$\begin{bmatrix} \mathbf{F}_{d, \text{el}} \\ \mathbf{M}_{d, \text{el}} \end{bmatrix} = \begin{bmatrix} \mathbf{c}_u & \mathbf{c}_{u\theta} \\ \mathbf{c}_{\theta u} & \mathbf{c}_\theta \end{bmatrix} \begin{bmatrix} \dot{\mathbf{u}}_{\text{el}} \\ \dot{\theta}_{\text{el}} \end{bmatrix}$$

The spring and damping matrices are subdivided into four parts depending on whether they relate to translation or rotation. The matrices which couple translation and

rotations are optional in the input data. The total elastic joint forces and moments can be written using the spring and damping forces and moments as

$$\mathbf{F}_{\text{el}} = -\mathbf{F}_{s, \text{el}} - \mathbf{F}_{d, \text{el}}$$
$$\mathbf{M}_{\text{el}} = -\mathbf{M}_{s, \text{el}} - \mathbf{M}_{d, \text{el}}$$

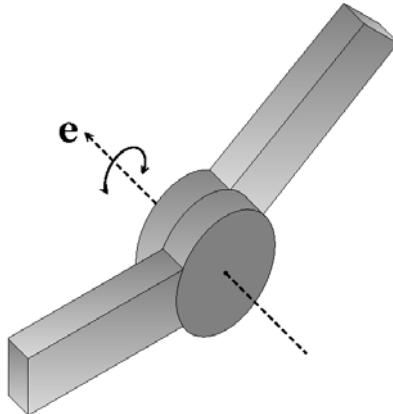
The contribution to the virtual work is

$$\delta W = \mathbf{F}_{\text{el}} \cdot \delta \mathbf{u}_{\text{el}} + \mathbf{M}_{\text{el}} \cdot \delta \theta_{\text{el}}$$

### *Hinge Joint Theory*

---

The [Hinge Joint](#), also known as a revolute joint, has one rotational degree of freedom between the two components. The two components are free to rotate relative to each other about the axis of the joint.



The following is an addition to the [Prismatic Joint Theory](#), some of which is also applicable for the hinge joint.

---

#### **HINGE JOINT AXIS AND ITS LOCAL COORDINATE SYSTEM**

In 2D, the initial joint axes ( $e_{10}$ ,  $e_{20}$ , and  $e_{30}$ ) are always global spatial axes with the initial joint axis ( $e_{10}$ ) being the out-of-plane axis ( $e_z$ ):

$$\mathbf{e}_{10} = \mathbf{e}_z$$

$$\mathbf{e}_{20} = \mathbf{e}_x$$

$$\mathbf{e}_{30} = \mathbf{e}_y$$

### HINGE JOINT FORMULATION

For a hinge joint, the destination attachment is free to rotate relative to the source attachment about the joint axis. The relative rotation about the joint axis ( $\theta$ ) is the degree of freedom.

To formulate this kind of connection, the motion of the destination attachment is prescribed in terms of the motion of the source attachment with the following relations:

$$\mathbf{b}_{dst} = \tilde{\mathbf{b}}$$

$$\mathbf{u}_{c, dst} = \mathbf{u}_{c, src}$$

$$\begin{bmatrix} \tilde{a} \\ \tilde{\mathbf{b}} \end{bmatrix} = \begin{bmatrix} a_r \\ \mathbf{b}_r \end{bmatrix} \otimes \begin{bmatrix} a_{src} \\ \mathbf{b}_{src} \end{bmatrix}$$

The relative quaternion in the global spatial coordinate system ( $a_r, \mathbf{b}_r$ ) is defined as:

$$a_r = \cos(\theta/2)$$

$$\mathbf{b}_r = \sin(\theta/2)\mathbf{e}_1$$

In 2D, the motion of the destination attachment is prescribed in terms of the motion of the source attachment by:

$$\phi_{dst} = \phi_{src} + \theta$$

$$\mathbf{u}_{c, dst} = \mathbf{u}_{c, src}$$

#### *Joint Elasticity in Hinge Joint*

The elastic degrees of freedom are written as

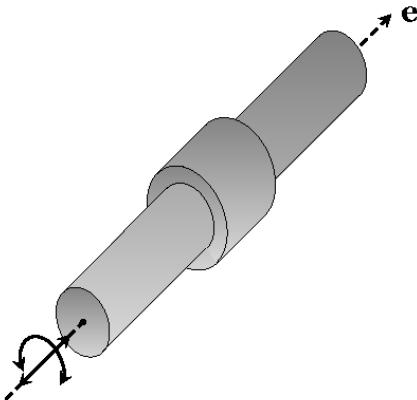
$$\mathbf{u}_{el} = u_1\mathbf{e}_1 + u_2\mathbf{e}_2 + u_3\mathbf{e}_3$$

$$\theta_{el} = \theta_2\mathbf{e}_2 + \theta_3\mathbf{e}_3$$

## *Cylindrical Joint Theory*

---

The [Cylindrical Joint](#) has one translational and one rotational degree of freedom between the two connected components. The components are free to slide and rotate relative to each other along the axis of joint. A cylindrical joint can be thought of as a combination of a prismatic joint and a hinge joint.



The following is an addition to the [Prismatic Joint Theory](#) and the [Hinge Joint Theory](#), which are also applicable for the cylindrical joint.

---

### **CYLINDRICAL JOINT FORMULATION**

For a cylindrical joint, the destination attachment is free to translate and rotate relative to the source attachment along the axis of joint. Thus, the degrees of freedom in this joint are the relative displacement ( $u$ ) along the joint axis and the relative rotation ( $\theta$ ) about the joint axis.

To formulate this kind of connection, the motion of the destination attachment is prescribed in terms of the motion of the source attachment as:

$$\begin{aligned}\mathbf{b}_{\text{dst}} &= \tilde{\mathbf{b}} \\ \mathbf{u}_{\text{c, dst}} &= \mathbf{u}_{\text{c, src}} + \mathbf{u}_r\end{aligned}$$

$$\begin{bmatrix} \tilde{a} \\ \tilde{\mathbf{b}} \end{bmatrix} = \begin{bmatrix} a_r \\ \mathbf{b}_r \end{bmatrix} \otimes \begin{bmatrix} a_{src} \\ \mathbf{b}_{src} \end{bmatrix}$$

The relative quaternion ( $a_r, \mathbf{b}_r$ ) and the relative displacement vector ( $\mathbf{u}_r$ ) in the global spatial coordinate system are defined as:

$$\begin{aligned} a_r &= \cos(\theta/2) \\ \mathbf{b}_r &= \sin(\theta/2)\mathbf{e}_1 \\ \mathbf{u}_r &= u\mathbf{e}_1 \end{aligned}$$

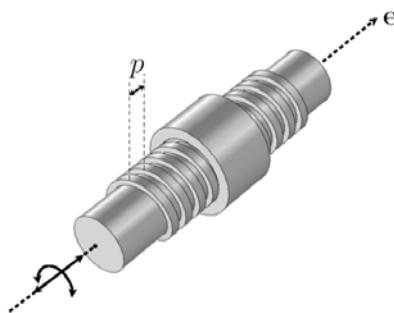
#### *Joint Elasticity in Cylindrical Joint*

The elastic degrees of freedom are written as

$$\begin{aligned} \mathbf{u}_{el} &= u_2\mathbf{e}_2 + u_3\mathbf{e}_3 \\ \theta_{el} &= \theta_2\mathbf{e}_2 + \theta_3\mathbf{e}_3 \end{aligned}$$

#### *Screw Joint Theory*

The **Screw Joint** is similar to the cylindrical joint except that the relative displacement and the relative rotation, instead of being independent, are linearly related to each other. Hence, this joint effectively has only one degree of freedom.



The following is an addition to the [Cylindrical Joint Theory](#), which is also applicable for the screw joint.

## SCREW JOINT FORMULATION

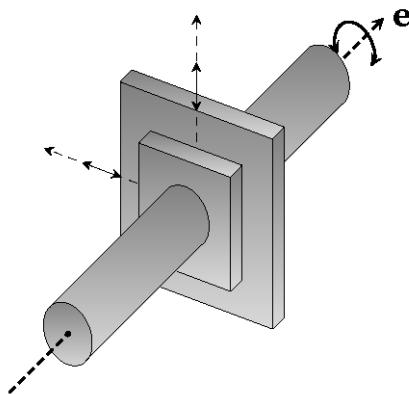
For the screw joint, an extra constraint is added to enforce the relation between the relative displacement ( $u$ ) and the relative rotation ( $\theta$ ):

$$u = n \cdot p \cdot \frac{\theta}{2\pi}$$

Here  $n$  and  $p$  are the number of starts and the pitch of the screw thread, respectively.

## Planar Joint Theory

The [Planar Joint](#) has two translational and one rotational degrees of freedom between the two components. Both components are free to translate relative to each other in a plane perpendicular to the axis of joint, and they are also free to rotate about this axis.



The following is an addition to the [Cylindrical Joint Theory](#), which is also applicable to the planar joint.

## PLANAR JOINT FORMULATION

For a planar joint, the destination boundaries are free to translate relative to the source boundaries in a plane perpendicular to the joint axis and free to rotate relative to the source boundaries about this axis. The degrees of freedom are two relative displacements along the second and third axes of the joint ( $u_2, u_3$ ) and one relative rotation about the joint axis ( $\theta$ ).

The joint formulation is similar to the cylindrical joint; the only difference is in the definition of the relative displacement vector ( $\mathbf{u}_r$ ):

$$\mathbf{u}_r = u_2 \mathbf{e}_2 + u_3 \mathbf{e}_3$$

### *Joint Elasticity in Planar Joint*

The elastic degrees of freedom are written as

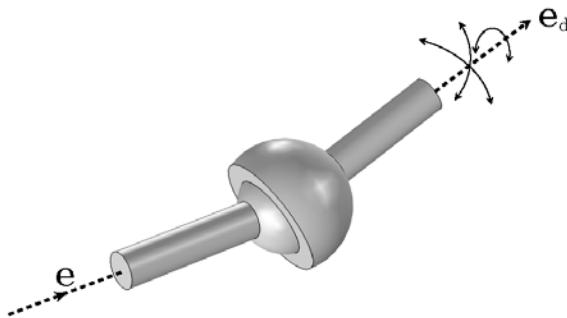
$$\mathbf{u}_{el} = u_1 \mathbf{e}_1$$

$$\theta_{el} = \theta_2 \mathbf{e}_2 + \theta_3 \mathbf{e}_3$$

### *Ball Joint Theory*

---

The **Ball Joint**, also known as a spherical joint, has three rotational degrees of freedom between the two attached components. The two components are free to rotate relative to each other along all three axes.



The following is an addition to the [Hinge Joint Theory](#), some of which is also applicable to the ball joint.

---

#### **BALL JOINT AXIS AND ITS LOCAL COORDINATE SYSTEM**

Unlike other joints, the ball joint has no particular axis about which the relative degrees of freedom between the two components are possible. This joint has rotational symmetry, and hence, relative rotation is allowed about any axis. However, application of constraints, springs, dampers, and so forth needs the source ( $\mathbf{e}_1$ ) and destination ( $\mathbf{e}_{d1}$ ) axes. The local coordinate system of this joint is defined using the source axis.

You specify the initial source axis ( $\mathbf{e}_{10}$ ) and initial destination axis ( $\mathbf{e}_{d10}$ ) in terms of components along the axes of the selected coordinate system. The initial source axis must point toward the center of joint, whereas the initial destination axis must point away from the center of joint.

The process of creating a rotated source axis, rotated destination axis, and a corresponding local coordinate system is the same as for the hinge joint. The auxiliary axis is taken as the initial destination axis unless it is parallel to the initial source axis.

### **BALL JOINT FORMULATION**

For a ball joint, the destination attachment is free to rotate relative to the source attachment about all three axes. The added degrees of freedom are contained in a relative quaternion ( $a, b, c$ , and  $d$ ), which represents the rotation in 3D. The rest of the formulation is similar to that of the hinge joint.

#### *Joint Elasticity in Ball Joint*

Since only translations are constrained in a ball joint, the elastic degrees of freedom are

$$\mathbf{u}_{el} = u_1 \mathbf{e}_1 + u_2 \mathbf{e}_2 + u_3 \mathbf{e}_3$$

Therefore, there are only elastic forces and no elastic moments or coupling matrices.

$$\mathbf{F}_{s, el} = \mathbf{k}_u \mathbf{u}_{el}$$

$$\mathbf{F}_{d, el} = \mathbf{c}_u \dot{\mathbf{u}}_{el}$$

The total elastic joint forces can be written using the spring and damping forces as

$$\mathbf{F}_{el} = -\mathbf{F}_{s, el} - \mathbf{F}_{d, el}$$

The contribution to the virtual work is

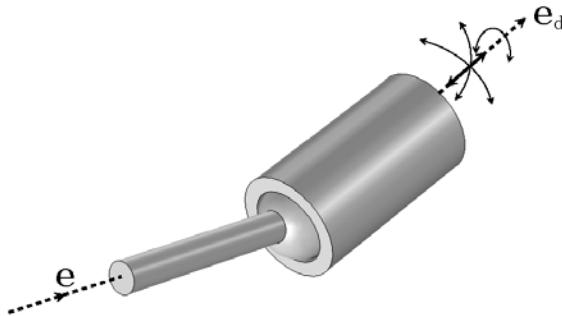
$$\delta W = \mathbf{F}_{el} \cdot \delta \mathbf{u}_{el}$$

#### *Slot Joint Theory*

---

The **Slot Joint** has three rotational and one translational degrees of freedom between the two attached components. The components are free to rotate relative to each other

about all three axes and also free to translate along the joint translation axis. The slot joint can be seen as a combination of the prismatic joint and the ball joint.



The following is an addition to the [Prismatic Joint Theory](#) and the [Ball Joint Theory](#), which are also applicable to the slot joint.

#### **SLOT JOINT AXIS AND ITS LOCAL COORDINATE SYSTEM**

For a slot joint, the translational motion is free along the joint translational axis, which can be either the source axis or the destination axis. The translational degree of freedom is added along the joint translational axis.

You can specify the initial joint translational axis ( $\mathbf{e}_{tr0}$ ) to be either

$$\mathbf{e}_{tr0} = \mathbf{e}_{10} \text{ or } \mathbf{e}_{tr0} = \mathbf{e}_{d10}$$

For the slot joint, the process of defining the source axis, destination axis, and local coordinate system is the same as for the ball joint.

The joint coordinate system is always attached to the source part even if the joint translational axis is defined by the destination axis.

#### **SLOT JOINT FORMULATION**

The slot joint formulation is similar to the ball joint except that an extra translational degree of freedom is needed. The joint translational axis is used for the relative displacement vector, and the rotational degrees of freedom are represented by a relative quaternion as in the ball joint.

### *Joint Elasticity in Slot Joint*

The elastic degree of freedom depends on whether the joint translational axis is defined to be the source axis or the destination axis. In both cases, there are two constrained translations, but it is only when the source axis is the joint translational axis that the elastic displacement can be expressed in the joint coordinate system direction

$$\mathbf{u}_{el} = u_2 \mathbf{e}_2 + u_3 \mathbf{e}_3$$

When the joint translational axis is the destination axis, then

$$\mathbf{u}_{el} = u_2 \mathbf{e}_{d2} + u_3 \mathbf{e}_{d3}$$

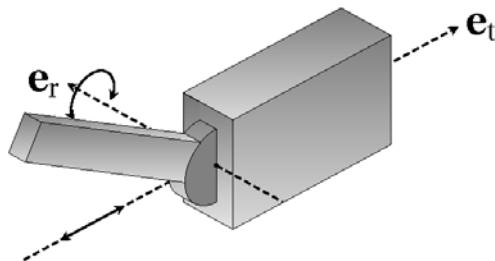
where  $\mathbf{e}_{d2}$  and  $\mathbf{e}_{d3}$  are second and third axis of the destination coordinate system, respectively. In this case, the stiffness and damping matrices are given along the destination axes when the joint coordinate system is selected.

Except for this difference, the formulation is the same as for the ball joint.

### *Reduced Slot Joint Theory*

---

The **Reduced Slot Joint** has one rotational and one translational degree of freedom between the two attached components. Both components of the system are free to rotate about the joint rotational axis and free to translate along the joint translation axis. The reduced slot joint can be thought of as a combination of the prismatic joint and the hinge joint, with different axes for the allowed degrees of freedom. The translational axes and the rotational axis are orthogonal to each other.



The reduced slot joint in 2D is similar to the reduced slot joint in 3D. The only difference is the initial joint rotational axis ( $\mathbf{e}_{\text{rot}0}$ ), which does not change in 2D and is always the out-of-plane axis ( $\mathbf{e}_z$ ).



The following is an addition to the [Cylindrical Joint Theory](#), which is also applicable for the reduced slot joint.

#### **REDUCED SLOT JOINT AXES AND ITS LOCAL COORDINATE SYSTEM**

The translational motion for a reduced slot joint is free along the joint translational axis, which can either be attached on the source attachment or the destination attachment. The translational degree of freedom is added along the joint translational axis.

In addition to the translational axis, information about the joint rotational axis is also required. You specify its components in the selected coordinate system. In this joint, the initial joint axis ( $\mathbf{e}_{10}$ ) and initial second axis ( $\mathbf{e}_{20}$ ) are the same as the initial joint translational axis ( $\mathbf{e}_{\text{tr}0}$ ) and initial joint rotational axis ( $\mathbf{e}_{\text{rot}0}$ ), respectively. If these axes are not orthogonal on input, then the rotational axis is adjusted in the plane spanned by  $\mathbf{e}_{\text{tr}0}$  and  $\mathbf{e}_{\text{rot}0}$  so as to fulfill that criterion.

#### **REDUCED SLOT JOINT FORMULATION**

The joint formulation is similar to the cylindrical joint except that the translational degree of freedom is defined along the joint translation axis and the rotational degree of freedom is defined about the joint rotational axis.

##### *Joint Elasticity in Reduced Slot Joint*

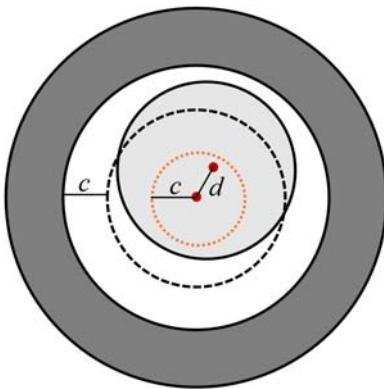
The elastic degrees of freedom are written as

$$\mathbf{u}_{\text{el}} = u_2 \mathbf{e}_2 + u_3 \mathbf{e}_3$$

$$\theta_{\text{el}} = \theta_1 \mathbf{e}_1 + \theta_3 \mathbf{e}_3$$

#### *Clearance Joint Theory*

The [Clearance Joint](#) allows two points that are connected to a pair of attachments or rigid domains to move within a specified clearance distance. The components connected through clearance joint are free to rotate relative to each other about all three axes and free to translate relative to each other within the specified clearance limit.



The following is an addition to the [Ball Joint Theory](#) and [Hinge Joint Theory](#), some of which is also applicable for the clearance joint.

#### CLEARANCE JOINT FORMULATION

In 3D, the formulation of the clearance joint is similar to a ball joint, except for the translational constraints. Analogously, in 2D, the clearance joint formulation is similar to a hinge joint. Instead of translational constraints, a penalty-based formulation is used to enforce that the distance between specified points on both selected components is within the clearance distance.

The penalty force is computed as:

$$F_j = if(gap \leq 0, -p_j \cdot gap, 0)$$

where  $F_j$  is the joint force magnitude, and  $p_j$  is the penalty factor input having a default value of  $0.1 * diag * E_{equ}$ . In this expression,  $E_{equ}$  is an effective Young's modulus, and  $diag$  is the geometric diagonal of the bounding box of the geometry.

The  $gap$  variable is the difference between the clearance ( $c$ ) and the distance between the connection points ( $d$ ).

$$gap = c - d$$

The distance between connection points ( $d$ ) is calculated as:

$$d = |(\mathbf{X}_{src} + \mathbf{u}_{src}) - (\mathbf{X}_{dst} + \mathbf{u}_{dst})|$$

Here  $\mathbf{X}_{\text{src}}$  and  $\mathbf{X}_{\text{dst}}$  are the undeformed locations of the two connection points, and  $\mathbf{u}_{\text{src}}$  and  $\mathbf{u}_{\text{dst}}$  are the corresponding displacements.

### *Fixed Joint Theory*

---

The **Fixed Joint** rigidly connects the two components and does not allow any relative motion between them. It does not add any degrees of freedom.



The following is an addition to the [Prismatic Joint Theory](#), some of which is also applicable for the fixed joint.

#### **FIXED JOINT FORMULATION**

Constraints are added stating that the both parts have the same value of displacements and rotations at the center of joint. In 3D six constraints are added:

$$\begin{aligned}\mathbf{u}_{c, \text{dst}} &= \mathbf{u}_{c, \text{src}} \\ \mathbf{b}_{\text{dst}} &= \mathbf{b}_{\text{src}}\end{aligned}$$

In 2D three constraints are added:

$$\begin{aligned}\mathbf{u}_{c, \text{dst}} &= \mathbf{u}_{c, \text{src}} \\ \phi_{\text{dst}} &= \phi_{\text{src}}\end{aligned}$$

The position of the center of joint only affects interpretation of the joint forces and moments.

#### *Joint Elasticity in Fixed Joint*

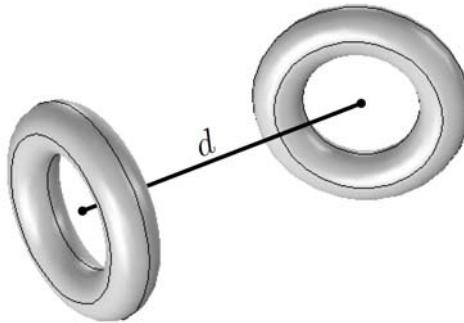
The elastic degrees of freedom are written as

$$\begin{aligned}\mathbf{u}_{\text{el}} &= u_1 \mathbf{e}_1 + u_2 \mathbf{e}_2 + u_3 \mathbf{e}_3 \\ \theta_{\text{el}} &= \theta_1 \mathbf{e}_1 + \theta_2 \mathbf{e}_2 + \theta_3 \mathbf{e}_3\end{aligned}$$

### *Distance Joint Theory*

---

The **Distance Joint** maintains a specified distance between two points that are connected to a pair of attachments or rigid domains. As a default, the initial distance is maintained, so that the joint can be considered as a rigid link with ball joints at both ends. It is also possible to vary the distance so that the joint acts as a telescopic link.



The following is an addition to the [Prismatic Joint Theory](#), some of which is also applicable for the distance joint.

#### DISTANCE JOINT FORMULATION

The formulation of the distance joint differs completely from the formulation of the other joint types. Only a constraint is applied to maintain the distance between the specified points on both selected components. This distance constraint can be written as:

$$|(\mathbf{X}_{\text{src}} + \mathbf{u}_{\text{src}}) - (\mathbf{X}_{\text{dst}} + \mathbf{u}_{\text{dst}})| = |\mathbf{X}_{\text{src}} - \mathbf{X}_{\text{dst}}| + \Delta$$

Here  $\mathbf{X}_{\text{src}}$  and  $\mathbf{X}_{\text{dst}}$  are the undeformed locations of the two connection points, and  $\mathbf{u}_{\text{src}}$  and  $\mathbf{u}_{\text{dst}}$  are the corresponding displacements. It is possible to vary the distance between the points through the extension value  $\Delta$ .

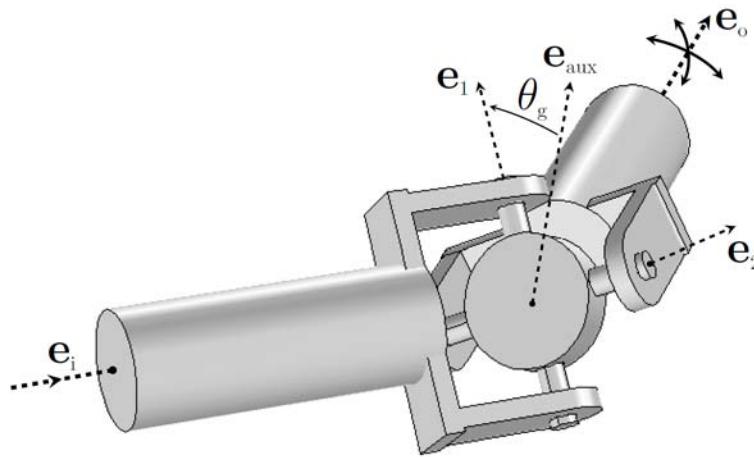
This constraint can be simplified and is used in the following form:

$$(\mathbf{u}_{\text{src}} - \mathbf{u}_{\text{dst}}) \cdot (\mathbf{u}_{\text{src}} - \mathbf{u}_{\text{dst}}) + 2(\mathbf{u}_{\text{src}} - \mathbf{u}_{\text{dst}}) \cdot (\mathbf{X}_{\text{src}} - \mathbf{X}_{\text{dst}}) = \Delta^2 + 2\Delta|\mathbf{X}_{\text{src}} - \mathbf{X}_{\text{dst}}|$$

#### *Universal Joint Theory*

An [Universal Joint](#), also known as Hooke's joint or Cardan joint, is commonly used to transmit rotational motion between shafts that have different orientations. Physically this joint is made up of a pair of hinge joints, oriented perpendicular to each other, connected by a cross shaft or gimbal. This joint is a kind of abstract joint where the

gimbal is not modeled and both the shafts are directly connected to each other. In general, this joint is used to transfer mechanical power from one shaft to another shaft. The added degrees of freedom in this joint are the relative rotations about the two legs of the gimbal that form the joint axis and second axis.



The following is an addition to the [Prismatic Joint Theory](#), some of which is also applicable for the universal joint.

#### JOINT AXIS AND LOCAL COORDINATE SYSTEM

In this joint, the initial orientation of input and output shafts are based on user inputs. The initial auxiliary axis is defined as perpendicular to the plane spanned by the shafts:

$$\mathbf{e}_{aux0} = \mathbf{e}_{i0} \times \mathbf{e}_{o0}$$

If both the shafts have the same initial orientation, the initial output shaft orientation in this expression is replaced with an arbitrary axis, which is not parallel to the initial input shaft orientation.

You can also define the auxiliary axis explicitly. In this case, a projection is made in the plane perpendicular to the initial input shaft orientation to ensure the orthogonality of initial auxiliary axis and initial input shaft orientation.

The initial axis connecting the gimbal and input shaft or initial joint axis can be defined as being the same as the auxiliary axis

$$\mathbf{e}_{10} = \mathbf{e}_{\text{aux}0}$$

Alternatively, the initial joint axis can be defined by rotating the initial auxiliary axis ( $\mathbf{e}_{\text{aux}0}$ ) about the initial input shaft orientation ( $\mathbf{e}_{i0}$ ) with a specified angle ( $\theta_{g0}$ ):

$$\mathbf{e}_{10} = \mathbf{e}_{\text{aux}0} \cos(\theta_{g0}) + (\mathbf{e}_{i0} \times \mathbf{e}_{\text{aux}0}) \sin(\theta_{g0}) + \mathbf{e}_{i0} (\mathbf{e}_{i0} \cdot \mathbf{e}_{\text{aux}0}) (1 - \cos(\theta_{g0}))$$

The initial second axis and initial third axis are formed by:

$$\begin{aligned}\mathbf{e}_{20} &= \mathbf{e}_{10} \times \mathbf{e}_{o0} \\ \mathbf{e}_{30} &= \mathbf{e}_{10} \times \mathbf{e}_{20}\end{aligned}$$

The rotated local coordinate system computed as:

$$\begin{aligned}\mathbf{e}_1 &= R_{\text{src}} \mathbf{e}_{10} \\ \mathbf{e}_{30} &= R_{\text{dst}} \mathbf{e}_{20} \\ \mathbf{e}_3 &= \mathbf{e}_1 \times \mathbf{e}_2\end{aligned}$$

The current orientation of input and output shafts is computed by multiplying the initial directions with the corresponding rotation matrices:

$$\begin{aligned}\mathbf{e}_i &= R_{\text{src}} \mathbf{e}_{i0} \\ \mathbf{e}_o &= R_{\text{dst}} \mathbf{e}_{o0}\end{aligned}$$

#### UNIVERSAL JOINT FORMULATION

The formulation is similar to that of a hinge joint, except the definition of the relative quaternion, which is written as

$$(\mathbf{a}_r, \mathbf{b}_r) = (\mathbf{a}_{r2}, \mathbf{b}_{r2}) \otimes (\mathbf{a}_{r1}, \mathbf{b}_{r1})$$

where

$$\begin{aligned}\mathbf{a}_{r1} &= \cos(\theta_1/2) & \mathbf{b}_{r1} &= \sin(\theta_1/2) \mathbf{e}_1 \\ \mathbf{a}_{r2} &= \cos(\theta_2/2) & \mathbf{b}_{r2} &= \sin(\theta_2/2) \mathbf{e}_2\end{aligned}$$

### *Joint Elasticity in Universal Joint*

The elastic degrees of freedom are written as

$$\mathbf{u}_{el} = u_1 \mathbf{e}_1 + u_2 \mathbf{e}_2 + u_3 \mathbf{e}_3$$

$$\theta_{el} = \theta_3 \mathbf{e}_3$$

### **POSTPROCESSING VARIABLES**

For the universal joint, some extra variables for use in postprocessing are defined.

#### *Angular Velocity and Total Rotation of Input and Output Shafts*

The angular velocities and total rotation for both input and output shafts are computed:

$$\omega_i = \omega_i \cdot \mathbf{e}_i$$

$$\omega_o = \omega_o \cdot \mathbf{e}_o$$

$$\theta_i = \int \omega_i \, dt$$

$$\theta_o = \int \omega_o \, dt$$

In order to evaluate the integrals, two extra ODE degrees of freedom are created for each universal joint.

#### *Torsional Moment in the Shafts*

The torsional moments in the input and output shafts are computed using

$$M_i = \mathbf{M}_i \cdot \mathbf{e}_i$$

$$M_o = \mathbf{M}_o \cdot \mathbf{e}_o$$

#### *Angle Between Shafts*

The angle between the shafts (bend angle)  $\beta$  is computed using

$$\beta = \arccos(\mathbf{e}_i \cdot \mathbf{e}_o)$$

### *Theory for Joint Constraints*

---

**Constraints** are used to restrict the relative motion between the two components sharing a joint. You can specify the upper and lower bound of the relative motion. Applying constraints on the relative motion can be interpreted as placing an imaginary

stopper that restricts the motion of the destination attachment with respect to the source attachment in a prescribed direction. [Translational Constraints](#), [Rotational Constraints](#), [Inclination Angle Constraints](#), and [Axial Rotation Constraints](#) are discussed in this section.

### TRANSLATIONAL CONSTRAINTS

Translational constraints are available for the [Prismatic Joint](#), [Cylindrical Joint](#), [Screw Joint](#), [Planar Joint](#), [Slot Joint](#), and [Reduced Slot Joint](#) features.

The maximum relative displacement ( $u_{\max}$ ) and minimum relative displacement ( $u_{\min}$ ) can be prescribed. The activation condition for the constraint is:

$$(u > u_{\max}) \parallel (u < u_{\min})$$

A penalty method is used to enforce the constraints.

#### *Penalty Method*

The constraint on the relative displacement is enforced using a stiff spring between the components. The stiffness of the spring is defined by the penalty factor. When the constraint is activated in a dynamic analysis, the time steps are reduced in order to conserve the momentum in the system.

The default value of the penalty factor  $p_u$  is evaluated as:

$$p_u = \left( \frac{\frac{du}{dt}}{\delta u_{\max}} \right)^2 m_e \quad (3-1)$$

The numerator in [Equation 3-1](#) is an assumed relative velocity between the components when the constraint is applied, and the denominator is the maximum allowable penetration. This ratio of the relative velocity and the maximum allowable penetration decides the required stiffness of the spring (the penalty factor). The default value of this ratio is  $10^3 \text{ s}^{-1}$ . This could, for example, be a penetration of 1 mm with an impact speed of 1 m/s. You can change this ratio based on the dynamics of the system and the allowable constraint violation.



For very high penalty factors, the solution can be inaccurate unless the tolerances are tightened.

The factor  $m_e$  in the penalty factor expression is the effective mass at the joint, defined as

$$m_e = \frac{(m_{src}m_{dst})}{(m_{src} + m_{dst})}$$

The source and destination masses  $m_{src}$  and  $m_{dst}$  are defined as:

- The mass of the domains attached to the source or destination attachments in case of flexible components.
- The mass of the attached rigid domain in case of rigid components.

The following assumptions are made in the definition of the effective mass:

- Both parts are free to move. If only one of the parts is free, you should use the mass associated with that part as the effective mass.
- The computation of  $m_{src}$  and  $m_{dest}$  considers the domains that are directly adjacent to the source and destination attachment. These domains cannot be the only parts affecting the dynamics of an impact.
- The computation of  $m_{src}$  and  $m_{dest}$  also do not account for the mass added using Added Mass features or Mass and Moment of Inertia features.

The default value for the maximum time step allowed in order to capture the event accurately is evaluated as:

$$\Delta t_{max} = \frac{(T/2)}{(30)} \quad T = 2\pi \sqrt{\frac{m_e}{p_u}}$$

where  $T$  is the time period for free vibration of a spring-mass system having a stiffness equal to the penalty factor and mass equal to the effective mass. As can be seen from the expressions above, the default maximum time step is approximately 0.1 ms.

The constraint is violated for half of the period ( $T/2$ ). To capture that phenomenon accurately, approximately 30 time steps are used. There is a tradeoff between accuracy and computation time. A larger number of steps reduces the maximum allowable time step and increases the accuracy in conserving the momentum. However, this also increases the computation time.

To prevent the solver from taking time steps larger than the maximum allowable time step, the penalty factor actually used is:

$$p_{mu} = p_u + ((u > u_{max}) \parallel (u < u_{min})) \cdot if((\Delta t > \Delta t_{max}), \infty, 0)$$

This means that modified penalty factor  $p_{\text{mu}}$  is equal to the given penalty factor as long as the time step taken by the solver is smaller than the maximum allowable time step. If the constraint is violated, and the time step taken by the solver is larger than the maximum allowable time step, the solver is forced to decrease the time step.



The penalty factor is not modified for quasistatic and stationary-parametric types of analyses because the conservation of momentum is not relevant in these cases.

The force required to constrain the relative motion is

$$F_u = \begin{cases} -p_{\text{mu}} \cdot (u - u_{\text{max}}) & u > u_{\text{max}} \\ -p_{\text{mu}} \cdot (u - u_{\text{min}}) & u < u_{\text{min}} \\ 0 & \text{else} \end{cases}$$

To enforce a constraint condition, a virtual work contribution is added when constraints are active:

$$\delta W = F_u \cdot \delta u$$

In the case of a [Planar Joint](#), the relative displacement is constrained along a user-defined axis (the *constraint direction*) oriented in a plane perpendicular to the joint axis. The relative displacement in the constraint direction,  $u_c$ , is:

$$u_c = (u_2 \mathbf{e}_2 + u_3 \mathbf{e}_3) \cdot (R_{\text{src}} \mathbf{e}_c)$$

where  $\mathbf{e}_c$  is the normalized constraint axis given by the user.

### ROTATIONAL CONSTRAINTS

Rotational constraints are available for the [Hinge Joint](#), [Cylindrical Joint](#), [Screw Joint](#), [Planar Joint](#), and [Reduced Slot Joint](#) features.

The theory for rotational constraints is similar to what is described above for [Translational Constraints](#).

Here, the penalty factor should be considered as a torsional spring. The effective mass in the default penalty factor is replaced by an effective moment of inertia, having an analogous definition.

## INCLINATION ANGLE CONSTRAINTS

Inclination angle constraints are available for the [Ball Joint](#) and [Slot Joint](#) features.

Here, the relative rotation to which the constraint is applied is the inclination of the destination axis with respect to the reference axis (also called an inclination angle or polar angle). The reference axis can be chosen as the initial source axis, initial destination axis, or as a user-defined arbitrary axis. The reference axis is always following the source attachment.

The constraint is applied only on the upper limit of the inclination angle. The upper limit of the inclination angle should be less than 180°. The constraint on the inclination angle is independent of the azimuthal angle and cannot be a function of azimuthal angle.

You give the maximum relative inclination angle ( $\theta_{\max}$ ) as a limit on the inclination of the destination attachment with respect to the reference direction. The actual constraint is applied to the cosine of the inclination angle:

$$\cos \theta = \mathbf{e}_R \cdot \mathbf{e}_{d1}$$

where  $\mathbf{e}_R$  is the reference axis and  $\mathbf{e}_{d1}$  is the destination axis.

The activation condition for an inclination angle constraint is written as:

$$\cos \theta < \cos \theta_{\max}$$

The default value for the penalty factor and maximum allowable time step is computed as for [Rotational Constraints](#). The moments of inertia for the source and destination components are computed about an axis that is passing through the joint center and is perpendicular to both the source and destination axes.

In order to prevent the solver from taking time steps larger than the maximum allowable time step, a modified penalty factor is defined as:

$$p_{m\theta} = p_{\theta} + (\cos \theta < \cos \theta_{\max}) \cdot \text{if}((\Delta t > \Delta t_{\max}), \infty, 0)$$

The moment required to enforce the inclination angle constraint is made proportional to the difference in the cosines of the angles, rather than to the difference in the angles themselves. This avoids operating with inverse trigonometric functions.

$$M_{\theta} = \begin{cases} p_{m\theta} \cdot (\cos \theta - \cos \theta_{\max}) \cdot \sin \theta_{\max} & \cos \theta < \cos \theta_{\max} \\ 0 & \text{else} \end{cases}$$

To enforce the constraint condition, a virtual work contribution is added for the cosine of the inclination angle:

$$\delta W = M_\theta \delta\theta = -\frac{M_\theta}{\sin\theta} \cdot \delta(\cos\theta) \approx -\frac{M_\theta}{\sin\theta_{\max}} \cdot \delta(\cos\theta)$$

### AXIAL ROTATION CONSTRAINTS

Axial rotation constraints are available for the [Ball Joint](#) and [Slot Joint](#) features.

The relative rotation, on which the constraint is applied, is the relative rotation of the destination attachment about its own axis.

The constraint can be applied as both the upper and lower limits on the relative rotation of the destination attachment about its own axis. This constraint is uniquely defined as long as the maximum relative rotation is less than 180 degrees.

You prescribe the maximum and minimum axial relative rotation of destination attachment  $\Psi_{\max}$  and  $\Psi_{\min}$ . The cosine of the relative axial rotation of the destination attachment is written as:

$$\cos\psi = 1 - 2 \cdot (\mathbf{e}_r \cdot \mathbf{e}_{d1})^2 \cdot (1 - a_r^2)$$

$$\mathbf{e}_r = \frac{1}{\sqrt{(b_r^2 + c_r^2 + d_r^2)}} \cdot \begin{bmatrix} b_r \\ c_r \\ d_r \end{bmatrix}$$

Here  $(a_r, b_r, c_r, d_r)$  represents the relative quaternion, and  $\mathbf{e}_r$  is the normalized relative rotation axis.

The activation conditions for the maximum and minimum relative axial rotation constraints are written as:

$$(\cos\psi < \cos\psi_{\max}) \cdot (\mathbf{e}_r \cdot \mathbf{e}_{d1} > 0)$$

$$(\cos\psi < \cos\psi_{\min}) \cdot (\mathbf{e}_r \cdot \mathbf{e}_{d1} < 0)$$

The default value computation for the penalty factor and maximum allowable time step is the same as for [Rotational Constraints](#). The moment of inertia for the source and destination components are computed about an axis that is passing through the center of joint and is parallel to the destination axis.

The modified penalty factor, constraint moment, and virtual work contribution are similar to the [Inclination Angle Constraints](#). The only change is in the constraint activation conditions.

### *Theory for Joint Locking*

---

The [Locking](#) feature is used to lock the relative motion between the two components connected to a joint. You specify limits on the relative motion in the form of an upper and lower bound. This can be interpreted as an imaginary object (like a snap hook) that locks one of the free relative degrees of freedom of the destination attachment when it reaches the limiting value.

The Locking feature differs from [Constraints](#) in a sense that the value of the particular joint degrees of freedom becomes constant for the rest of the analysis after it attains its prescribed limiting value.

The [Translational Locking](#), [Rotational Locking](#), [Inclination Angle Locking](#), and [Axial Rotation Locking](#) types are discussed in this section.

#### **TRANSLATIONAL LOCKING**

Translational locking is available for the [Prismatic Joint](#), [Cylindrical Joint](#), [Screw Joint](#), [Planar Joint](#), [Slot Joint](#), and [Reduced Slot Joint](#) features.

In this locking type, a lock is applied to the relative displacement ( $u$ ). When the lock is active, its relative displacement becomes constant at the prescribed limiting value and the relative velocity goes to zero. If lock conditions are active for all the joint degrees of freedom, then the source and destination attachment move as a single object.

You prescribe the maximum ( $u_{\max}$ ) and minimum ( $u_{\min}$ ) limits of relative displacement. The activation condition for locking is:

$$(u > u_{\max}) \parallel (u < u_{\min})$$

Locking is implemented using a penalty method where the relative velocity is forced to become zero for the rest of the time. There is a loss of energy during locking, which can be interpreted as the energy given to the imaginary object responsible for locking the degrees of freedom. Due to the locking phenomenon, stress waves are generated in the flexible components. You can add damping to the problem to suppress unwanted stress waves.

The locking indicators ( $i_{\text{umax}}$  and  $i_{\text{umin}}$ ) are ODE variables used to ensure that the lock is active for the rest of the simulation. The values of the indicators are switched from zero to a nonzero value when the lock is active, through the following equations:

$$\frac{d}{dt}(i_{\text{umax}}) = (u > u_{\text{max}})$$

$$\frac{d}{dt}(i_{\text{umin}}) = (u < u_{\text{min}})$$

The force required to lock the relative motion is

$$F_u = \begin{cases} -p_{u1} \cdot (u - u_{\text{max}}) - p_{u2} \cdot \frac{du}{dt} & i_{\text{umax}} > 0 \\ -p_{u1} \cdot (u - u_{\text{min}}) - p_{u2} \cdot \frac{du}{dt} & i_{\text{umin}} > 0 \\ 0 & \text{else} \end{cases}$$

where  $p_{u1}$  and  $p_{u2}$  are the penalty factors.  $p_{u1}$  can be interpreted as a spring constant and its default value is a function of an equivalent Young's modulus and geometric dimensions.  $p_{u2}$  is a damping coefficient and its default value is 10 ms times the value of  $p_{u1}$ . these penalty factors are inputs and can you can modify them as required.

The value of the penalty factor ( $p_{u2}$ ) controls the rate of decay of the relative velocity. A large value gives a high rate of decay but can cause convergence problems.

To enforce the locking condition, a contribution is added to the virtual work:

$$\delta W = F_u \cdot \delta u$$

In the case of a [Planar Joint](#), the relative displacement is locked along a user-defined axis (the *locking direction*) oriented in a plane perpendicular to the joint axis. The relative displacement in the locking direction,  $u_1$ , is:

$$u_1 = (u_2 \mathbf{e}_2 + u_3 \mathbf{e}_3) \cdot (R_{\text{src}} \mathbf{e}_1)$$

where  $\mathbf{e}_1$  is the normalized locking axis given by the user.

#### ROTATIONAL LOCKING

Rotational locking is available for the [Hinge Joint](#), [Cylindrical Joint](#), [Screw Joint](#), [Planar Joint](#), and [Reduced Slot Joint](#) features.

The theory for rotational constraints is similar to what is described above for [Translational Locking](#).

Here, the penalty factor should be considered as a torsional spring.

### INCLINATION ANGLE LOCKING

Inclination angle locking is available for the [Ball Joint](#) and [Slot Joint](#) features.

Here, the lock is applied on the inclination of the destination axis relative to the reference axis (the inclination angle or polar angle). The ODE variable used as locking indicator is computed from

$$\frac{d}{dt}(i_{\theta_{\max}}) = (\cos \theta < \cos \theta_{\max})$$

The moment required to apply an inclination angle lock is written as

$$M_{\theta} = \begin{cases} \left( p_{\theta 1} \cdot (\cos \theta - \cos \theta_{\max}) + p_{\theta 2} \cdot \frac{d}{dt} \cos \theta \right) \cdot \sin \theta_{\max} & i_{\theta_{\max}} > 0 \\ 0 & \text{else} \end{cases}$$

To apply the lock, virtual work contributions are added for the cosine of the inclination angle:

$$\delta W = M_{\theta} \delta \theta = -\frac{M_{\theta}}{\sin \theta} \cdot \delta(\cos \theta) \approx -\frac{M_{\theta}}{\sin \theta_{\max}} \cdot \delta(\cos \theta)$$

### AXIAL ROTATION LOCKING

Axial rotation locking is available for the [Ball Joint](#) and [Slot Joint](#) features.

Here, the lock is applied to the relative rotation of the destination attachment about its own axis. There are two ODE variables used as locking indicator. They are updated using

$$\frac{d}{dt}(i_{\psi_{\max}}) = (\cos \psi < \cos \psi_{\max}) \cdot ((\mathbf{e}_r \cdot \mathbf{e}_{d1}) > 0)$$

$$\frac{d}{dt}(i_{\psi_{\min}}) = (\cos \psi < \cos \psi_{\min}) \cdot ((\mathbf{e}_r \cdot \mathbf{e}_{d1}) < 0)$$

The moment required to enforce the lock on the axial rotation is:

$$M_\psi = \begin{cases} \left( p_{\psi 1} \cdot (\cos \psi - \cos \psi_{\max}) + p_{\psi 2} \cdot \frac{d}{dt} \cos \psi \right) \cdot \sin \psi_{\max} & i_{\psi \max} > 0 \\ \left( p_{\psi 1} \cdot (\cos \psi - \cos \psi_{\min}) + p_{\psi 2} \cdot \frac{d}{dt} \cos \psi \right) \cdot \sin \psi_{\min} & i_{\psi \min} > 0 \\ 0 & \text{else} \end{cases}$$

To apply the lock, a virtual work contribution is added for the cosine of the axial rotation of the destination boundaries:

$$\delta W = M_\psi \delta \psi = -\frac{M_\psi}{\sin \psi} \cdot \delta(\cos \psi) \approx \begin{cases} -\frac{M_\psi}{\sin \psi_{\max}} \cdot \delta(\cos \psi) & i_{\psi \max} > 0 \\ -\frac{M_\psi}{\sin \psi_{\min}} \cdot \delta(\cos \psi) & i_{\psi \min} > 0 \end{cases}$$

### *Theory for Springs and Dampers on Joints*

---

You can use the [Spring and Damper](#) feature to apply a spring or a dashpot to act on the relative motion in a joint. A dashpot can be used to account for various kinds of losses at the joint. The spring can be given a predeformation so that the spring equilibrium position is not the same as the initial state.

The [Translational Spring and Damper](#), [Rotational Spring and Damper](#), and [Inclination Angle and Axial Rotation Spring and Damper](#) versions of this feature are discussed in this section.

#### **TRANSLATIONAL SPRING AND DAMPER**

The translational type of spring and damper is available for the [Prismatic Joint](#), [Cylindrical Joint](#), [Screw Joint](#), [Planar Joint](#), [Slot Joint](#), and [Reduced Slot Joint](#) features.

The spring force, which is proportional to the relative displacement is

$$F_e = -k_u(u - u_0)$$

where  $k_u$  is the spring constant and  $u_0$  is the predeformation.

You can also specify the spring force as function of extension explicitly, as

$$F_e = -F_s(u_s)$$

where  $u_s$  is the spring extension.

The damping force, which is proportional to the relative velocity, is:

$$F_d = -c_u \dot{u}$$

where  $c_u$  is the damping coefficient.

The virtual work contribution is

$$\delta W = (F_e + F_d)\delta u$$

In the case of a [Planar Joint](#), springs and dashpots are attached along user-defined axes oriented in a plane perpendicular to the joint axis. The spring axis and dashpot axis can differ. The forces are proportional to the relative displacement along the spring axis ( $u_s$ ) and relative velocity along the dashpot axis ( $v_{dp}$ ), respectively:

$$u_s = (u_2 \mathbf{e}_2 + u_3 \mathbf{e}_3) \cdot (R_{src} \mathbf{e}_s)$$
$$v_{dp} = (v_2 \mathbf{e}_2 + v_3 \mathbf{e}_3) \cdot (R_{src} \mathbf{e}_{dp})$$

where  $\mathbf{e}_s$  is the normalized spring axis and  $\mathbf{e}_{dp}$  is the normalized damper axis.

#### **ROTATIONAL SPRING AND DAMPER**

The rotational type of spring and damper is available for the [Hinge Joint](#), [Cylindrical Joint](#), [Screw Joint](#), [Planar Joint](#), and [Reduced Slot Joint](#) features.

The theory for rotational spring and damper is similar to what is described above for [Translational Spring and Damper](#). All translational quantities are replaced by corresponding angular quantities.

#### **INCLINATION ANGLE AND AXIAL ROTATION SPRING AND DAMPER**

The inclination angle and axial rotation types of spring and damper are available for the [Ball Joint](#) and [Slot Joint](#) features.

The spring and damper here can affect two types of relative motion — the inclination angle and the axial rotation of the destination attachment. The inclination angle and axial rotation are computed as:

$$\theta = \arccos(\min(\max(\{ \cos \theta \}, -1), 1))$$

$$\psi = \arccos(\min(\max(\{ \cos \psi \}, -1), 1)) \cdot ((\mathbf{e}_r \cdot \mathbf{e}_{d1} > 0) - (\mathbf{e}_r \cdot \mathbf{e}_{d1} < 0))$$



The definitions of inclination angle and axial rotation, as well as the expressions of cosines, are discussed in the [Inclination Angle Constraints](#) and [Axial Rotation Constraints](#) sections.

As these rotations are computed using the inverse cosine function, they are unique as long as the magnitudes of the actual rotations ( $\theta$  and  $\Psi$ ) are less than  $180^\circ$ .

The spring and damping forces are for both cases similar to those used for a standard [Rotational Spring and Damper](#).

### *Theory for Prescribed Motion on Joints*

With [Prescribed Motion](#) you can control the relative motion in a joint, either by prescribing the displacement or the velocity. This is useful when the degrees of freedom in the joint are not free but are a known function of time. This is common in the field of robotics, for example.

The [Translational Prescribed Motion](#) and [Rotational Prescribed Motion](#) versions of this feature are discussed in this section.

#### **TRANSLATIONAL PRESCRIBED MOTION**

Translational prescribed motion is available for the [Prismatic Joint](#), [Cylindrical Joint](#), [Screw Joint](#), [Planar Joint](#), [Slot Joint](#), and [Reduced Slot Joint](#) features.

A constraint is added to prescribe the relative displacement in the joint. If no reaction forces are computed, then the constraint is enforced as a pointwise constraint:

$$u = u_p$$

where  $u_p$  is the prescribed relative displacement. If reaction force computation has been requested by selecting **Evaluate reaction forces**, then a weak constraint is used instead. An extra degree of freedom,  $R_F$ , is added for the reaction force. The contribution to the virtual work is

$$(u - u_p) \delta R_F + R_F \delta(u - u_p)$$

If a unidirectional constraint is used (by selecting **Apply reaction only on joint variables**), then the form of the virtual work is changed to

$$(u - u_p)\delta R_F + R_F\delta u$$

When the relative velocity is prescribed, an extra equation is added to find the relative displacement by integration of the relative velocity:

$$\frac{d}{dt}(u_p) = v_p$$

where  $v_p$  is the prescribed relative velocity and  $u_p$  is the velocity integration variable which then is used in the constraint. In case of a stationary study, the relative displacement in the joint is constrained to zero.

You can also set a deactivation condition on the prescribed motion constraints. Once the specified deactivation condition is fulfilled and the deactivation indicator expression ( $i_{up}$ ), a Boolean variable, is set to true, the constraints are removed from the system. This makes the previously prescribed joint degrees of freedom free.

In the case of a [Planar Joint](#), the relative displacements or velocities are prescribed along two axes oriented in a plane perpendicular to the joint axis. The first axis is specified by the user, while the second axis is in the same plane, and orthogonal to the first axis. The relative displacement along prescribed motion axis and along its orthogonal direction can be written as:

$$\begin{aligned} u_{ep} &= (u_2 \mathbf{e}_2 + u_3 \mathbf{e}_3) \cdot (R_{src} \mathbf{e}_p) \\ u_{ep2} &= (u_2 \mathbf{e}_2 + u_3 \mathbf{e}_3) \cdot (\mathbf{e}_1 \times R_{src} \mathbf{e}_p) \end{aligned}$$

where  $\mathbf{e}_p$  is the normalized prescribed motion axis.

#### ROTATIONAL PRESCRIBED MOTION

Rotational prescribed motion is available for the [Hinge Joint](#), [Cylindrical Joint](#), [Screw Joint](#), [Planar Joint](#), [Reduced Slot Joint](#), [Ball Joint](#), and [Slot Joint](#) features.

The theory for rotational prescribed motion is similar to that of [Translational Prescribed Motion](#). The difference is that the rotation or angular velocity is prescribed instead of displacement or translational velocity.

For the [Ball Joint](#) and [Slot Joint](#) features, relative motion is possible in all three directions. You prescribe the motion using a rotation axis and the magnitude of rotation or angular velocity about this axis.

The following constraint is added to [Ball Joint](#) and [Slot Joint](#) features to prescribe the relative rotation in a joint:

$$\mathbf{b}_r = \sin\left(\frac{\phi_p}{2}\right) \cdot \mathbf{e}_p$$

where  $\phi_p$  is the magnitude of the prescribed relative rotation, and  $\mathbf{e}_p$  is the axis of relative rotation.

If a relative angular velocity is prescribed, an extra equation is added to find the relative rotation by integration of the relative angular velocity:

$$\frac{d}{dt}(\phi_p) = \Omega_p$$

where  $\Omega_p$  is the prescribed relative angular velocity, and  $\phi_p$  is the angular velocity integration variable.

---

Evaluating reaction force for the prescribed motion by selecting **Evaluate reaction forces** in the **Reaction force settings** section of the settings for **Prescribed Motion** has several benefits over evaluating it using joint forces and moments computed using weak constraints:

- It includes the contributions of springs, dampers, and friction applied on the joint.
- It works even if there are redundant constraints in the system, especially in a rigid body system, since joint constraints cannot be applied in the weak form in such a system.
- It is computationally more efficient as it adds fewer degrees of freedom.

---

### *Theory for Applied Forces and Moments on Joints*

---

The **Applied Force and Moment** feature is used to apply forces and moments on all types of joints. It can be applied at the source or destination joint attachment, or directly to the joint degrees of freedom.

When the load is applied directly to the joint degrees of freedom, the virtual work contributions added depend on the joint type:

- With scalar force  $F$ , for joints with one translational degree of freedom  $u$  (Prismatic, Cylindrical, Screw, Slot, Reduced Slot):

$$F\delta u$$

- With vector force  $\mathbf{F}$ , used in the Planar joint which has two translational degrees of freedom expanded into a global relative displacement vector  $\mathbf{u}_r$ :

$$\mathbf{F} \cdot \delta \mathbf{u}_r$$

This means that the component of  $\mathbf{F}$  which is orthogonal to the plane of the joint translation is removed.

- With scalar moment  $M$ , for joints with one rotational degree of freedom  $\theta$  (Hinge, Cylindrical, Screw, Planar, Reduced Slot):

$$M\delta\theta$$

- With vector moment  $\mathbf{M}$ , for joints with three rotational degrees of freedom represented by the quaternion  $a_r, \mathbf{b}_r$  (Ball, Slot):

$$2(\{0, \mathbf{M}\} \otimes \{a_r, \mathbf{b}_r\} \otimes \delta\{a_r, \mathbf{b}_r\})$$



The theory for applying forces and moments on attachments is the same as described in the [Rigid Connector Theory](#) section in the *Structural Mechanics Module User's Guide*.



The links to sections described in external guides, such as the *Structural Mechanics Module User's Guide*, do not work in the PDF, only from the online help.

### Theory for Friction in Joints

The [Friction](#) feature is used to add frictional loss to a joint. It is available for [Prismatic Joint](#), [Hinge Joint](#), [Cylindrical Joint](#), [Screw Joint](#), [Planar Joint](#), and [Ball Joint](#).

The friction force is modeled using a continuous friction law, which is capable of modeling sliding-sticking phenomena. A strict application of Coulomb's law involves discrete transition from sticking to sliding and vice versa, as dictated by a vanishing relative velocity. These discrete transitions cause numerical difficulties and to avoid them, the friction force is approximated with a continuous friction law:

$$F_f = -\mu N \frac{v}{|v|} \left( 1 - e^{-\frac{|v|}{v_0}} \right)$$

where  $F_f$  is the friction force,  $\mu$  is the frictional coefficient,  $N$  is the normal force,  $v$  is the slip velocity, and  $v_0$  is the characteristic slip velocity. The term

$$\left( 1 - e^{\frac{-|v|}{v_0}} \right)$$

is called the regularization factor.

The regularization factor smooths the friction force discontinuity. The characteristic slip velocity should be chosen to be small in comparison to the characteristic relative velocities encountered during the simulation. The continuous friction law describes both sliding and sticking behavior, that is, it completely replaces Coulomb's law. Sticking is replaced by creeping between the contacting bodies with a small relative velocity.

The exact interpretation of the friction model for each joint type is described in the following. In general, an expression for the friction force such as

$$F_f = -\min(\mu N + F_{f,r}, F_{f,max}) \frac{\dot{u}}{|\dot{u}|} \left( 1 - e^{\frac{-|\dot{u}|}{v_0}} \right)$$

is used. Here  $F_{f,r}$  is the additional sliding resistance,  $F_{f,max}$  is the maximum friction force, and  $\dot{u}$  is the relative velocity in the joint. Since the joints are abstractions, there is no unique way to define the normal force  $N$ , to which the friction force is proportional. For each joint there are several available methods for computing  $N$  and you must select the most representative one.



Friction can only be applied to the joints in a time-dependent analysis.

## PRISMATIC JOINT

### Friction Force

The total friction force is defined as

$$F_f = -\min(\mu N + F_{f,r}, F_{f,max}) \frac{\dot{u}}{|\dot{u}|} \left( 1 - e^{\frac{-|\dot{u}|}{v_0}} \right)$$

The total normal force can be computed from joint forces, interference fit, or the combination of these two.

#### *Normal Force Case 1: Contribution from Joint Forces*

**Case-1.1:** Normal force computed from attachment reaction forces:

$$N = \sum |\mathbf{F}_a \cdot \mathbf{n}|$$

where  $\mathbf{F}_a$  are the reaction forces on the attachment boundaries, and  $\mathbf{n}$  is the surface normal in the deformed state.

**Case-1.2:** Normal force computed from joint forces and moments:

$$N = |\mathbf{F}_j - (\mathbf{F}_j \cdot \mathbf{e})\mathbf{e}| + |\alpha_{11}M_{j11}| + |\alpha_{12}M_{j12}| + |\alpha_{13}M_{j13}| \quad (3D)$$

$$N = |\mathbf{F}_j - (\mathbf{F}_j \cdot \mathbf{e})\mathbf{e}| + |\alpha M_{j13}| \quad (2D)$$

where  $\mathbf{F}_j$  are the joint forces,  $\mathbf{e}$  is the joint axis,  $M_{jli}$  are the components of the joint moment in joint local coordinate system, and  $\alpha_{li}$  are contribution factors given by the user.

**Case-1.3:** Normal force computed through joint force in a specified direction.

$$N = |(\mathbf{F}_j - (\mathbf{F}_j \cdot \mathbf{e})\mathbf{e}) \cdot (R_{src}\mathbf{e}_{n0})|$$

where  $\mathbf{e}_{n0}$  is a direction given by the user.  $R_{src}$  represents the rotation matrix for the rotation of the source attachment.



This method is available only in the prismatic joint.

---

#### *Normal Force Case 2: Contribution from Interference Fit*

**Case-2.1:** Normal force defined by pressure and contact area,

$$N = pA$$

where  $p$  is the contact pressure, and  $A$  is the contact area.

**Case-2.2:** Normal force defined by total force

$$N = F_n$$

where  $F_n$  is the total normal force.

#### *Virtual Work and Power Loss*

The virtual work contribution from the total friction force is obtained through a multiplication by the variation of the relative displacement between the two parts:

$$\delta W = F_f \cdot \delta u$$

The energy dissipation rate caused by friction can be written as

$$Q_f = -F_f \cdot \dot{u}$$

#### **HINGE JOINT**

Most of the theory is the same as for prismatic joints. Any differences are described here.

#### *Friction Moment*

The total friction moment is defined as:

$$M_f = -\min(\mu N + F_{f, r}, F_{f, \max}) \frac{\dot{\theta}}{|\dot{\theta}|} \left( 1 - e^{\frac{-|\dot{\theta}|}{(v_0/r)}} \right) r$$

where  $\dot{\theta}$  is the relative angular velocity in the joint, and  $r$  is the hinge radius, supplied by the user.

#### *Normal Force*

**Case-1.2:** Normal force computed through joint forces and moments:

$$N = |\mathbf{F}_j - (\mathbf{F}_j \cdot \mathbf{e})\mathbf{e}| + |\alpha_{12}M_{j12}| + |\alpha_{13}M_{j13}| \quad (3D)$$

$$N = |\mathbf{F}_j - (\mathbf{F}_j \cdot \mathbf{e})\mathbf{e}| \quad (2D)$$

#### *Virtual Work and Power Loss*

The virtual work contribution from the total friction moment is obtained through a multiplication by variation of the relative rotation between the two parts

$$\delta W = M_f \cdot \delta \theta$$

The energy dissipation rate caused by the friction moment can be written as

$$Q_f = -M_f \cdot \dot{\theta}$$

### CYLINDRICAL JOINT

Most of the theory is the same as for the hinge joint. Any differences are described here.

#### *Friction Force and Moment*

The total friction force and total friction moment are defined as:

$$F_f = -\min(\mu N + F_{f, r}, F_{f, \max}) \sin(\lambda) \left( 1 - e^{-\frac{|\dot{u}|}{v_0}} \right)$$

$$M_f = -\min(\mu N + F_{f, r}, F_{f, \max}) \cos(\lambda) \left( 1 - e^{-\frac{|\dot{\theta}|}{(v_0/r)}} \right) r$$

$$\tan(\lambda) = \frac{\dot{u}}{r\dot{\theta}}$$

where  $r$  is the entered cylinder radius.

#### *Virtual Work and Power Loss*

The virtual work contributions of the total friction force and the total friction moment are obtained through a multiplication by the variations of the relative displacement and relative rotation between the two parts, respectively.

$$\delta W = F_f \cdot \delta u + M_f \cdot \delta \theta$$

The energy dissipation rate caused by friction can be written as

$$Q_f = -F_f \cdot \dot{u} - M_f \cdot \dot{\theta}$$

### SCREW JOINT

Most of the theory is same as for cylindrical joints. Any differences are described here.

#### *Friction Force and Moment*

The total friction force and total friction moment are defined as:

$$F_f = -\min(\mu N + F_{f,r}, F_{f,max}) \sin(\lambda) \frac{\dot{u}}{|\dot{u}|} \left( 1 - e^{-\frac{|\dot{u}|}{v_0}} \right)$$

$$M_f = -\min(\mu N + F_{f,r}, F_{f,max}) \cos(\lambda) \frac{\dot{u}}{|\dot{u}|} \left( 1 - e^{-\frac{|\dot{\theta}|}{(v_0/r)}} \right) r$$

$$\tan(\lambda) = \frac{np}{2\pi r}$$

where  $n$  is the number of starts,  $p$  is the pitch, and  $r$  is the screw radius.

#### Normal Force

**Case-1.1:** The normal force computed from attachment reaction forces is:

$$N = \sum |\mathbf{F}_a \cdot \mathbf{n}| + \sum |\mathbf{F}_a \cdot \mathbf{e}| \cos(\lambda)$$

**Case-1.2:** The normal force computed through joint forces and moments is:

$$N = |\mathbf{F}_j - (\mathbf{F}_j \cdot \mathbf{e})\mathbf{e}| + |\mathbf{F}_j \cdot \mathbf{e}| \cos(\lambda) + |\alpha_{12}M_{j12}| + |\alpha_{13}M_{j13}|$$

#### PLANAR JOINT

Most of the theory is same as for prismatic joints. Any differences are described here.

There are two ways to model the friction force in a planar joint. In the first approach, the total friction force is applied at the center of joint. This is similar to other joint types, but in this approach, no friction moment is caused by the relative rotation. In the second approach, the distributed friction forces are applied on the selected boundaries, and hence there is an effective friction moment for relative rotation.

#### Friction Force: Total Force at Center of Joint

The total friction force is defined as:

$$\mathbf{F}_f = -\min(\mu N + F_{f,r}, F_{f,max}) \frac{\dot{\mathbf{u}}_r}{|\dot{\mathbf{u}}_r|} \left( 1 - e^{-\frac{|\dot{\mathbf{u}}_r|}{v_0}} \right)$$

where  $\dot{\mathbf{u}}_r$  is the relative velocity vector in the joint.

### Friction Force: Distributed Force on Selected Boundaries

The distributed friction force is defined as:

$$\mathbf{f}_f = -\min(\mu N + F_{f,r}, F_{f,max}) \frac{\dot{\mathbf{u}}_{rt}}{|\dot{\mathbf{u}}_{rt}| A} \left( 1 - e^{-\frac{|\dot{\mathbf{u}}_{rt}|}{v_0}} \right)$$

$$\mathbf{u}_{rt} = \mathbf{u}_r + (R_{dst} - R_{src})(\mathbf{X} - \mathbf{X}_c)$$

where  $\mathbf{X}$  is the material coordinates on the selected boundaries,  $\mathbf{X}_c$  is the center of joint, and  $A$  is the total contact area.

### Normal Force

**Case-1.2:** The normal force computed through joint forces and moments is:

$$N = |\mathbf{F}_j \cdot \mathbf{e}| + |\alpha_{12} M_{j12}| + |\alpha_{13} M_{j13}|$$

### Virtual Work and Power Loss: Total Force at Center of Joint

The virtual work contribution from the total friction force is obtained through a multiplication by the variation of the relative displacement vector between the two parts:

$$\delta W = \mathbf{F}_f \cdot \delta \mathbf{u}_r$$

The energy dissipation rate caused by friction can be written as

$$Q_f = -\mathbf{F}_f \cdot \dot{\mathbf{u}}_r$$

### Virtual Work and Power Loss: Distributed Force on Selected Boundaries

The virtual work contribution of the distributed friction force is obtained through a multiplication by the variation of the relative displacement vector between the two parts on the selected boundaries.

$$\delta W = \int \mathbf{f}_f \cdot \delta \mathbf{u}_{rt} dA$$

The energy dissipation rate caused by friction can be written as:

$$Q_f = \int \mathbf{f}_f \cdot \dot{\mathbf{u}}_{rt} dA$$

## BALL JOINT

Most of the theory is same as in for hinge joints. Any differences are described here.

### Friction Moment

The total friction moment is defined as

$$\mathbf{M}_f = -\min(\mu N + F_{f,r}, F_{f,max}) \frac{\dot{\theta}}{|\dot{\theta}|} \left( 1 - e^{-\frac{|\dot{\theta}|}{(v_0/r)}} \right) \frac{\pi r}{4}$$

where  $\dot{\theta}$  is the relative angular velocity vector at joint and  $r$  is the ball radius. The factor  $\pi r/4$  is the effective ball radius.

### Normal Force

**Case-1.2:** The normal force computed through joint forces is:

$$N = |\mathbf{F}_j|$$

### Virtual Work and Power Loss

The virtual work contribution from the total friction moment is obtained through a multiplication by the variation of the relative rotation between the two parts,

$$\delta W = 2\{0, \mathbf{M}_f\}^T \otimes \{a_r, \mathbf{b}_r\}^T \cdot \{\delta a_r, \delta \mathbf{b}_r\}^T$$

where  $\{a_r, \mathbf{b}_r\}$  is the relative quaternion in the joint.

The energy dissipation rate caused by friction can be written as

$$Q_f = -\mathbf{M}_f \cdot \dot{\theta}$$

### Theory for Joint Forces and Moments Computation

---

The joint forces and moments are evaluated in the global spatial coordinate system as well as in a joint local coordinate system. The joint forces and moments are referred to the center of joint in the current position. For the joints that have translational degrees of freedom, the center of joint should be interpreted as fixed to the source side.

## SUMMING REACTION FORCES OVER THE BOUNDARIES

This is the default method for computing the joint forces and moments when flexible domains are attached to the joint. The computation can be done on either the source or destination attachment. The convention is that the force and moment are the ones

acting on the attachment from the joint. The sign changes if you switch between using the source or destination attachment for the evaluation.

The joint forces are computed by integrating (that is, a summation over all nodes) the reaction forces on the attached boundaries.

$$\mathbf{F} = \sum_i^{\text{nodes}} \mathbf{F}_i$$

The joint moments are computed first by integrating the cross product of reaction forces and position vector of each node relative to the centroid of the attached boundaries and then transferring it to the center of the joint.

$$\begin{aligned} \mathbf{M}_a &= \sum_i^{\text{nodes}} (\mathbf{R}_a(\mathbf{x}_i - \mathbf{x}_a)) \times \mathbf{F}_i \\ \mathbf{M} &= \mathbf{M}_a + ((\mathbf{x}_a + \mathbf{u}_a) - (\mathbf{x}_c + \mathbf{u}_{c, \text{src}})) \times \mathbf{F} \end{aligned}$$

Here the subscript ‘a’ indicates the centroid of attachment, and ‘c’ indicates the center of joint.  $\mathbf{R}_a$  is the rotation matrix representing the rotation of the attachment, and  $\mathbf{u}$  represents the displacements.

### USING WEAK CONSTRAINTS

When using weak constraints, the joint constraints are applied in a weak form, and the values of the Lagrange multipliers give the joint forces ( $\mathbf{F}_w$ ) and moments ( $\mathbf{M}_w$ ). This works for both flexible and rigid components. The use of Lagrange multipliers can, however, have an effect on the structure of the equation system, which limits the solvers that can be used.

The joint forces and moments are computed as

$$\begin{aligned} \mathbf{F} &= \mathbf{F}_w \\ \mathbf{M} &= \mathbf{M}_w + (\mathbf{u}_{c, \text{dst}} - \mathbf{u}_{c, \text{src}}) \times \mathbf{F}_w \end{aligned}$$

The correction term for the moment appears in joints that have translational degrees of freedom because  $\mathbf{M}_w$  is computed on the destination attachment, which can have moved relative to the source attachment.

The sign convention in this case is that the forces and moments should be interpreted as acting on the destination attachment.

### USING PENALTY FORMULATION

When using penalty method, the joint constraints are applied using a penalty formulation, and the values of the penalty forces ( $\mathbf{F}_p$ ) and moments ( $\mathbf{M}_p$ ) give the joint forces and moments. This works for both flexible and rigid components. Moreover, this also works for overconstrained rigid-body systems in which the weak constraints method might not work due to an insufficient number of equations for the Lagrange multipliers.

Since the value of penalty factor directly affects the joint constraints, it should be chosen judiciously. The penalty factor value should be large enough to ensure that the joint constraints are not violated beyond a permissible limit. At the same time the penalty factor should not be too high, since that will make the system of equations numerically ill-conditioned.

The penalty forces ( $\mathbf{F}_p$ ) and moments ( $\mathbf{M}_p$ ) are computed as

$$\begin{aligned}\mathbf{F}_p &= p_j \cdot \mathbf{c}_u \\ \mathbf{M}_p &= (0.1d)^2 p_j \cdot \mathbf{c}_\theta\end{aligned}$$

where  $\mathbf{c}_u$  and  $\mathbf{c}_\theta$  are the translational and rotational violations of the joint constraints, respectively. The penalty factor  $p_j$  is the input having a default value,  $0.1dE_{\text{equ}}$ . In this expression,  $E_{\text{equ}}$  is an effective Young's modulus, and  $d$  is the geometric diagonal of the bounding box of the geometry.

The computation of joint forces and moments using penalty forces and moments is as described above for the weak constraints method:

$$\begin{aligned}\mathbf{F} &= \mathbf{F}_p \\ \mathbf{M} &= \mathbf{M}_p + (\mathbf{u}_{c, \text{dst}} - \mathbf{u}_{c, \text{src}}) \times \mathbf{F}_p\end{aligned}$$

# Cam-Follower Theory

In this section:

- [Cam](#)
- [Follower](#)
- [Closest Point Determination](#)
- [Cam with Boundary Selection](#)
- [Cam with Edge Selection](#)
- [Cam-Follower Connection Force Computation](#)

## *Cam*

---

A cam is defined through a set of boundaries or edges that is being followed by a point called “follower”. The cam selection can be defined on rigid as well as on elastic bodies. Thus, cam boundaries or edges can undergo any kind of rigid body motion as well as deformation. When defining a cam selection, it should also be possible to use boundaries or edges that are defined outside the physics interface.

## *Follower*

---

A follower is defined through a point that follows the cam boundaries or edges with certain optional offset. The following options are supported to define a follower point:

- Centroid of selected entities (domains, boundaries, edges, or points)
- Geometric point (single point)

The follower point can be a part of rigid or elastic bodies. In case there is a requirement of multiple points to follow the cam boundaries or edges, multiple instances of this feature should be used.

## *Closest Point Determination*

---

A general extrusion operator is defined in order to find a point on the cam boundaries or edges that is closest to the follower point. The general extrusion operator used has the following settings:

- Source selection: Cam boundaries or edges

- Destination map: Spatial position of the follower point
- Source frame: Spatial frame
- Mesh search method: Closest point

### *Cam with Boundary Selection*

---

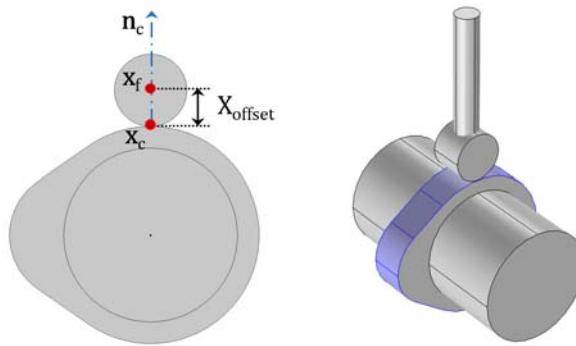


Figure 3-1: Example of a cam-follower with cam having boundary selection.

#### **CONNECTION CONSTRAINT**

The following constraint is added in order to impose the cam and follower connection:

$$\text{gap} = 0$$

In case the connection is conditionally active, the constraint expression is modified to the following:

$$\text{gap} \cdot (\text{slip} \leq \Delta) = 0$$

Here  $\Delta$  is the connection tolerance. The **gap** and **slip** variables used in the connection constraint are defined in the following section.

#### **GAP AND SLIP DEFINITION**

The **gap** variable is a measure of distance between the follower point and its closest point on cam boundaries in the spatial normal direction.

The **slip** variable is a measure of distance between the follower point and its closest point on cam boundaries. Hence it is used to activate or deactivate connection constraint based on the specified connection tolerance.

The gap and slip between the follower point and cam boundaries can be defined as:

$$\text{gap} = (\mathbf{x}_f - \mathbf{x}_c) \cdot \mathbf{n}_c - X_{\text{offset}}$$

$$\text{slip} = \|\mathbf{x}_f - \mathbf{x}_c\| - X_{\text{offset}}$$

Here

- $\mathbf{x}_f$  is the spatial position of the follower point.
- $\mathbf{x}_c$  is the spatial position of a closet point on the cam boundaries.
- $\mathbf{n}_c$  is the spatial normal of a closet point on the cam boundaries.
- $X_{\text{offset}}$  is the initial offset/gap between the follower point and cam boundaries in the normal direction.

### *Cam with Edge Selection*

---

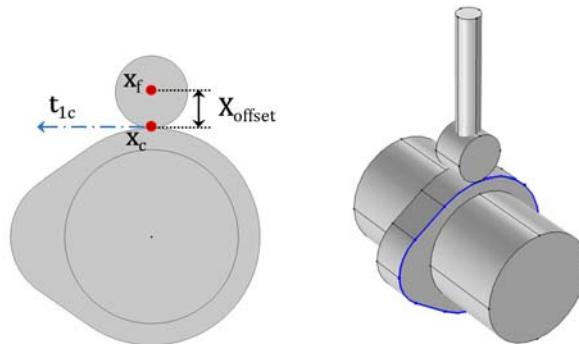


Figure 3-2: Example of a cam-follower with cam having edge selection.

#### **CONNECTION CONSTRAINT**

The following constraint is added in order to impose the cam and follower connection:

$$\mathbf{gap} = \mathbf{0}$$

In case the connection is conditionally active, the constraint expression is modified to the following:

$$\mathbf{gap} \cdot (\text{slip} \leq \Delta) = \mathbf{0}$$

Here  $\Delta$  is the connection tolerance. The **gap** and slip used in the connection constraint are defined in the following section.

#### **GAP AND SLIP DEFINITION**

The **gap** variable is a measure of displacement vector between the follower point and its closest point on cam boundaries.

The **slip** variable is a measure of distance between the follower point and its closest point on cam boundaries in spatial tangent direction. Hence it is used to activate or deactivate connection constraint based on the specified connection tolerance.

The **gap** and slip between the follower point and cam edges can be defined as:

$$\mathbf{gap} = (\mathbf{x}_f - \mathbf{x}_c) - \mathbf{x}_{\text{offset}} - \text{slip} \cdot \mathbf{t}_{1c}$$

$$\mathbf{x}_{\text{offset}} = \mathbf{R}_c \cdot (\mathbf{X}_{\text{offset}} - (\mathbf{X}_{\text{offset}} \cdot \mathbf{t}_{10c}) \mathbf{t}_{10c})$$

$$\text{slip} = (\mathbf{x}_f - \mathbf{x}_c) \cdot \mathbf{t}_{1c}$$

Here

- $\mathbf{x}_f$  is the spatial position of the follower point.
- $\mathbf{x}_c$  is the spatial position of a closet point on the cam edges.
- $\mathbf{t}_{1c}$  is the spatial tangent of a closet point on the cam edges.
- $\mathbf{X}_{\text{offset}}$  is the initial offset or gap vector between the follower point and cam edges.
- $\mathbf{t}_{10c}$  is the material tangent of a closet point on the cam edges.
- $\mathbf{R}_c$  is the rotation matrix of the cam.

#### *Cam-Follower Connection Force Computation*

---

The cam-follower connection constraints can be imposed in several forms:

- Pointwise constraint
- Weak constraint
- Penalty method

The connection constraint enforces a continuous contact between the cam and follower; however, in order to find out whether the follower actually remains continuously in contact with the cam, the connection force can be computed. A

negative value of the connection force throughout the rotation of cam indicates the continuous contact between the cam and follower.

The pointwise constraint is the default form of the cam-follower connection constraints. In this form of constraint, the reaction forces representing the connection forces are not computed. In order to find connection forces, weak constraints or a penalty method can be used.

#### **USING WEAK CONSTRAINTS**

When using weak constraints, the connection constraints are applied in a weak form, and the values of the Lagrange multipliers give the connection forces ( $\mathbf{F}_w$ ). The use of Lagrange multipliers can, however, have an effect on the structure of the equation system, which limits the solvers that can be used.

#### **USING PENALTY FORMULATION**

When using penalty method, the connection constraints are applied using penalty formulation, and the values of the penalty forces ( $\mathbf{F}_p$ ) give the connection forces.

Since the value of penalty factor directly affects the connection constraints, it should be chosen judiciously. The penalty factor value should be large enough to ensure that the connection constraints are not violated beyond a permissible limit. At the same time the penalty factor should not be too high, because that will make the system of equations numerically ill-conditioned.

The penalty forces ( $\mathbf{F}_p$ ) are computed as

$$\mathbf{F}_p = p_c \cdot \mathbf{gap}$$

where  $p_c$  is the penalty factor input having a default value,  $0.1dE_{\text{equ}}$ . In this expression,  $E_{\text{equ}}$  is an effective Young's modulus, and  $d$  is the geometric diagonal of the bounding box of the geometry.

# Gear Theory

In this section:

- Spur Gear Theory
- Helical Gear Theory
- Bevel Gear Theory
- Worm Gear Theory
- Spur Rack Theory
- Helical Rack Theory

## *Spur Gear Theory*

---

The **Spur Gear** is defined as a rigid body. A finite stiffness for the gear mesh (or gear tooth) can be specified in the **Gear Pair** node while connecting to other gears. Similar to the rigid domain, the spur gear is a material model, which is mutually exclusive to all other material models. The only material property needed is the mass density.

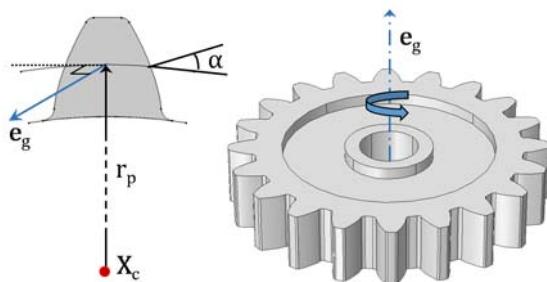


Figure 3-3: Sketch of an external spur gear, showing the tooth cross section and various parameters, such as center of rotation, gear axis, pitch radius, and pressure angle.

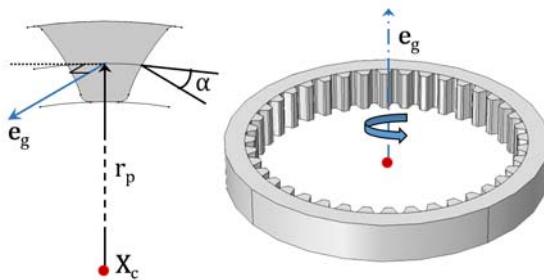


Figure 3-4: Sketch of an internal spur gear, showing the tooth cross section and various parameters such as center of rotation, gear axis, pitch radius, and pressure angle.



For details about the theory of a rigid body, see the [Rigid Domain Model](#) section in the *Structural Mechanics Module User's Guide*.

### CONNECTION TO OTHER GEARS

A spur gear can be connected to the following gear types:

- Another [Spur Gear](#) or a [Helical Gear](#) through the [Gear Pair](#) node.
- A [Worm Gear](#) through the [Worm and Wheel](#) node.
- A [Spur Rack](#) or a [Helical Rack](#) through the [Rack and Pinion](#) node.

A single gear can be connected to one or more gears.

### MOUNTING METHODS

The motion of a spur gear can be constrained directly using its subnode. Alternatively, it can be mounted on a shaft. There are several ways through which a gear can be mounted on a shaft. Some of the commonly used methods are as follows:

- Directly mounted on a flexible shaft
- Mounted on a rigid or flexible shaft through a rigid or elastic fixed joint
- Mounted on a rigid or flexible shaft through a rigid or elastic hinge joint
- Mounted on a ground (a fixed shaft) through a rigid or elastic hinge joint

### CENTER OF ROTATION

Gears are defined through rigid body degrees of freedom. These degrees of freedom are created at the center of rotation and the rotation is interpreted about this point. By

default, the center of rotation is set to the center of mass, but there are other ways to define it explicitly. This is the point where you interpret the forces and moment that act on the gear due to meshing with other gears.

### GEAR AXIS

The gear axis is the axis of rotation of the gear that passes through the center of rotation. This axis is used to create the gear's local coordinate system in the Gear Pair node. The gear rotation, a degree of freedom in the Gear Pair node, is also interpreted about this axis.

### GEAR PROPERTIES

The following quantities are required to define a spur gear:

- Number of teeth ( $n$ )
- Pitch diameter ( $d_p$ )
- Pressure angle ( $\alpha$ )

The transverse module of the gear is defined as:

$$m_t = \frac{d_p}{n}$$

The normal module of the gear is defined as:

$$m_n = m_t$$

### *Helical Gear Theory*

---

The [Helical Gear](#) model is defined as a rigid body. A finite stiffness for the gear mesh (or gear tooth) can be specified in the **Gear Pair** node while connecting to other gears. Similar to the rigid domain, the helical gear is a material model, which is mutually exclusive to all other material models. The only material property needed is the mass density.

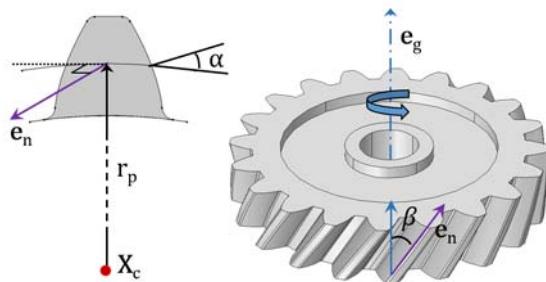


Figure 3-5: Sketch of an external helical gear, showing the tooth cross section and various parameters such as center of rotation, gear axis, pitch radius, pressure angle, and helix angle.

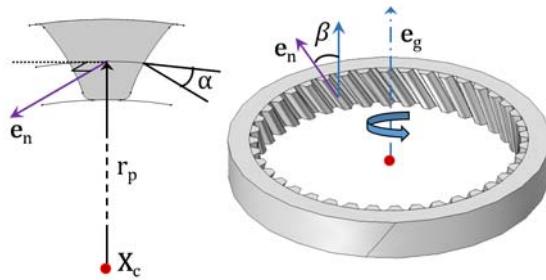


Figure 3-6: Sketch of an internal helical gear, showing the tooth cross section and various parameters such as center of rotation, gear axis, pitch radius, pressure angle, and helix angle.



For details about the theory of a rigid body, see the [Rigid Domain Model](#) section in the *Structural Mechanics Module User's Guide*.



The following section is an addition to the [Spur Gear Theory](#) section, which is also applicable for the helical gear.

### GEAR PROPERTIES

The additional parameter helix angle ( $\beta$ ) is required to define a helical gear. The positive helix angle is interpreted as right-handed and the negative helix angle is interpreted as left-handed.

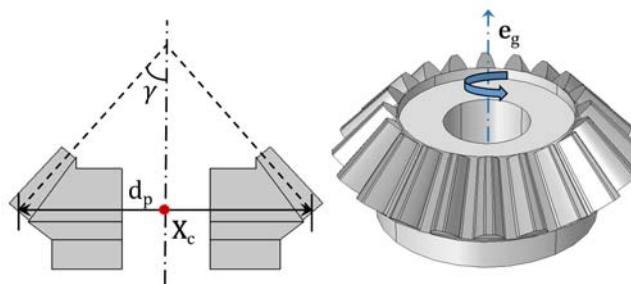
The normal module of the gear is defined as:

$$m_n = m_t \cos \beta$$

### *Bevel Gear Theory*

---

The [Bevel Gear](#) model is defined as a rigid body. A finite stiffness for the gear mesh (or gear tooth) can be specified in the **Gear Pair** node while connecting to other gears. Similar to the rigid domain, the bevel gear is a material model, which is mutually exclusive to all other material models. The only material property needed is the mass density.



*Figure 3-7: Sketch of a bevel gear, showing the gear cross section and various parameters such as center of rotation, gear axis, pitch diameter, and cone angle.*



For details about the theory of a rigid body, see the [Rigid Domain Model](#) in the *Structural Mechanics Module User's Guide*.



The following section is an addition to the [Spur Gear Theory](#) section, some of which is also applicable for the bevel gear.

#### **CONNECTION TO OTHER GEARS**

A bevel gear can be connected to another [Bevel Gear](#) through the **Gear Pair** node. A single gear can be connected to one or more gears.

#### **GEAR PROPERTIES**

The additional parameter cone angle ( $\gamma$ ) is required to define a bevel gear. The positive cone angle results in a decrease in diameter along the gear axis.

## Worm Gear Theory

---

The **Worm Gear** model is defined as a rigid body. A finite stiffness for the gear mesh (or gear tooth) can be specified in the **Worm and Wheel** node while connecting to a wheel. Similar to the rigid domain, the worm gear is a material model, which is mutually exclusive to all other material models. The only material property needed is the mass density.

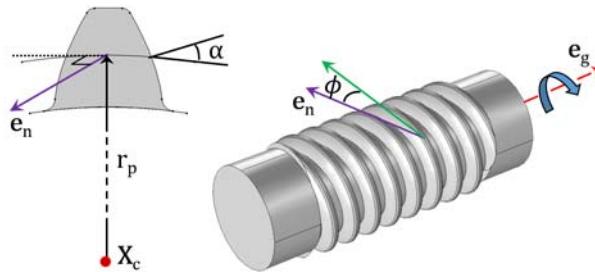


Figure 3-8: Sketch of a worm gear, showing the tooth cross-section and various parameters such as center of rotation, gear axis, pitch radius, pressure angle, and lead angle.



For details about the theory of a rigid body, see the [Rigid Domain Model](#) section in the *Structural Mechanics Module User's Guide*.



The following section is an addition to the [Spur Gear Theory](#), section some of which is also applicable for the worm gear.

### CONNECTION TO OTHER GEARS

A worm gear can be connected to a [Helical Gear](#), or [Spur Gear](#) through the [Worm and Wheel](#) node. A single gear can be connected to one or more gears.

### GEAR PROPERTIES

The following quantities are required to define a worm gear.

- Number of starts ( $n$ )
- Pitch diameter ( $d_p$ )
- Pressure angle ( $\alpha$ )

- Lead angle ( $\phi$ )
- Handedness: Right or left

The helix angle for the right-handed worm gear is defined as:

$$\beta = \frac{\pi}{2} - \phi$$

The helix angle for the left-handed worm gear is defined as:

$$\beta = -\left(\frac{\pi}{2} - \phi\right)$$

The transverse module of the gear is defined as:

$$m_t = \frac{d_p}{n}$$

The normal module of the gear is defined as:

$$m_n = m_t \cos \beta$$

### *Spur Rack Theory*

---

The **Spur Rack** or straight rack is defined as a rigid body. A finite stiffness for the rack mesh (or rack tooth) can be specified in the **Rack and Pinion** node while connecting to a pinion. Similar to the rigid domain, the spur rack is a material model, which is

mutually exclusive to all other material models. The only material property needed is the mass density.

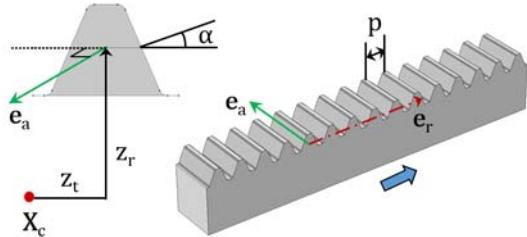


Figure 3-9: Sketch of a spur rack, showing the tooth cross-section and various parameters such as center of rotation, rack axis, pitch, radial offset, tangential offset, and pressure angle.



For details about the theory of a rigid body, see the [Rigid Domain Model](#) section in the *Structural Mechanics Module User's Guide*.



The following section is an addition to the [Spur Gear Theory](#) section, some of which is also applicable for the spur rack.

## CONNECTION TO OTHER GEARS

A spur rack can be connected to a [Spur Gear](#) or to a [Helical Gear](#) through the [Rack and Pinion](#) node. A single rack can be connected to one or more gears.

## RACK AXIS

The rack axis is the axis of translation of the rack that passes through the center of rotation. This axis is used to create the rack local coordinate system in the rack and pinion node. The rack displacement, a degree of freedom in the rack and pinion node, is also interpreted about this axis.

## RACK PROPERTIES

The following quantities are required to define a spur rack.

- Pitch ( $p$ )
- Pressure angle ( $\alpha$ )

The normal pitch of the rack is defined as:

$$p_n = p$$

## Helical Rack Theory

The **Helical Rack** is defined as a rigid body. A finite stiffness for the rack mesh (or rack tooth) can be specified in the **Rack and Pinion** node while connecting to a pinion. Similar to the rigid domain, the helical rack is a material model, which is mutually exclusive to all other material models. The only material property needed is the mass density.

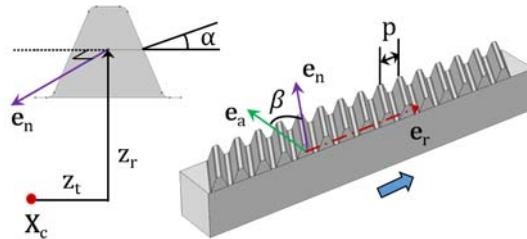


Figure 3-10: Sketch of a helical rack, showing the tooth cross-section and various parameters such as center of rotation, rack axis, pitch, radial offset, tangential offset, pressure angle, and helix angle.



For details about the theory of a rigid body, see the [Rigid Domain Model](#) section in the *Structural Mechanics Module User's Guide*.



The following section is an addition to the [Spur Rack Theory](#) section, which is also applicable for the helical rack.

### RACK PROPERTIES

The additional parameter helix angle ( $\beta$ ) is required to define a helical rack.

The normal pitch of the rack is defined as:

$$p_n = p \cos \beta$$

# Gear Pair Theory

In this section:

- Theory for Gear Pairs
- Theory for Worm and Wheel
- Theory for Rack and Pinion
- Theory for Gear Elasticity
- Theory for Transmission Error
- Theory for Backlash
- Theory for Gear Friction

## *Theory for Gear Pairs*

---

The **Gear Pair** node connects two spur gears, helical gears, or bevel gears in such a way that at the contact point, they have no relative motion along the line of action. The remaining displacements and rotations of both gears are independent of each other.

In case of a line contact model, one additional constraint is added to restrict the relative rotation about a line joining the two gear centers. When friction is included for a gear pair, frictional forces in the plane perpendicular to the line of action are added on both the gears.

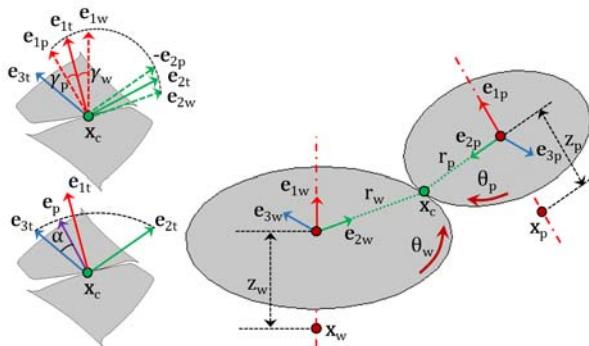


Figure 3-11: Sketch of a gear pair where the line of action is computed by the clockwise rotation of the tangent.

The degrees of freedom at the gear pair are  $\theta_{wh}$  and  $\theta_{pn}$ . They are defined as the rotation of the wheel and pinion about the first axis of their respective local coordinate system.

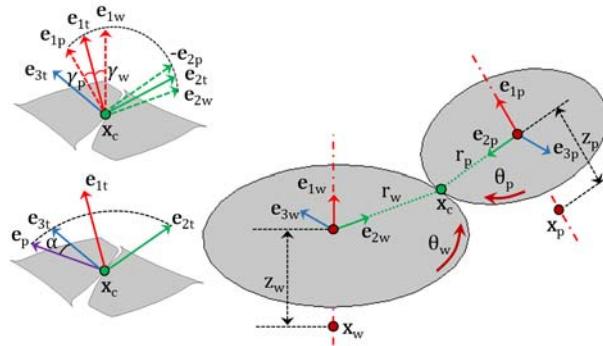


Figure 3-12: Sketch of a gear pair where the line of action is computed by the counterclockwise rotation of the tangent.

The rest of this section discusses the following topics:

- Gear Pair Compatibility Criteria
- Gear Local Coordinate System
- Gear Tooth Coordinate System
- Line of Action
- Gear Ratio
- Contact Point Position and Offset
- Gear Pair Constraints
- Contact Forces and Moments

#### GEAR PAIR COMPATIBILITY CRITERIA

In a **Gear Pair** node, you can select any two gears defined for the model. However, for the correct tooth meshing, a set of gears must fulfill the compatibility criteria.

##### Normal Module

The normal module of both gears must be the same.

$$m_{n, wh} = m_{n, pn}$$

##### Pressure Angle

The pressure angle of both gears must be the same.

$$\alpha_{wh} = \alpha_{pn}$$

### Configuration

For parallel axis helical gears, the sum of the helix angles must be zero.

$$\beta_{wh} + \beta_{pn} = 0$$

For crossed axis helical gears, the sum of the helix angle must be nonzero.

$$\beta_{wh} + \beta_{pn} \neq 0$$

### GEAR LOCAL COORDINATE SYSTEM

The local coordinate system for each gear is created using the gear axis and the center of rotation of both gears. This local coordinate system is attached to the gear; however, it does not rotate with the gear local rotation.

#### Initial Local Coordinate System

The first axis for the wheel ( $\mathbf{e}_{10, wh}$ ) and pinion ( $\mathbf{e}_{10, pn}$ ) is an input on the respective nodes. In 2D, it is assumed to be the out-of-plane direction.

For a parallel or intersecting configuration, the second axis is defined as:

$$\begin{aligned}\mathbf{e}_{20, wh} &= \mathbf{e}_{0, cd} - (\mathbf{e}_{0, cd} \cdot \mathbf{e}_{10, wh})\mathbf{e}_{10, wh} \\ \mathbf{e}_{20, pn} &= -(\mathbf{e}_{0, cd} - (\mathbf{e}_{0, cd} \cdot \mathbf{e}_{10, pn})\mathbf{e}_{10, pn})\end{aligned}$$

$$\mathbf{e}_{0, cd} = \frac{\mathbf{x}_{c, pn} - \mathbf{x}_{c, wh}}{\|\mathbf{x}_{c, pn} - \mathbf{x}_{c, wh}\|}$$

where  $\mathbf{x}_{c, wh}$  and  $\mathbf{x}_{c, pn}$  are the center of rotation of the wheel and pinion, respectively.

For a configuration that is neither parallel nor intersecting, the second axis is defined as:

$$\begin{aligned}\mathbf{e}_{20, wh} &= \mathbf{e}_{20} \operatorname{sgn}(\mathbf{e}_{20} \cdot \mathbf{e}_{0, cd}) \\ \mathbf{e}_{20, pn} &= -\mathbf{e}_{20, wh}\end{aligned}$$

$$\mathbf{e}_{20} = \mathbf{e}_{10, wh} \times \mathbf{e}_{10, pn}$$

In 2D:

$$\begin{aligned}\mathbf{e}_{20, wh} &= \mathbf{e}_{0, cd} \\ \mathbf{e}_{20, pn} &= -\mathbf{e}_{0, cd}\end{aligned}$$

The third axis is defined as follows:

$$\begin{aligned}\mathbf{e}_{30, \text{wh}} &= \mathbf{e}_{10, \text{wh}} \times \mathbf{e}_{20, \text{wh}} \\ \mathbf{e}_{30, \text{pn}} &= \mathbf{e}_{10, \text{pn}} \times \mathbf{e}_{20, \text{pn}}\end{aligned}$$

#### *Local Coordinate System*

The first axis is defined as follows:

$$\begin{aligned}\mathbf{e}_{1, \text{wh}} &= R_{\text{wh}} \cdot \mathbf{e}_{10, \text{wh}} \\ \mathbf{e}_{1, \text{pn}} &= R_{\text{pn}} \cdot \mathbf{e}_{10, \text{pn}}\end{aligned}$$

The second axis is defined as follows:

$$\begin{aligned}\mathbf{e}_{2, \text{wh}} &= (R_{\text{wh}} \cdot \mathbf{e}_{20, \text{wh}}) \cos \theta_{\text{wh}} - (R_{\text{wh}} \cdot \mathbf{e}_{30, \text{wh}}) \sin \theta_{\text{wh}} \\ \mathbf{e}_{2, \text{pn}} &= (R_{\text{pn}} \cdot \mathbf{e}_{20, \text{pn}}) \cos \theta_{\text{pn}} - (R_{\text{pn}} \cdot \mathbf{e}_{30, \text{pn}}) \sin \theta_{\text{pn}}\end{aligned}$$

The third axis is defined as follows:

$$\begin{aligned}\mathbf{e}_{3, \text{wh}} &= (R_{\text{wh}} \cdot \mathbf{e}_{20, \text{wh}}) \sin \theta_{\text{wh}} + (R_{\text{wh}} \cdot \mathbf{e}_{30, \text{wh}}) \cos \theta_{\text{wh}} \\ \mathbf{e}_{3, \text{pn}} &= (R_{\text{pn}} \cdot \mathbf{e}_{20, \text{pn}}) \sin \theta_{\text{pn}} + (R_{\text{pn}} \cdot \mathbf{e}_{30, \text{pn}}) \cos \theta_{\text{pn}}\end{aligned}$$

where  $R_{\text{wh}}$  and  $R_{\text{pn}}$  are the rotation matrices of the wheel and the pinion, respectively. The angles  $\theta_{\text{wh}}$  and  $\theta_{\text{pn}}$  are the rotations of the wheel and pinion about the first axis of their respective local coordinate system.

#### **GEAR TOOTH COORDINATE SYSTEM**

The gear tooth coordinate system is defined by rotating the gear local coordinate system with the helix angle and cone angle.

$$\begin{aligned}\mathbf{e}_{it, \text{wh}} &= T_{\text{wh}}^T \cdot \mathbf{e}_{i, \text{wh}} & i &= 1, 2, 3 \\ \mathbf{e}_{it, \text{pn}} &= T_{\text{pn}}^T \cdot \mathbf{e}_{i, \text{pn}} & i &= 1, 2, 3\end{aligned}$$

where  $T_{\text{wh}}$  and  $T_{\text{pn}}$  are the tooth transformation matrix for the wheel and pinion, respectively.

The gear-to-tooth transformation matrix is defined as:

$$T = \begin{bmatrix} \cos\gamma\cos\beta & -\sin\gamma & \cos\gamma\sin\beta \\ \sin\gamma\cos\beta & \cos\gamma & \sin\gamma\sin\beta \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}$$



For the pinion tooth coordinate system, the sign of the second axis ( $\mathbf{e}_{2t,pn}$ ) and third axis ( $\mathbf{e}_{3t,pn}$ ) are reversed to match the directions with the wheel tooth coordinate system.

#### LINE OF ACTION

The line of action is the normal direction of the gear tooth surface at the contact point on the pitch circle. This is the direction along which the motion is transferred from one gear to another gear. It is defined by rotating the third axis of the tooth coordinate system ( $\mathbf{e}_{3t}$ ) about the first axis of the tooth coordinate system ( $\mathbf{e}_{1t}$ ) with the pressure angle ( $\alpha$ ).

The line of action, also known as the pressure angle direction, is defined as follows:

For clockwise rotation of the third axis:

$$\mathbf{e}_{pa} = \mathbf{e}_{2t,wh} \sin\alpha + \mathbf{e}_{2t,wh} \cos\alpha$$

For counterclockwise rotation of the third axis:

$$\mathbf{e}_{pa} = -\mathbf{e}_{2t,wh} \sin\alpha + \mathbf{e}_{2t,wh} \cos\alpha$$

#### GEAR RATIO

The gear ratio ( $g_r$ ) of a gear pair is defined as the ratio of angular velocities of the wheel ( $\omega_{wh}$ ) and the pinion ( $\omega_{pn}$ ):

$$g_r = \frac{\omega_{wh}}{\omega_{pn}}$$

It can also be written as the ratio of number of teeth of the pinion ( $n_{pn}$ ) and the wheel ( $n_{wh}$ ):

$$g_r = \frac{n_{pn}}{n_{wh}}$$

## CONTACT POINT POSITION AND OFFSET

### Contact Point Position

The point of contact ( $\mathbf{x}_{cp}$ ) on a gear in its local coordinate system can be defined as:

$$\begin{aligned}\mathbf{x}_{cp, wh} &= \mathbf{x}_{c, wh} + \mathbf{u}_{wh} + z_{wh} \mathbf{e}_{1, wh} + (r_{wh} - z_{wh} \tan \gamma_{wh}) \mathbf{e}_{2, wh} \\ \mathbf{x}_{cp, pn} &= \mathbf{x}_{c, pn} + \mathbf{u}_{pn} + z_{pn} \mathbf{e}_{1, pn} + (r_{pn} - z_{pn} \tan \gamma_{pn}) \mathbf{e}_{2, pn}\end{aligned}$$

where, for the wheel and pinion, respectively:

- $\mathbf{x}_{cp, wh}$  and  $\mathbf{x}_{cp, pn}$  are the positions of the contact points
- $\mathbf{x}_{c, wh}$  and  $\mathbf{x}_{c, pn}$  are the centers of rotation
- $\mathbf{u}_{wh}$  and  $\mathbf{u}_{pn}$  are the displacement vectors at the center of rotation
- $z_{wh}$  and  $z_{pn}$  are the contact point offsets from the center of rotation in the axial direction
- $r_{wh}$  and  $r_{pn}$  are the pitch radii
- $\gamma_{wh}$  and  $\gamma_{pn}$  are the cone angles

In 2D, the expressions for the position of contact points reduce to:

$$\begin{aligned}\mathbf{x}_{cp, wh} &= \mathbf{x}_{cp, wh} = \mathbf{x}_{c, wh} + \mathbf{u}_{wh} + r_{wh} \mathbf{e}_{2, wh} \\ \mathbf{x}_{cp, pn} &= \mathbf{x}_{cp, pn} + \mathbf{u}_{pn} + r_{pn} \mathbf{e}_{2, pn}\end{aligned}$$

### Contact Point Offset

The contact point offset from the wheel or pinion center of rotation in the axial direction is defined as follows:

For a parallel or intersecting configuration, the contact point offset from the pinion center ( $z_{pn}$ ) is the input. The contact point offset from the wheel center ( $z_{wh}$ ) is defined as:

$$z_{wh} = (\mathbf{x}_{c, pn} - \mathbf{x}_{c, wh} + z_{pn} \mathbf{e}_{10, pn} + (r_{pn} - z_{pn} \tan \gamma_{pn}) \mathbf{e}_{20, pn}) \cdot \mathbf{e}_{10, wh}$$

For a configuration that is neither parallel nor intersecting, the contact point offset from the pinion center ( $z_{pn}$ ) and the contact point offset from the wheel center ( $z_{wh}$ ) are defined as:

$$z_{pn} = \frac{-(\mathbf{x}_{cd} \cdot \mathbf{e}_{10, pn}) - (\mathbf{x}_{cd} \cdot \mathbf{e}_{10, wh}) x_{wh, pn}}{1 - x_{wh, pn}^2}$$

$$z_{wh} = \frac{(\mathbf{x}_{cd} \cdot \mathbf{e}_{10, wh}) - (\mathbf{x}_{cd} \cdot \mathbf{e}_{10, pn})x_{wh, pn}}{1 - x_{wh, pn}^2}$$

where

$$\begin{aligned}\mathbf{x}_{cd} &= \mathbf{x}_{c, pn} - \mathbf{x}_{c, wh} \\ x_{wh, pn} &= \mathbf{e}_{10, wh} \cdot \mathbf{e}_{10, pn}\end{aligned}$$

## GEAR PAIR CONSTRAINTS

### Rotation Constraint

This constraint relates the pinion rotation to the wheel rotation:

$$\theta_{pn} = -\frac{\theta_{wh}}{g_r} + \theta_{el} + \theta_{et} + \theta_{bl}$$

where  $\theta_{el}$ ,  $\theta_{et}$ , and  $\theta_{bl}$  are the transmission error (elasticity), transmission error (static), and transmission error (backlash), respectively.

By default, there is no elasticity, transmission error, or backlash in a gear pair hence all these transmission errors are zero.



For more details about these transmission errors, see the [Theory for Gear Elasticity](#), [Theory for Transmission Error](#), and [Theory for Backlash](#) sections.

### Point Contact Constraint

At the contact point, the relative motion along the line of action must be zero. This can be written as:

$$(\mathbf{x}_{cp, pn} - \mathbf{x}_{cp, wh}) \cdot \mathbf{e}_{pa} = 0$$

### Line Contact Constraint

In case of a line contact model, an additional constraint is added at the second contact point. This can be written as:

$$\left( \mathbf{x}_{cp2, pn} - \mathbf{x}_{cp2, wh} - \frac{w_{pn}}{2} \theta_{cl} \mathbf{e}_{3, pn} \right) \cdot \mathbf{e}_{pa} = 0$$

where  $\mathbf{x}_{cp2,wh}$  and  $\mathbf{x}_{cp2,pn}$  are the position of the second contact point and can be defined as:

$$\left( \mathbf{x}_{cp2,wh} = \text{subst}\left(x_{cp,wh}, z_{wh}, z_{wh} + \frac{w_{pn}}{2} \frac{\cos\gamma_{wh}}{\cos\gamma_{pn}}\right) \right)$$

$$\mathbf{x}_{cp2,pn} = \text{subst}\left(x_{cp,pn}, z_{pn}, z_{pn} + \frac{w_{pn}}{2}\right)$$

Here,  $w_{pn}$  and  $\theta_{cl}$  are the working width of the pinion and the relative rotation about the centerline, respectively. By default, there is no elasticity on a gear pair; hence, the relative rotation about the centerline is zero.

#### *Unique Triad Constraint*

In case of planetary gears, there is a constraint to ensure that the local coordinate system of both the gears are still unique and well defined. This can be written as:

$$\mathbf{e}_{2t,wh} \cdot \mathbf{e}_{3t,pn} = 0$$

## **CONTACT FORCES AND MOMENTS**

#### *Contact Force*

When you compute a contact force using weak constraints, the point contact constraint is implemented in the weak form and the corresponding Lagrange multiplier gives the contact force ( $F_c$ ).

Similarly, when you compute a contact force using a penalty method, the point contact constraint is implemented using the penalty factor and the corresponding penalty force gives the contact force ( $F_c$ ).

For the case of a line contact model, the second contact force ( $F_{c2}$ ) is computed by implementing the line contact constraint in the weak form or by using the penalty factor.

#### *Contact Moment*

When you compute a contact force using weak constraints, the unique triad constraint is implemented in the weak form and the corresponding Lagrange multiplier gives the moment at the contact point ( $M_c$ ).

Similarly, when you compute a contact force computation using a penalty method, the unique triad constraint is implemented using the penalty factor and the corresponding penalty moment gives the moment at the contact point ( $M_c$ ).

### Force and Moment at Gear Centers

The forces at the gear centers are defined as:

$$\mathbf{F}_{wh} = -F_c \mathbf{e}_{pa}$$

$$\mathbf{F}_{pn} = F_c \mathbf{e}_{pa}$$

The moments at the gear centers are defined as:

$$\mathbf{M}_{wh} = (\mathbf{x}_{cp, pn} - \mathbf{x}_{wh} - \mathbf{u}_{wh}) \times (-F_c \mathbf{e}_{pa}) + M_c (\mathbf{e}_{2t, wh} \times \mathbf{e}_{3t, pn})$$

$$\mathbf{M}_{pn} = (\mathbf{x}_{cp, pn} - \mathbf{x}_{pn} - \mathbf{u}_{pn}) \times (F_c \mathbf{e}_{pa}) + (-M_c) (\mathbf{e}_{2t, wh} \times \mathbf{e}_{3t, pn})$$

For a line contact model, additional contributions from the contact force  $F_{c2}$  are added to the forces and moments at the gear centers. These contributions can be defined as:

$$\mathbf{F}_{wh} = -F_{c2} \mathbf{e}_{pa}$$

$$\mathbf{F}_{pn} = F_{c2} \mathbf{e}_{pa}$$

$$\mathbf{M}_{wh} = (\mathbf{x}_{cp2, pn} - \mathbf{x}_{wh} - \mathbf{u}_{wh}) \times (-F_{c2} \mathbf{e}_{pa})$$

$$\mathbf{M}_{pn} = (\mathbf{x}_{cp2, pn} - \mathbf{x}_{pn} - \mathbf{u}_{pn}) \times (F_{c2} \mathbf{e}_{pa})$$

### Theory for Worm and Wheel

---

The worm and wheel node connects a worm gear with a spur or helical gear in such a way that at the contact point they have no relative motion along the line of action. The remaining displacements and rotations of both the gears are independent of each other.

When friction is included on a worm and wheel node, frictional forces in the plane perpendicular to the line of action are added on both the gears.

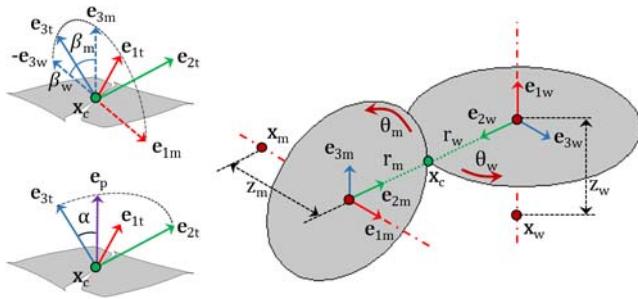


Figure 3-13: Sketch of a worm and wheel pair where line of action is computed by the clockwise rotation of the tangent.

The degrees of freedom at the worm and wheel are  $\theta_{wm}$  and  $\theta_{wh}$ . They are defined as the rotation of the worm and wheel about the first axis of their respective local coordinate system.

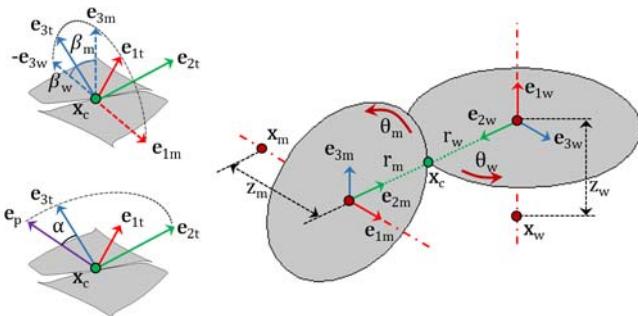


Figure 3-14: The sketch of a worm and wheel pair where line of action is computed by the counterclockwise rotation of the tangent.



In the worm and wheel node, the configuration is always neither parallel nor intersecting and the contact model is point contact.

---

The rest of the theory for the worm and wheel node is similar to the [Theory for Gear Pairs](#).



The gear pair node connects a wheel with a pinion, whereas the worm and wheel node connects a worm with a wheel. To get worm and wheel node specific equations, replace wheel with worm and pinion with wheel everywhere.

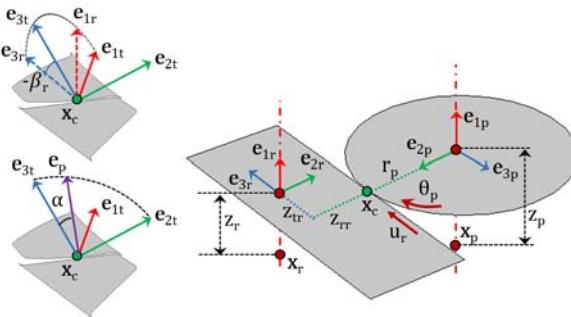
---

### *Theory for Rack and Pinion*

---

The rack and pinion node connects a spur or helical rack with a spur or helical gear in such a way that at the contact point they have no relative motion along the line of action. The remaining displacements and rotations of the rack and gear are independent of each other.

For a line contact model, one additional constraint is added to restrict the relative rotation about a line joining the rack and gear centers. When friction is included on a rack and pinion node, frictional forces in the plane perpendicular to the line of action are added on the rack and pinion.



*Figure 3-15: The sketch of a rack and pinion pair where line of action is computed by the clockwise rotation of the tangent.*

The degrees of freedom at the rack and pinion are  $u_{rk}$  and  $\theta_{pn}$ . They are defined as the displacement of the rack along the third axis and the rotation of the pinion about the first axis of their respective local coordinate system, respectively.

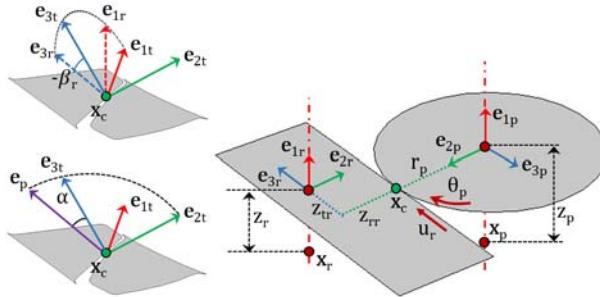


Figure 3-16: The sketch of a rack and pinion pair where line of action is computed by the counterclockwise rotation of the tangent.

---

The rest of the theory for the rack and pinion node is similar to the [Theory for Gear Pairs](#) with the following changes.



The gear pair node connects a wheel with a pinion, whereas the rack and pinion node connects a rack with a pinion. To get rack and pinion node specific equations, replace wheel with rack everywhere.

---

#### GEAR PAIR COMPATIBILITY CRITERIA

##### Normal Pitch

The normal pitch of the rack and pinion must be same.

$$p_{n, rk} = \pi m_{n, pn}$$




---

For more details, refer the [Gear Pair Compatibility Criteria](#) for the gear pair node.

---

## GEAR LOCAL COORDINATE SYSTEM

### Initial Local Coordinate System

The third axis for the rack ( $\mathbf{e}_{30, \text{rk}}$ ) and the first axis of the pinion ( $\mathbf{e}_{10, \text{pn}}$ ) is the input on the respective nodes. In 2D, the first axis of the rack and pinion are assumed to be the out-of-plane direction.

The second axis is defined as follows:

$$\mathbf{e}_{20, \text{rk}} = \mathbf{e}_{20} \text{sgn}(\mathbf{e}_{20} \cdot \mathbf{e}_{0, \text{cd}})$$

$$\mathbf{e}_{20, \text{pn}} = -\mathbf{e}_{20, \text{rk}}$$

$$\mathbf{e}_{20} = \mathbf{e}_{30, \text{rk}} \times \mathbf{e}_{10, \text{pn}}$$

$$\mathbf{e}_{0, \text{cd}} = \frac{\mathbf{x}_{c, \text{pn}} - \mathbf{x}_{c, \text{rk}}}{\|\mathbf{x}_{c, \text{pn}} - \mathbf{x}_{c, \text{rk}}\|}$$

The first axis of rack and the third axis of pinion are defined as follows:

$$\mathbf{e}_{10, \text{rk}} = \mathbf{e}_{20, \text{rk}} \times \mathbf{e}_{30, \text{rk}}$$

$$\mathbf{e}_{30, \text{pn}} = \mathbf{e}_{10, \text{pn}} \times \mathbf{e}_{20, \text{pn}}$$



For more details, refer the [Gear Local Coordinate System](#) for the gear pair node.

## CONTACT POINT POSITION AND OFFSET

### Contact Point Position

The point of contact ( $\mathbf{x}_{\text{cp}}$ ) on a rack in its local coordinate system can be defined as:

$$\mathbf{x}_{\text{cp}, \text{rk}} = \mathbf{x}_{c, \text{rk}} + \mathbf{u}_{\text{rk}} + z_{r, \text{rk}} \mathbf{e}_{1, \text{rk}} + z_{t, \text{rk}} \mathbf{e}_{2, \text{rk}} + (z_{t, \text{rk}} - u_{\text{rk}}) \mathbf{e}_{3, \text{rk}}$$

where  $z_{r, \text{rk}}$  and  $z_{t, \text{rk}}$  are the contact point offset from the center of rotation in the radial and tangential direction, respectively.

In 2D, the expression for the position of contact point reduces to:

$$\mathbf{x}_{\text{cp}, \text{rk}} = \mathbf{x}_{c, \text{rk}} + \mathbf{u}_{\text{rk}} + z_{r, \text{rk}} \mathbf{e}_{2, \text{rk}} + (z_{t, \text{rk}} - u_{\text{rk}}) \mathbf{e}_{3, \text{rk}}$$

### Contact Point Offset

The contact point offset from the pinion center ( $z_{pn}$ ) is the input.

The contact point offset from the rack center in the axial direction ( $z_{rk}$ ) is defined as:

$$z_{rk} = (\mathbf{x}_{c, pn} - \mathbf{x}_{c, rk} + z_{pn} \mathbf{e}_{10, pn}) \cdot \mathbf{e}_{10, rk}$$

The contact point offset from the rack center in the radial direction ( $z_{r,rk}$ ) is defined as:

$$z_{r, rk} = (\mathbf{x}_{c, pn} - \mathbf{x}_{c, rk}) \cdot \mathbf{e}_{20, rk} - r_{pn}$$

The contact point offset from the rack center in the tangential direction ( $z_{t,rk}$ ) is defined as:

$$z_{t, rk} = (\mathbf{x}_{c, pn} - \mathbf{x}_{c, rk} + z_{pn} \mathbf{e}_{10, pn}) \cdot \mathbf{e}_{30, rk}$$

In 2D, the expression reduces to:

$$z_{t, rk} = (\mathbf{x}_{c, pn} - \mathbf{x}_{c, rk}) \cdot \mathbf{e}_{30, rk}$$



For more details, refer the [Contact Point Position and Offset](#) for the gear pair node.

### GEAR PAIR CONSTRAINTS

#### *Translation-Rotation constraint*

This constraint relates the pinion rotation with the rack displacement:

$$\theta_{pn} = -\frac{u_{rk} \cos \beta_{rk}}{r_{pn} \cos \beta_{pn}} + \theta_{el} + \theta_{et} + \theta_{bl}$$



For more details, refer the [Gear Pair Constraints](#) for the gear pair node.

#### *Theory for Gear Elasticity*

The **Gear Elasticity** node can optionally be added on gear pairs. In this node, you can specify the elastic properties of the gear mesh, such as mesh stiffness, mesh damping, and contact ratio. By default, the gear mesh is assumed rigid.

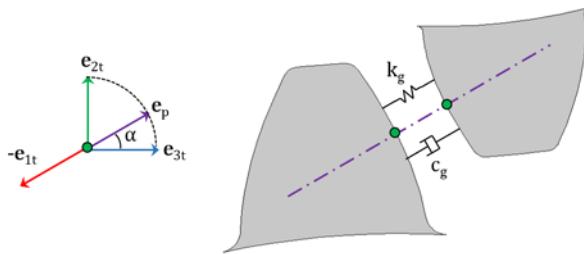


Figure 3-17: A sketch of a pair of teeth, showing the point of contact and the direction in which mesh stiffness and damping are interpreted.

The rest of this section discusses the following topics:

- Degrees of Freedom
- Elastic Moments
- Mesh Stiffness
- Mesh Damping

#### DEGREES OF FREEDOM

In this node, the transmission error (elasticity),  $\theta_{el}$ , is a degree of freedom. This transmission error is added in the rotation constraint of the gear pair.

In the case of a line contact model, one more degree of freedom,  $\theta_{cl}$ , is added. This is the relative rotation about the line joining the centers of two gears. The contribution of this rotation is added in the line contact constraint of the gear pair.

#### ELASTIC MOMENTS

The elastic moment about the first axis of the tooth coordinate system and its virtual work contribution are defined as:

$$M_{el} = -(k_g \theta_{el} + c_g \dot{\theta}_{el}) (\mathbf{e}_{pa} \cdot \mathbf{e}_{3, pn})^2 r_{pn}^2$$

$$\delta W = M_{el} \cdot \delta \theta_{el}$$

In the case of a line contact model, the elastic moment about the second axis of the tooth coordinate system and its virtual work contribution are defined as:

$$M_{cl} = -(k_g \theta_{cl} + c_g \dot{\theta}_{cl}) (\mathbf{e}_{pa} \cdot \mathbf{e}_{3, pn})^2 \frac{w_{pn}^2}{12}$$

$$\delta W = M_{cl} \cdot \delta \theta_{cl}$$

where

- $k_g$  is the mesh stiffness of gear pair
- $c_g$  is the mesh damping of gear pair
- $r_{pn}$  is the pitch radius of pinion
- $w_{pn}$  is the working width of pinion



The mesh damping contribution to the elastic moments will be zero in the case of a stationary analysis.

### MESH STIFFNESS

The mesh stiffness expressions in this section are derived for a **Gear Pair** node.



To get equations that are specific to the **Worm and Wheel** node, replace “wheel” with “worm” and “pinion” with “wheel” everywhere.

To get equations that are specific to the **Rack and Pinion** node, replace “wheel” with “rack” everywhere.

#### *Tooth Stiffness of Individual Gears*

The mesh stiffness of the gear pair is defined as:

$$k_g = \frac{k_{wh}k_{pn}}{k_{wh} + k_{pn}}$$

where  $k_{wh}$  and  $k_{pn}$  are the mesh stiffness of the wheel and pinion, respectively.

The mesh stiffness of the wheel, for different contact ratios, is defined as follows.



The mesh stiffness of the pinion can be derived in a similar manner.

**Contact ratio is constant and set to 1:**

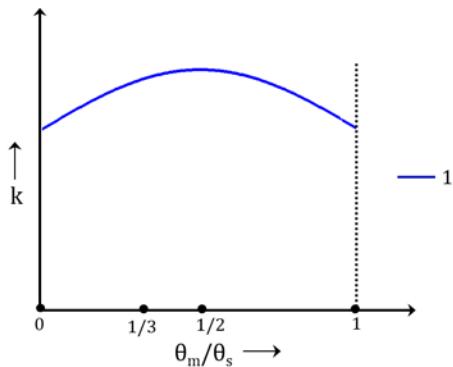


Figure 3-18: The graph shows the variation of the gear tooth stiffness in a mesh cycle.

The mesh stiffness of the wheel is defined as:

$$k_{wh} = k_{t, wh}(\theta_{m, wh})$$

where

- $k_{t, wh}$  is the gear tooth stiffness as function of mesh cycle
- $\theta_{m, wh}$  is the mesh cycle

The mesh cycle for the wheel is defined as:

$$\theta_{m, wh} = \text{mod}(\theta_{wh}, \theta_{s, wh}) \quad \theta_{s, wh} = \frac{2\pi}{n_{wh}}$$

---

The mesh cycle for the rack is different from other gear types and can be defined as:



$$u_{m, rk} = \text{mod}(u_{rk}, p_{rk})$$

**Contact ratio is constant and set to 2:**

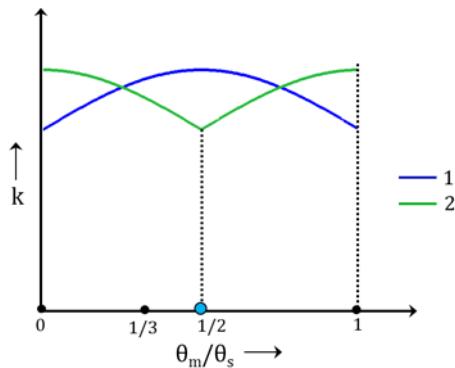


Figure 3-19: The graph shows the variation of the gear tooth stiffness of the first and second teeth in a mesh cycle.

The mesh stiffness of the wheel is defined as:

$$k_{wh} = k_{t1, wh}(\theta_{m, wh}) + k_{t2, wh}$$

where  $k_{t2, wh}$  is the tooth stiffness of the second tooth. It is defined as:

$$k_{t2, wh} = k_{t1, wh}\left(\theta_{m, wh} + \frac{\theta_{s, wh}}{2}\right)$$

**Contact ratio is constant and set to 3:**

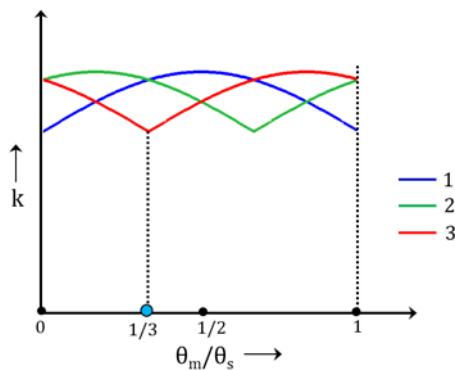


Figure 3-20: The graph shows the variation of the gear tooth stiffness of the first, second, and third teeth in a mesh cycle.

The mesh stiffness of the wheel is defined as:

$$k_{\text{wh}} = k_{t, \text{wh}}(\theta_{m, \text{wh}}) + k_{t2, \text{wh}} + k_{t3, \text{wh}}$$

where  $k_{t2, \text{wh}}$  and  $k_{t3, \text{wh}}$  are the tooth stiffness of the second and third teeth. respectively. They are defined as:

$$k_{t2, \text{wh}} = k_{t, \text{wh}}\left(\theta_{m, \text{wh}} + \frac{\theta_{s, \text{wh}}}{3}\right)$$

$$k_{t3, \text{wh}} = k_{t, \text{wh}}\left(\theta_{m, \text{wh}} + \frac{2\theta_{s, \text{wh}}}{3}\right)$$

**Contact ratio is varying and maximum contact ratio is set to 2:**

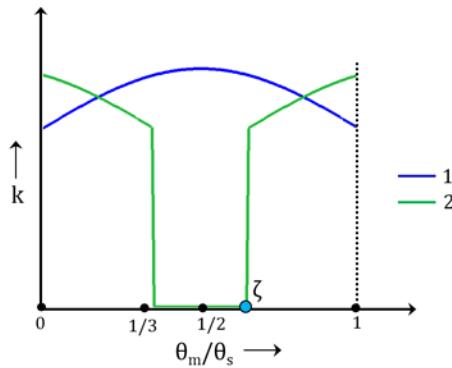


Figure 3-21: The graph shows the variation of the gear tooth stiffness of the first and second teeth in a mesh cycle.

The mesh stiffness of the wheel is defined as:

$$k_{\text{wh}} = k_{t, \text{wh}}(\theta_{m, \text{wh}}) + k_{t2f, \text{wh}} + k_{t2b, \text{wh}}$$

where  $k_{t2f, \text{wh}}$  and  $k_{t2b, \text{pn}}$  are the tooth stiffness of the second forward and backward teeth. They are defined as:

$$k_{t2f, \text{wh}} = (\theta_{m, \text{wh}} < (1 - \zeta)\theta_{s, \text{wh}}) \cdot k_{t, \text{wh}}(\theta_{m, \text{wh}} + \zeta\theta_{s, \text{wh}})$$

$$k_{t2b, \text{wh}} = (\theta_{m, \text{wh}} \geq \zeta\theta_{s, \text{wh}}) \cdot k_{t, \text{wh}}(\theta_{m, \text{wh}} - \zeta\theta_{s, \text{wh}})$$

where  $\zeta$  is the next tooth engagement position in the mesh cycle. The value of  $\zeta$  must be in the following range:

$$\frac{1}{2} \leq \zeta \leq 1$$

**Contact ratio is varying and maximum contact ratio is set to 3:**

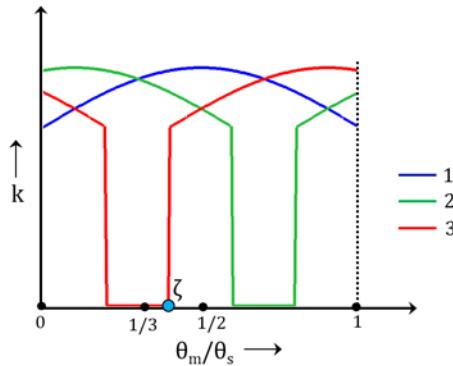


Figure 3-22: The graph shows the variation of the gear tooth stiffness of the first, second, and third teeth in a mesh cycle.

The mesh stiffness of the wheel is defined as:

$$k_{wh} = k_{t, wh}(\theta_m, wh) + k_{t2f, wh} + k_{t2b, wh} + k_{t3f, wh} + k_{t3b, wh}$$

where

- $k_{t2f, wh}$  and  $k_{t2b, pn}$  are the tooth stiffness of the second forward and backward teeth
- $k_{t3f, wh}$  and  $k_{t3b, pn}$  are the tooth stiffness of the third forward and backward teeth

$$\begin{aligned} k_{t2f, wh} &= (\theta_m, wh < (1 - \zeta)\theta_s, wh) \cdot k_{t, wh}(\theta_m, wh + \zeta\theta_s, wh) \\ k_{t2b, wh} &= (\theta_m, wh \geq 2\zeta\theta_s, wh) \cdot k_{t, wh}(\theta_m, wh - 2\zeta\theta_s, wh) \\ k_{t3f, wh} &= (\theta_m, wh < (1 - 2\zeta)\theta_s, wh) \cdot k_{t, wh}(\theta_m, wh + 2\zeta\theta_s, wh) \\ k_{t3b, wh} &= (\theta_m, wh \geq \zeta\theta_s, wh) \cdot k_{t, wh}(\theta_m, wh - \zeta\theta_s, wh) \end{aligned}$$

The value of  $\zeta$  must be in the following range:

$$\frac{1}{3} \leq \zeta \leq \frac{1}{2}$$

*Total Stiffness of Gear Pair*

The mesh stiffness of gear pair is defined as follows:

Function of wheel mesh cycle:

$$k_g = k_g(\theta_{m, wh})$$
$$\theta_{m, wh} = \text{mod}(\theta_{wh}, \theta_{s, wh}) \quad \theta_{s, wh} = \frac{2\pi}{n_{wh}}$$

Function of wheel full revolution:

$$k_g = k_g(\theta_{f, wh})$$
$$\theta_{f, wh} = \text{mod}(\theta_{wh}, 2\pi)$$

Function of pinion mesh cycle:

$$k_g = k_g(\theta_{m, pn})$$
$$\theta_{m, pn} = \text{mod}(\theta_{pn}, \theta_{s, pn}) \quad \theta_{s, pn} = \frac{2\pi}{n_{pn}}$$

Function of pinion full revolution:

$$k_g = k_g(\theta_{f, pn})$$
$$\theta_{f, pn} = \text{mod}(\theta_{pn}, 2\pi)$$

#### MESH DAMPING

The mesh damping of a gear pair can be a constant value or it can be written as a function of mesh cycle or mesh stiffness.

#### *Theory for Transmission Error*

---

The **transmission Error** node can optionally be added on gear pairs. In this node, the static transmission error can be specified and added in the rotation constraint of the gear pair. By default, there is no transmission on the gear pair and it is assumed to be an ideal pair.

The static transmission error could be due to geometrical errors and geometrical modifications, such as gear runouts, misalignments, tooth tip, and root relief.

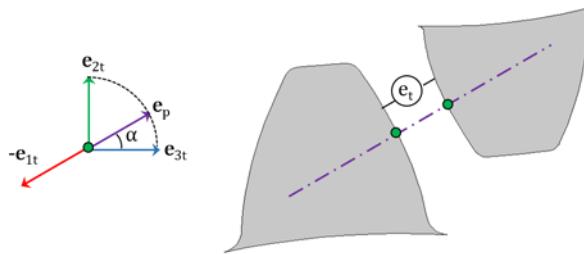


Figure 3-23: Sketch of a pair of teeth, showing the point of contact and the direction in which the transmission error is interpreted.

The transmission error expression in this section is derived for a Gear Pair node.



To get equations specific to the Worm and Wheel node, replace “wheel” with “worm” and “pinion” with “wheel” everywhere.

To get equations specific to the Rack and Pinion node, replace “wheel” with “rack” everywhere.

The static transmission error can be defined as:

$$\theta_{et} = \frac{e_t}{r_{pn}} \left\| (\mathbf{e}_{pa} \cdot \mathbf{e}_{3, pn}) \right\| \quad (3-2)$$

$$e_t = e_{wh}(\theta_{m, wh}) + e_{pn}(\theta_{m, pn})$$

where  $e_{wh}$  and  $e_{pn}$  are the wheel and pinion transmission errors, respectively.

In [Equation 3-2](#),  $e_{wh}$  and  $e_{pn}$  are written as a function of respective mesh cycle. However, it is also possible to write them individually as a function of a full revolution.

Similarly, it is also possible to directly specify  $e_t$  as a function of the mesh cycle and a full revolution of a wheel or pinion.

### Theory for Backlash

The **Backlash** node can optionally be added on gear pairs. The transmission error due to backlash is added in the rotation constraint of the gear pair. By default, there is no backlash on a gear pair.

The backlash on a gear pair plays an important role for unloaded gears. For loaded gears, it does not affect the dynamics of gears to a large extent.



The backlash is taken into account only in the case of a time-dependent analysis.

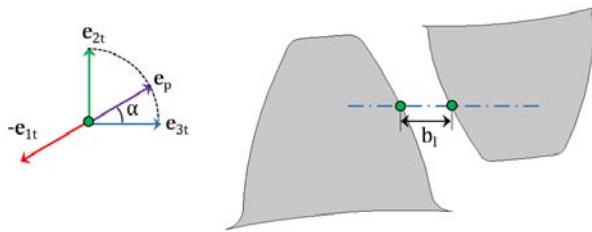


Figure 3-24: Sketch of a pair of teeth, showing the point of contact and the direction in which backlash is interpreted.

The transmission error (backlash),  $\theta_{bl}$ , is degree of freedom in this node. The backlash moment about the first axis of the tooth coordinate system and its virtual work contribution are defined as:

$$M_b = -p_{bm}((\theta_{bl} < 0)\theta_{bl} + (\theta_{bl} > \theta_b)(\theta_{bl} - \theta_b))$$

$$\delta W = M_b \cdot \delta\theta_{bl}$$

$$\theta_b = \frac{\|b_l\|}{r_{pn}}$$

where  $b_l$  and  $p_{bm}$  are the backlash and modified penalty factor, respectively.

To prevent the solver from taking larger time steps, the penalty factor is modified in the following manner:

$$p_{bm} = p_b + \text{if}(((\theta_{bl} < 0) + (\theta_{bl} > \theta_b)) \cdot (\Delta t > \Delta t_{\max}), \infty, 0)$$



The penalty factor is not modified for quasistatic analysis or when the manual control on time stepping is deselected.

The **Friction** node can optionally be added on gear pairs. In cases where friction is included on a gear pair, frictional forces in the plane perpendicular to the line of action are added on both gears.

By default, there is no friction on [Gear Pair](#) and [Rack and Pinion](#). It is, however, included for [Worm and Wheel](#). The worm and wheel pair has always neither a parallel nor an intersecting configuration, and in this configuration there is always a slip even under perfect conditions.



Friction can be applied to gear pairs only in time-dependent analysis.

---

### FRICITION MODEL

The friction force is modeled using a continuous friction law, which is capable of modeling sliding-sticking phenomena. A strict application of Coulomb's law involves discrete transition from sticking to sliding and vice versa, as dictated by a vanishing relative velocity. These discrete transitions cause numerical difficulties and to avoid them, the friction force is approximated with a continuous friction law:

$$F_f = -\mu N \frac{v}{|v|} \left( 1 - e^{-\frac{|v|}{v_0}} \right)$$

where  $F_f$  is the friction force,  $\mu$  is the frictional coefficient,  $N$  is the normal force,  $v$  is the slip velocity, and  $v_0$  is the characteristic slip velocity. The term

$$\left( 1 - e^{-\frac{|v|}{v_0}} \right)$$

is called the regularization factor.

The regularization factor smooths the friction force discontinuity. The characteristic slip velocity should be made small in comparison to the characteristic relative velocities encountered during the simulation. The continuous friction law describes both sliding and sticking behavior; that is, it completely replaces Coulomb's law. Sticking is replaced by creeping between the contacting bodies with a small relative velocity.

### NORMAL FORCE

The normal force at the contact point can be taken as the computed contact force. Alternatively, it can also be entered explicitly.

### SLIP AT CONTACT POINT

The slip vector at the contact point of a gear pair is defined as:

$$\begin{aligned}\mathbf{u}_s &= (\mathbf{x}_{cp, pn} - \mathbf{x}_{cp, wh}) + (r_{pn} \theta_{pn} \mathbf{e}'_{3, pn} - r_{wh} \theta_{wh} \mathbf{e}'_{3, wh}) \\ \mathbf{e}'_{3, wh} &= \mathbf{e}_{3, wh} - (\mathbf{e}_{3, wh} \cdot \mathbf{e}_{pa}) \mathbf{e}_{pa} \\ \mathbf{e}'_{3, pn} &= \mathbf{e}_{3, pn} - (\mathbf{e}_{3, pn} \cdot \mathbf{e}_{pa}) \mathbf{e}_{pa}\end{aligned}$$

A similar expression can be written for a worm and wheel pair. However, for a rack and pinion pair, the definition is slightly different and the slip vector is defined as:

$$\mathbf{u}_s = (\mathbf{x}_{cp, pn} - \mathbf{x}_{cp, rk}) + (r_{pn} \theta_{pn} \mathbf{e}'_{3, pn} - u_{rk} \mathbf{e}'_{3, rk})$$

In 2D, the slip vector expression reduces to:

$$\mathbf{u}_s = (\mathbf{x}_{cp, pn} - \mathbf{x}_{cp, rk})$$

In the case of a line contact model, the slip at the second contact point is defined as:

$$\mathbf{u}_{s2} = (\mathbf{x}_{cp2, pn} - \mathbf{x}_{cp2, wh}) + (r_{pn} \theta_{pn} \mathbf{e}'_{3, pn} - r_{wh} \theta_{wh} \mathbf{e}'_{3, wh})$$

### FRICITION FORCE

The friction force at the contact point and its virtual work contribution are defined as:

$$\begin{aligned}\mathbf{F}_f &= -\min(\mu N + F_{f, r}, F_{f, \max}) \frac{\dot{\mathbf{u}}_s}{\|\dot{\mathbf{u}}_s\|} \left( 1 - e^{-\frac{\|\dot{\mathbf{u}}_s\|}{v_0}} \right) \\ \delta W &= \mathbf{F}_f \cdot \delta \mathbf{u}_s\end{aligned}$$

where  $F_{f, r}$  and  $F_{f, \max}$  are the additional sliding resistance and the maximum friction force, respectively.

Similarly, in the case of a line contact model, the friction force at the second contact point and its virtual work contribution is defined as:

$$\mathbf{F}_{f2} = -\min(\mu N_2 + F_{f,r}, F_{f,max}) \frac{\dot{\mathbf{u}}_{s2}}{\|\dot{\mathbf{u}}_{s2}\|} \left( 1 - e^{-\frac{\|\dot{\mathbf{u}}_{s2}\|}{v_0}} \right)$$

$$\delta W = \mathbf{F}_{f2} \cdot \delta \mathbf{u}_{s2}$$

### ENERGY DISSIPATION RATE

The energy dissipation rate due to friction at the contact point is defined as:

$$Q_f = -\mathbf{F}_f \cdot \dot{\mathbf{u}}_s$$

For a line contact model, the additional contribution in the energy dissipation rate is defined as:

$$Q_f = -\mathbf{F}_{f2} \cdot \dot{\mathbf{u}}_{s2}$$

# Rigid Body Contact Theory

In this section:

- [Theory of Rigid Body Contact](#)
- [Spherical to Spherical Formulation](#)
- [Spherical to Arbitrary Formulation](#)
- [Theory of Friction in Rigid Body Contact](#)

## *Theory of Rigid Body Contact*

---

The [Rigid Body Contact](#) is used to model structural contact between rigid bodies. It is used to model contact between a spherical (3D) or circular (2D) rigid body, and a destination selection defined on spherical (3D) or circular (2D) rigid bodies, or on any arbitrary shaped rigid or elastic bodies.

A penalty method is used to calculate the contact force between the source and destination. The penalty factor defines the stiffness of a spring inserted between the source and destination boundaries.

Depending on the shape of the destination, two types of formulations are available in **Rigid Body Contact**. They are:

- [Spherical to Spherical Formulation](#).
- [Spherical to Arbitrary Formulation](#).

The formulation is the same in 2D, except that all spherical shaped bodies are replaced by circular shaped bodies.

## *Spherical to Spherical Formulation*

---

This formulation is used to model contact between two spherical rigid bodies. When both source and destination are rigid spheres, a center-based contact method is used. The gap between source and destination boundaries is computed. If the gap is less than zero, a penalty force is applied to prevent penetration.

The gap distance is computed from the spatial positions of the source and destination centers, and the radii of the source and destination spheres. The gap distance computation depends on the location of the source with respect to the destination. If

the source is located outside the destination, the outer boundaries of the destination may come in contact with the source. If the source is located inside the destination, the inside boundaries of the destination may come in contact with the source.

When the source is outside the destination, the *gap* is defined as

$$gap = d - r_s - r_d$$

Here,  $d$  is the distance between source and destination centers,  $r_s$  is the radius of the source sphere, and  $r_d$  is the radius of the destination sphere.

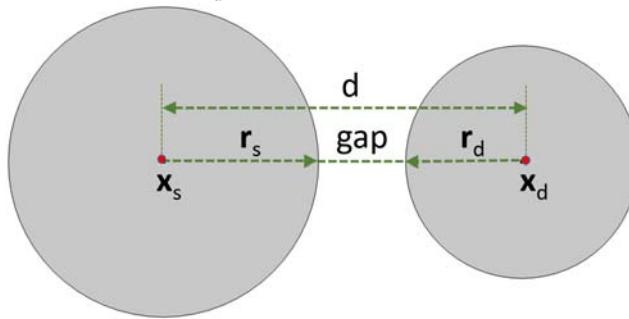


Figure 3-25: Rigid body contact between a spherical source and destination. The source is located outside the destination.

If the source is inside the destination sphere, the *gap* is defined as

$$gap = r_d - r_s - d$$

The center distance between the source and destination spheres ( $d$ ) is calculated as

$$d = |(\mathbf{X}_{src} + \mathbf{u}_{src}) - (\mathbf{X}_{dst} + \mathbf{u}_{dst})|$$

Here  $\mathbf{X}_{src}$  and  $\mathbf{X}_{dst}$  are the undeformed locations of the source and destination centers, and  $\mathbf{u}_{src}$  and  $\mathbf{u}_{dst}$  are the corresponding displacements.

A direction vector from source center to destination center ( $\mathbf{e}_c$ ) is defined as

$$\mathbf{e}_c = \frac{(\mathbf{X}_{dst} + \mathbf{u}_{dst}) - (\mathbf{X}_{src} + \mathbf{u}_{src})}{d}$$

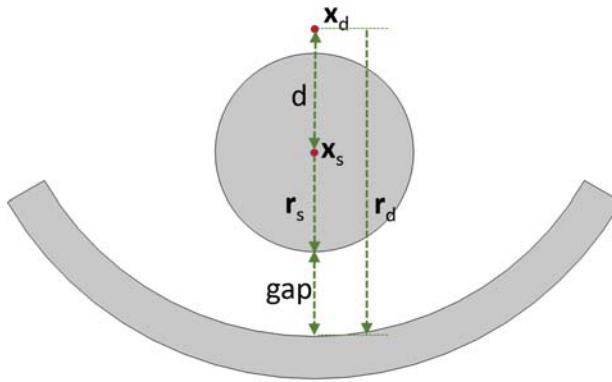


Figure 3-26: Rigid body contact between a spherical source and destination, showing gap distance. Here source is located inside the destination.

#### *The Penalty Method*

This method is based on penalization of the physical gap such that an approximate solution for non-penetration condition is obtained. The contact force ( $F_n$ ) is calculated as

$$F_n = if(gap < 0, -p_n \cdot gap, 0)$$

where  $p_n$  is the penalty factor. If the penalty factor control is automatic, the value of  $p_n$  is calculated from the penalty factor control multiplier  $f_p$  as

$$p_n = 10^{10} f_p \cdot diag$$

In this expression,  $diag$  is the geometric diagonal of the bounding box of the geometry.

#### *The Penalty, Dynamic Method*

This method is based on a viscous formulation for transient simulation. Here, the contact force is calculated based on penalization of the rate of the physical gap. The contact force ( $F_n$ ) is calculated as

$$F_n = if(gap < 0, -p_n \cdot gap, 0) + if(gap < 0, -p_{nv} \cdot \min\left(\frac{\partial}{\partial t} gap, 0\right), 0)$$

where  $\frac{\partial}{\partial t} gap$  is the time derivative of the gap distance and  $p_{nv}$  is the viscous penalty factor.

If the viscous penalty factor control is automatic, the value of  $p_{nv}$  is calculated from the characteristic time  $\tau_n$  as  $\tau_n p_n$ .

If the penalty factor control is viscous only, the stiffness terms of the penalty contact are omitted from the formulation. In this case, the contact force ( $F_n$ ) is calculated as

$$F_n = if\left(gap < 0, -p_{nv} \cdot \min\left(\frac{\partial}{\partial t}gap, 0\right), 0\right)$$

Here, if viscous penalty factor control is automatic, the value of  $p_{nv}$  is calculated from the characteristic time  $\tau_n$  as

$$p_{nv} = 10^{10} \tau_n \cdot diag$$

In this expression, *diag* is the geometric diagonal of the bounding box of the geometry.

### *Spherical to Arbitrary Formulation*

---

Using [Rigid Body Contact](#), it is also possible to model the structural contact between a spherical source and arbitrary shaped boundaries of rigid or elastic bodies. In the case of irregular shaped boundaries, contact is detected only at the point closest to the source body. In case of multiple points of destination having the same distance from the source center, only a single point is chosen for contact.

From the source center, the closest point on the outer boundary of the destination is determined using a General Extrusion operator. The gap distance is defined as the distance between the outer boundary of the source and the closest point on the destination boundary. This method uses a penalty-based formulation, where the contact force is applied if the gap between the source and destination becomes negative. The input penalty factor determines the stiffness of the spring preventing penetration.

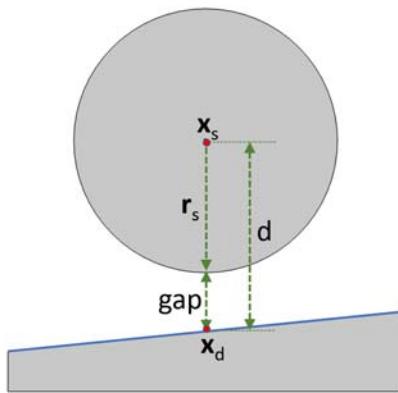


Figure 3-27: Rigid body contact between a spherical source and an arbitrary shaped destination, showing the gap distance.

The gap between source and destination is defined as

$$gap = d - r_s$$

Here,  $r_s$  is the radius of source sphere.

The distance between source center and the contact point on the destination boundaries ( $d$ ) is calculated as

$$d = |(\mathbf{X}_{src} + \mathbf{u}_{src}) - (\mathbf{x}_{dst})|$$

Here  $\mathbf{X}_{src}$  is the undeformed location of the source center,  $\mathbf{x}_{dst}$  is the spatial position vector of the closest point on the destination boundaries, and  $\mathbf{u}_{src}$  is the displacement of the source center.

The direction vector from the source center to the contact point on the destination ( $\mathbf{e}_c$ ) is defined as

$$\mathbf{e}_c = \frac{(\mathbf{x}_{dst}) - (\mathbf{X}_{src} + \mathbf{u}_{src})}{d}$$

The contact force for the penalty and the penalty dynamic methods are same as in the [Spherical to Spherical Formulation](#).

## Theory of Friction in Rigid Body Contact

---

The [Friction \(Rigid Body Contact\)](#) feature is used to add frictional forces and losses to the [Rigid Body Contact](#).

The friction force is modeled using a continuous friction law, capable of modeling sliding-sticking phenomena. A strict application of Coulomb's law involves a discrete transition from sticking to sliding, and vice versa, as dictated by a vanishing relative velocity. These discrete transitions cause numerical difficulties, and to avoid them the friction force is approximated with a continuous friction law. The magnitude of the friction force is calculated as

$$F_f = \min(\mu F_n + F_{f,r}, F_{f,max}) \left( 1 - e^{-\frac{|\mathbf{v}_{\text{slip}}|}{v_0}} \right)$$

Here,  $F_n$  is the contact force,  $F_{f,r}$  is the additional sliding resistance,  $F_{f,max}$  is the maximum friction force,  $\mathbf{v}_{\text{slip}}$  is the slip velocity vector and  $v_0$  is the characteristic slip velocity.

The characteristic slip velocity should be chosen small compared to characteristic relative velocities encountered during the simulation. The continuous friction law describes both sliding and sticking behavior, and it replaces Coulomb's law. Sticking is replaced by creeping between the contacting bodies with a small relative velocity below characteristic slip velocity.

### *Friction in the Spherical to Spherical Formulation*

In the [Spherical to Spherical Formulation](#), the slip velocity is calculated as

$$\mathbf{v}_{\text{slip}} = (\mathbf{v}_{c,s} - \mathbf{v}_{c,d}) - ((\mathbf{v}_{c,s} - \mathbf{v}_{c,d}) \cdot \mathbf{e}_c) \mathbf{e}_c$$

where  $\mathbf{v}_{c,s}$  and  $\mathbf{v}_{c,d}$  are the velocities of the source and destination spheres at the contact point. They are defined as

$$\mathbf{v}_{c,s} = \mathbf{v}_{\text{src}} + \boldsymbol{\omega}_{\text{src}} \times \mathbf{r}_s$$

$$\mathbf{v}_{c,d} = \mathbf{v}_{\text{dst}} + \boldsymbol{\omega}_{\text{dst}} \times \mathbf{r}_d$$

where  $\mathbf{v}_{\text{src}}$  and  $\mathbf{v}_{\text{dst}}$  are the velocity vectors of the centers of the source and destination spheres, and the angular velocity vectors of the source and destination spheres are given by  $\boldsymbol{\omega}_{\text{src}}$  and  $\boldsymbol{\omega}_{\text{dst}}$ . The distance vectors from the centers of the source and destination, to the contact point, are given by  $\mathbf{r}_s$  and  $\mathbf{r}_d$ . They are defined as

$$\mathbf{r}_s = r_s \mathbf{e}_c$$

$$\mathbf{r}_d = -r_d \mathbf{e}_c$$

where  $r_s$  and  $r_d$  are the radii of the source and destination spheres.

If the inside boundaries of the destination sphere are in contact with the outside boundaries of the source phase, the distance vector from the center of the source sphere to the contact point is replaced by

$$\mathbf{r}_s = -r_s \mathbf{e}_c$$

The friction force vector, applied on the source sphere is defined as

$$\mathbf{F}_f = -F_f \frac{\mathbf{v}_{\text{slip}}}{|\mathbf{v}_{\text{slip}}|}$$

where  $F_f$  is the magnitude of the friction force. The friction force vector is applied also on the destination sphere, but with opposite sign.

The frictional moment applied at the center of source and destination spheres is defined as

$$\mathbf{M}_{fs} = \mathbf{r}_s \times \mathbf{F}_f$$

$$\mathbf{M}_{fd} = -\mathbf{r}_d \times \mathbf{F}_f$$

In 3D, the quaternion moments corresponding to the physical moments for the geometrically nonlinear case are defined as

$$\mathbf{R}_{fs} = 2\{0, \mathbf{M}_{fs}\} \otimes \mathbf{q}_{\text{src}}$$

$$\mathbf{R}_{fd} = 2\{0, \mathbf{M}_{fd}\} \otimes \mathbf{q}_{\text{dst}}$$

where  $\mathbf{q}_{\text{src}}$  and  $\mathbf{q}_{\text{dst}}$  are the quaternions of the source and destination spheres.

The energy dissipation rate caused by friction can be written as

$$Q_f = -\mathbf{F}_f \cdot \mathbf{v}_{\text{slip}}$$

#### *Friction in Spherical to Arbitrary Formulation*

In the [Spherical to Arbitrary Formulation](#), the slip velocity is calculated as

$$\mathbf{v}_{\text{slip}} = (\mathbf{v}_{c,s} - \mathbf{v}_{c,d}) - ((\mathbf{v}_{c,s} - \mathbf{v}_{c,d}) \cdot \mathbf{e}_c) \mathbf{e}_c$$

where  $\mathbf{v}_{c,s}$  is the velocity of the source sphere at the contact point. It is defined as

$$\mathbf{v}_{c,s} = \mathbf{v}_{src} + \omega_{src} \times \mathbf{r}_s$$

where  $\mathbf{v}_{src}$  is the velocity vector of the center of the source and  $\omega_{src}$  is the angular velocity vector of the source sphere. The distance vector from the center to the contact point for the source is defined as

$$\mathbf{r}_s = r_s \mathbf{e}_c$$

where  $r_s$  is the radius of the source sphere. The velocity of the destination boundary, at the contact point, is  $\mathbf{v}_{c,d}$ .

The friction force vector, applied on the source sphere, is defined as

$$\mathbf{F}_f = -F_f \frac{\mathbf{v}_{slip}}{|\mathbf{v}_{slip}|}$$

where  $F_f$  is the magnitude of the friction force. The friction force vector applied on the destination sphere is equal, but opposite in direction.

The frictional moment applied at the center of the source is defined as

$$\mathbf{M}_{fs} = \mathbf{r}_s \times \mathbf{F}_f$$

In 3D, the quaternion moment corresponding to the physical moment for the geometrically nonlinear case is defined as

$$\mathbf{R}_{fs} = 2\{0, \mathbf{M}_{fs}\} \otimes \mathbf{q}_{src}$$

where  $\mathbf{q}_{src}$  is the quaternion of source sphere.

The energy dissipation rate caused by friction can be written as

$$Q_f = -\mathbf{F}_f \cdot \mathbf{v}_{slip}$$

# 4

## Multibody Dynamics

This chapter describes the Multibody Dynamics interface, which is found under the **Structural Mechanics** branch (

In this chapter:

- [The Multibody Dynamics Interface](#)

# The Multibody Dynamics Interface

The **Multibody Dynamics (mbd)** interface () found under the **Structural Mechanics** branch () when adding a physics interface, is intended for analysis of mechanical assemblies. The parts in the assembly can be rigid or flexible, and are connected by various types of joints, gears, chains, springs, or dampers. Flexible parts can be defined using solid, shell, or beam elements. There are many types of joints, such as hinges or ball joints, which can be used based on the type of connection required between components. There are many types of gears, such as spur, helical, bevel, worm, or rack, which can be used for power transmission. Computed results are displacements, velocities, accelerations, joint forces, gear contact forces, and — in flexible parts — stresses. The joints can be given properties such as spring constants, damping, friction, and limits on movement. The gears pairs can also be given properties such as gear mesh stiffness, mesh damping, backlash, transmission error, and friction.

The Linear Elastic Material is the default material model. It adds the equations for the displacements in a linear elastic solid and has a Settings window to define the elastic and inertia properties of a material. As an alternative, a domain can be made into a rigid body, using the Rigid Domain material model. A rigid domain adds the rigid body dynamics equations and has a Settings window to define inertia properties of a material.

With the Nonlinear Structural Materials Module, you can also incorporate nonlinear material models like hyperelasticity or plasticity by adding a Solid Mechanics interface to the model.

When the **Multibody Dynamics** interface is added, these default nodes are also added to the **Model Builder** — **Linear Elastic Material, Free** (a boundary condition where boundaries are free, with no loads or constraints), and **Initial Values** (only applicable for flexible domains). Then, from the **Physics** toolbar, add features that implement other multibody dynamics properties. You can also right-click **Multibody Dynamics** to select physics features from the context menu.

## SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to

different physics interfaces, the **Name** string must be unique. Only letters, numbers, and underscores (\_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is **mbd**.

## 2D APPROXIMATION

From the **2D approximation** list select **Plane stress** or **Plane strain** (the default). Plane stress is relevant for structures which are thin in the out-of-plane direction, such as a thin plate. Plane strain is relevant when the 2D model can be considered as a cut through an object which is long in the out-of-plane direction. This information is needed if the components are flexible and modeled using **Linear Elastic Material**. For more information see the theory section.

 When modeling using plane stress, the Solid Mechanics interface solves for the out-of-plane strain displacement derivative,  $\frac{\partial w}{\partial Z}$ , in addition to the displacement field **u**.

When combining Multibody Dynamics with other types of physics, there is often an assumption that the out-of-plane extension is infinitely long. This is the case in for example Acoustic-Structure interaction problems. In these cases, **Plane strain** is usually the correct choice.

## THICKNESS

 For 2D components, enter a value or expression for the **Thickness** *d*. The default value of 1 m is suitable for plane strain models, where it represents a unit-depth slice, for example. For plane stress models, enter the actual thickness, which should be small compared to the size of the plate for the plane stress assumption to be valid. In Acoustic-Structure Interaction problems, the **Thickness** should be set to 1 m.

Use a [Change Thickness](#) node to change thickness in parts of the geometry if necessary.

## STRUCTURAL TRANSIENT BEHAVIOR

From the **Structural transient behavior** list, select **Include inertial terms** or **Quasi static**. Use **Quasi static** to treat the mechanical behavior as quasi static (with no mass effects

— that is, no second-order time derivatives). Selecting this option gives a more efficient solution for problems where the variation in time is slow when compared to the natural frequencies of the system.

#### REFERENCE POINT FOR MOMENT COMPUTATION

Enter the coordinates for the **Reference point for moment computation**  $\mathbf{x}_{\text{ref}}$  (variable `refpnt`). The resulting moments (applied or as reactions) are then computed relative to this reference point. During the results and analysis stage, the coordinates can be changed in the **Parameters** section in the result nodes.

#### INITIAL VALUES

Specify initial values for a rigid body translation and rotation. The values specified here are, as a default, inherited by the initial values settings for **Rigid Domain** and **Linear Elastic Material**.

- **Center of rotation**  $\mathbf{X}_c$
- **Displacement at center of rotation**  $\mathbf{u}$
- **Velocity at center of rotation**  $\partial\mathbf{u}/\partial t$
- *For 3D components only:* **Axis of rotation**  $\Omega$ .
- **Angle of rotation**  $\phi$
- **Angular velocity**  $\omega$  (3D components) and  $\partial\phi/\partial t$  (2D components)

#### AUTOMATED MODEL SETUP

Use this section to automatically generate physics nodes in the Multibody Dynamics interface. This functionality makes the model setup easier and more robust for systems with many rigid bodies or gears.

By using the **Physics Node Generation** button (  ), you can automatically create **Rigid Domain**, **Gear**, and **Joint** nodes.

The **Physics Node Generation** has the following options:

- **Create Rigid Domains**
- **Create Gears**
- **Create Joints**

The **Create Rigid Domains** option (  ) automatically creates **Rigid Domain** nodes for geometrically disconnected objects. The **Rigid Domain Selections** input controls for which geometric objects to create a rigid domain. It is by default set to **From physics interface**, which creates rigid domain for all objects in the selection of the physics

interface. All available domain selections are also listed in the **Rigid Domain Selections** list, which makes it possible to create rigid domains for a subset of objects. Select the **Include mass and moment of inertia node** check box to add a **Mass and Moment of Inertia** subnode to each auto-generated **Rigid Domain** node. The **Density** of the created rigid domains is automatically set to zero when this check box is selected.

The **Create Gears** option (  ) automatically creates a **Gear** node for every gear geometry part present in the geometry sequence. This option requires that the gear geometry is built from the COMSOL **Part Library**. The domain selection for the auto-generated **Gear** node is same as the gear geometry part. Relevant definitions and parameters are retrieved from the gear geometry part, and entered in the settings for the corresponding gear node. The **Rigid Domain Selections** and **Include mass and moment of inertia** does not affect the **Create Gears** option.

The **Create Joints** option (  ) automatically creates **Joint** nodes from the geometry based on the shape of the boundaries. This option requires **Identity Boundary Pair** nodes created under **Definitions**. If the **Source Boundaries** and **Destination Boundaries** are of same standard shape, this option creates one **Joint** node for each active **Identity Boundary Pair** present. The possible shapes of boundaries for automatic joint creation are listed under **Joint types**.



The [Creating Joints Automatically](#) section describes the details of creating various joint types from the geometry.

## RESULTS

In this section you can modify the postprocessing defaults for the interface.

You can select a **Body defining reference frame** from the list of all rigid domains, attachments, and gears used in the model. It is used to visualize the relative motion of objects with respect to the selected body.

The variables **u\_ref**, **v\_ref**, and **w\_ref**, being the three components of the displacement field with respect to the reference body, are available in the result menus. Also, the variables **mbd.disp\_ref** and **mbd.vel\_ref** contain the total displacement and velocity with respect to the reference body, respectively.

When you change the reference body, you need to update the solution before you can access the new definitions of these variables. To do that, right-click the **Study** node and select **Update Solution** (  ) (or press F5).

## JOINTS SUMMARY

This is an information table that displays the name of all the joints used in a model and their source and destination attachments. This table summarizes all the connections present in a model and can be used to review the connections.

## RIGID BODY DOF SUMMARY

This is an information table that displays the number of degrees of freedom and constraints introduced by rigid domains and joints. You can use this information to determine whether your model has an appropriate number of constraints or not when running pure rigid body analysis.

Rigid Body DOF Summary				
"	N	DOF	Prescribed	Constraints
Rigid bodies	14	98	0	20
Prismatic joints	3	3	3	18
Hinge joints	8	8	0	48
Cylindrical joints	4	8	2	24
Ball joints	2	8	0	14
Total	-	125	5	124
Net	-	0	-	4

The contents of the columns are:

- **N:** The number of features of a certain type.
- **DOF:** The number of degrees of freedom added to the model by all features of this type.
- **Prescribed:** The number of degrees of freedom controlled by **Prescribed Motion** and similar conditions for all features of this type.
- **Constraints:** The number of constraints for all features of this type. Both internal constraints and constraints added explicitly by for example **Fixed** conditions are included.

The last two rows of the table contain a summary. In the **Total** row, the number of DOFs, prescribed conditions, and constraints are summed.

The **Net** row contains the net number of degrees of freedom of the model — that is, the difference between all degrees of freedom and the constraints and prescribed motions. A negative net number of degrees of freedom indicates that the mechanism is overconstrained and is not shown. In that case, the net number of constraints is instead displayed.

Some constraints are conditional and are not present during the whole analysis. Such constraints are counted in the summary table, irrespective of their actual status.

The contents of the table do not change if you suppress any of the contributing features from the settings in the study step.

In general, and if there are flexible bodies in the system, the model's total number of degrees of freedom increase significantly. Such degrees of freedom are not accounted for in the summary.

### DEPENDENT VARIABLES

The physics interface uses the global spatial components of the **Displacement field**  $u$  as dependent variables in the flexible domains. The default names for the components are  $(u, v, w)$  in 3D. In 2D the component names are  $(u, v)$ , and in 2D axial symmetry they are  $(u, w)$ . You can, however, not use the “missing” component name in the 2D cases as a parameter or variable name because it is still used internally.

You can change both the field name and the individual component names. If a new field name coincides with the name of another displacement field, the two fields (and the interfaces which define them) share degrees of freedom and dependent variable component names. You can use this behavior to connect a Multibody Dynamics interface to a Shell directly attached to the boundaries of the solid domain, or to a Solid Mechanics interface sharing a common boundary.

A new field name must not coincide with the name of a field of another type (that is, it must contain a displacement field), or with a component name belonging to some other field. Component names must be unique within a model except when two interfaces share a common field name.

Each rigid domain, attachment, and joint also adds a number of global dependent variables depending on the number of ODEs needed to represent its motion.

### DISCRETIZATION

In the Multibody Dynamics interface you can choose not only the order of the discretization, but also the type of shape functions: *Lagrange* or *serendipity*. For highly distorted elements, Lagrange shape functions provide better accuracy than serendipity shape functions of the same order. The serendipity shape functions, however, give significant reductions of the model size for a given mesh containing hexahedral, prism, or quadrilateral elements.

The discretization order applies to the flexible bodies. The default is to use **Linear** shape functions for the **Displacement field**. If you want to compute stresses with good accuracy, increase the shape function order to **Quadratic serendipity** or **Quadratic**.



- Domain, Boundary, Edge, Point, and Pair Nodes for the Multibody Dynamics Interface
- Multibody Dynamics Theory

### *Domain, Boundary, Edge, Point, and Pair Nodes for the Multibody Dynamics Interface*

The **Multibody Dynamics Interface** has these domain, boundary, edge, point, and pair nodes (and subnodes), listed in alphabetical order, available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or by right-clicking then main physics interface node to access the context menu (all users).



In general, to add a node, go to the **Physics** toolbar, no matter what operating system you are using. Subnodes are available by clicking the parent node and selecting it from the **Attributes** menu.



In the *COMSOL Multiphysics Reference Manual* see [Table 2-4](#) for links to common sections and [Table 2-5](#) to common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.



The links to the nodes described in external guides, such as the *Structural Mechanics Module User's Guide*, do not work in the PDF, only from the online help in COMSOL Multiphysics.

### **FEATURES AVAILABLE FROM SUBMENUS**

Many features for the Multibody Dynamics interface are added from submenus in the Physics toolbar groups or context menu (when you right-click the node). The submenu name is the same in both cases.

The submenus at the **Domain** level are **Gears**; **Volume Forces**; **Mass, Spring, and Damper**; **Domain Constraints**; and **More**.

There are also the **Edges** and **Points** submenus.

The submenus at the **Global** level are **Joints**, **Gear Pairs**, and **Variables**.



Right-click the **Multibody Dynamics** node, select from the **Joints** menu (if **No Sorting** is selected from the context menu) or the **Global>Joints** menu (if **Sorting by Space Dimension** is selected).

The [Attachment](#) node is available from the **Physics** toolbar, **Boundaries** menu.

The following nodes are available from the **Physics** toolbar, **Domains** menu.

- [Bevel Gear](#)
- [Helical Gear](#)
- [Helical Rack](#)
- [Initially Rigid](#)
- [Initial Values](#)
- [Rigid Domain](#)
- [Spur Gear](#)
- [Spur Rack](#)
- [Worm Gear](#)

The following nodes are available from the **Physics** toolbar, **Global** menu.

- Ball Joint
- Base Motion
- Cam-Follower
- Chain Drive
- Clearance Joint
- Cylindrical Joint
- Distance Joint
- Fixed Joint
- Gear Pair
- Hinge Joint
- Planar Joint
- Prismatic Joint
- Rack and Pinion
- Reduced Slot Joint
- Rigid Body Contact
- Screw Joint
- Slot Joint
- Universal Joint
- Worm and Wheel

These subnodes are available from the context menu (right-click the joint parent node) or from the **Physics** toolbar, **Attributes** menu:

- Applied Force and Moment
- Constraints
- Friction
- Locking
- Prescribed Motion
- Spring and Damper

These subnodes are automatically added to the joint parent node, or to one of its subnodes, when certain options are selected in the Settings window:

- Center of Joint Nodes
- Contact Area
- Destination Filter
- Destination Point Nodes
- Input Shaft Orientation
- Joint Axis
- Joint Elasticity
- Output Shaft Orientation
- Source Filter
- Source Point Nodes
- Sprocket Axis

These subnodes are automatically added to the gear pair parent node when an appropriate option is selected in the Settings window:

- [Backlash](#)
- [Friction \(Gear\)](#)
- [Gear Elasticity](#)
- [Transmission Error](#)

These subnodes are automatically added to the gear or rack parent node when an appropriate option is selected in the Settings window:

- [Gear Axis](#)
- [Rack Axis](#)

These subnodes are added to the rigid body contact parent node when an appropriate option is selected in the Settings window:

- [Destination Point Nodes](#)
- [Friction \(Rigid Body Contact\)](#)
- [Source Point Nodes](#)

The following nodes and subnodes are described for the Solid Mechanics interface in the *Structural Mechanics Module User's Guide* (listed in alphabetical order):

- [Added Mass](#)
- [Applied Force \(Rigid Connector\)](#)
- [Applied Force \(Rigid Domain\)](#)
- [Applied Moment \(Rigid Connector\)](#)
- [Applied Moment \(Rigid Domain\)](#)
- [Average Rotation](#)
- [Body Load](#)
- [Boundary Load](#)
- [Change Thickness \(2D\)](#)
- [Center of Mass Nodes](#)
- [Center of Rotation Nodes](#)
- [Contact](#)
- [Damping](#)
- [Destination Point \(for Spring-Damper\)](#)
- [Edge Load](#)
- [Fixed Constraint](#)
- [Fixed Constraint \(Rigid Domain\)](#)
- [Flexible Foundation](#)
- [Free](#)
- [Gravity](#)
- [Initial Stress and Strain](#)
- [Linear Elastic Material](#)
- [Location Nodes](#)
- [Mass and Moment of Inertia \(Rigid Connector\)](#)
- [Mass and Moment of Inertia \(Rigid Domain\)](#)
- [Moving Foundation](#)
- [Phase](#)
- [Point Load](#)
- [Predeformation](#)
- [Prescribed Displacement](#)
- [Prescribed Displacement/Rotation](#)
- [Radial Roller Bearing](#)
- [Rigid Connector](#)
- [Roller](#)
- [Rotating Frame](#)
- [Slip Velocity](#)
- [Source Point \(for Spring-Damper\)](#)
- [Spring Foundation](#)
- [Spring-Damper](#)
- [Thermal Expansion \(for Materials\)](#)
- [Wear](#)

### *Initial Values*

---

The **Initial Values** node adds initial values for the displacement field and velocity field in a flexible component. The data can serve as an initial condition for a transient simulation or as an initial guess for a nonlinear analysis. The initial values either can be directly prescribed in this node or can be inherited from the physics interface level. If needed, you can add more **Initial Values** nodes.

## DOMAIN SELECTION

The domains, in which a **Rigid Domain** material model is defined, are automatically not applicable.

## INITIAL VALUES

Select **From physics interface node** or **Locally defined**.

When using **From physics interface node**, the initial values for displacement field and velocity field are computed using the expressions entered for the rigid-body displacement, rotation, and velocities in the [Initial Values](#) section of the **Multibody Dynamics** node.

For **Locally defined**, enter values or expressions for the initial values of the **Displacement field  $\mathbf{u}$**  (the displacement components  $u$ ,  $v$ , and  $w$  in 3D) and the **Structural velocity field  $\partial\mathbf{u}/\partial t$** .

Select a **Consistent initialization — Default** or **Force initial values**. The default when the given initial values are inconsistent is to make them adapt to each other by adjusting all initial values. To enforce that the exact values given in this node should be maintained through the consistent initialization process, use **Force initial values**. When **Force initial values** is selected, click to select the applicable check box or boxes — **Translation along first axis**, **Translation along second axis**, and **Translation along third axis**. Only the selected degrees of freedom have initial values fixed during the consistent initialization process.

---



- For details about the settings at the physics interface level, see [Initial Values](#).
- For details about the initialization process, see [Initialization in Time-Dependent Studies](#).

---

## LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics>More>Initial Values**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Domains>More>Initial Values**

## *Initially Rigid*

---

Add the **Initially Rigid** node to enforce that a flexible domain is treated as a rigid body during the initialization of a time-dependent study.

This node has a domain selection and it can be only defined in domains where the **Linear Elastic Material** model is used. It overrides the **Initial Values** node and instead defines the initial values of the selected elastic domains itself.

### **INITIAL VALUES**

Select **From physics interface node** or **Locally defined**.

- When **From physics interface node** is selected, initial values for rigid body displacement, rotation, and velocities are inherited from the physics interface level.
- When **Locally defined** is selected, an **Initial Values** subnode is automatically added, in which you can initialize the rigid domain degrees of freedom.

Select a **Consistent initialization** — **Default** or **Force initial values** to enforce that the exact values given in this node should be maintained through the consistent initialization process. The default when the given initial values are inconsistent is to make them adapt to each other by adjusting all initial values. For **Force initial values** click to select the applicable check boxes — **Translation along first axis**, **Translation along second axis**, **Translation along third axis**, and **Total rotation**. Only the selected degrees of freedom have the initial values fixed during the consistent initialization process.

---

- The optional subnode is the same as for a **Rigid Domain**, and is described in [Initial Values \(Rigid Domain\)](#).
- For details about the input expressions at the physics interface level, see [Initial Values](#).
- See [Rigid Domain Model](#) in the *Structural Mechanics Module User's Guide* for more information.
- For details about the initialization process, see [Initialization in Time-Dependent Studies](#).

---

### **LOCATION IN USER INTERFACE**

*Context Menus*

**Multibody Dynamics**>**More**>**Initially Rigid**

## *Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Domains>More>Initially Rigid**

## *Attachment*

---

The **Attachment** node is used to define a set of boundaries on a flexible or rigid component, which can be used to connect it with other components through a joint in the Multibody Dynamics interface, or a [Spring-Damper](#) in any structural mechanics interface.

There are two formulations of the attachment. In the default formulation, all the selected boundaries behave as if they were connected by a common rigid body. In some cases, this give an unwanted stiffening or unrealistic local stresses. You can then switch to a flexible formulation where the constraint is applied only in an average sense.

Using attachments is optional when connecting a rigid component. It is, however, useful if you consider switching between a rigid and a flexible representation, since you then do not have to change the settings in the joints.

Attachments can be added to boundaries in a Multibody Dynamics or a Solid Mechanics interface, to edges in a Shell interface, and to points in a Beam interface. This makes it possible to use a joint for connecting to parts modeled in different physics interfaces.

### **CONNECTION TYPE**

Select **Rigid** or **Flexible**.

When the connection type is rigid, the whole attachment selection acts as a virtual rigid object. This will cause a stiffening of the part with the attachment, and local stress concentrations at the selected boundaries if the domain with the attachment is flexible.

In the flexible formulation, the selected boundaries are allowed to have internal deformations, and the kinematic constraints are fulfilled only in an average sense. This will essentially provide a statically equivalent load distribution over the boundaries.

If a selected boundary is located on a rigid domain, the connection type setting does not matter. The rigid formulation is always used

---

With the flexible formulation, some extra degrees of freedom are added for each attachment. In order to get good convergence properties for a nonlinear solution, these variables must have a good scaling.

When a new solver sequence is generated, then an appropriate manual scaling is automatically set for these variables.



If, however, an existing solver sequence was generated while **Connection Type** was set to **Rigid**, and you then change to **Flexible**, no such scaling will be present. In this case, you either have to regenerate the solver sequence, or set the scaling manually under the **Dependent Variables** node in each study step.

---

- [Attachments](#)



- [The Solid Mechanics Interface](#), [The Shell and Plate Interfaces](#), and [The Beam Interface](#) in the *Structural Mechanics Module User's Guide*.

---

## LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics>Attachment**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Boundaries>Multibody Dynamics>Attachment**

## Rigid Domain

---

Add the **Rigid Domain** node and select one or more domains to make them a rigid body.

These subnodes are described in the *Structural Mechanics Module User's Guide*:

- [Fixed Constraint \(Rigid Domain\)](#)
- [Prescribed Displacement/Rotation](#)
- [Applied Force \(Rigid Domain\)](#)
- [Applied Moment \(Rigid Domain\)](#)
- [Mass and Moment of Inertia \(Rigid Domain\)](#)
- [Spring Foundation \(Rigid Domain\)](#)

### RIGID DOMAIN

The default **Density**  $\rho$  is taken **From material**. In this case, the material assignment for the domain supplies the mass density. For **User defined**, enter another value or expression.

 The density is needed for dynamic analysis. It is also used when computing mass forces for gravitational or rotating frame loads, and when computing mass properties ([Computing Mass Properties](#)).

### CENTER OF ROTATION

Select a **Center of Rotation** — **Center of mass**, **Centroid of selected entities**, or **User defined**. The center of rotation affects how displacements are interpreted and is also used as the default in various subnodes.

- For **Center of mass**, the center of rotation is taken as the center of mass of the rigid domain.
- For **Centroid of selected entities** select an **Entity level** — **Boundary**, **Edge**, or **Point**. The available choices depend on geometrical dimension. The center of rotation is located at the centroid of the selected entities, which do not need to be related to

rigid domain itself. As a special case, you can select a single point and thus use that point as center of rotation.



When chosen, a default [Center of Rotation: Boundary](#), [Center of Rotation: Edge](#), or [Center of Rotation: Point](#) subnode is automatically added.

- For **User defined**, enter the **Global coordinates of center of rotation**,  $\mathbf{X}_c$ , in the table.

Select the **Offset** check box to add an optional offset vector to the definition of the center of rotation. Enter values for the offset vector  $\mathbf{X}_{\text{offset}}$ .

The center of rotation used is the sum of the vector obtained from any of the input methods and the offset vector.

$$\mathbf{X}_P = \mathbf{X}_{P,\text{input}} + \mathbf{X}_{\text{offset}}$$

#### INITIAL VALUES

Select **From physics interface node** or **Locally defined**.

- When **From physics interface node** is selected, initial values for rigid body displacement, rotation, and velocities are inherited from the physics interface level.
- When **Locally defined** is selected, an **Initial Values** subnode is automatically added, in which you can initialize the rigid domain degrees of freedom.

Select a **Consistent initialization — Default** or **Force initial values** to enforce that the exact values given in this node should be maintained through the consistent initialization process. The default when the given initial values are inconsistent is to make them adapt to each other by adjusting all initial values. For **Force initial values**, click to select the applicable check boxes — **Translation along first axis**, **Translation along second axis**,

**Translation along third axis, and Total rotation.** Only the selected degrees of freedom have the initial values fixed during the consistent initialization process.

---



- The optional subnode is described in [Initial Values \(Rigid Domain\)](#).
- For details about the input expressions at the physics interface level, see [Initial Values](#).
- See [Rigid Domain Model](#) in the *Structural Mechanics Module User's Guide* for more information.
- For details about the initialization process, see [Initialization in Time-Dependent Studies](#).

---

#### LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics>Rigid Domain**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Domains>Multibody Dynamics>Rigid Domain**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Domains>Multibody Dynamics>Initial Values**

#### *Center of Rotation Nodes*

---

For each of the following physics features, a subnode is automatically added as indicated.

- [Rigid Domain](#)
- [Initial Values \(Rigid Domain\)](#)
- Gears, as described under [Spur Gear](#)

#### **CENTER OF ROTATION: BOUNDARY**

Use the **Center of Rotation: Boundary** subnode to select a set of boundaries whose centroid represents the center of rotation.

### CENTER OF ROTATION: EDGE

Use the **Center of Rotation: Edge** subnode to select a set of edges whose centroid represents the center of rotation.

### CENTER OF ROTATION: POINT

Use the **Center of Rotation: Point** subnode to select a set of points whose centroid represents the center of rotation.

### LOCATION IN USER INTERFACE

These nodes are automatically added when **Centroid of selected entities** is selected in the parent node. They cannot be added or removed manually.

## *Initial Values (Rigid Domain)*

---

The **Initial Values** subnode is automatically added when you select that the initial values in a **Rigid Domain** node are **Locally defined**.

You can enter initial values for the rigid body displacement, rotation and velocities, which can serve as an initial condition for a transient simulation or as an initial guess for a nonlinear analysis. The initial values that you specify are interpreted in the selected coordinate system.

### CENTER OF ROTATION

Select an option from the list: **From parent**, **Centroid of selected entities**, or **User defined**. The given initial values are interpreted at the center of rotation.

- For **From parent**, the center of rotation is taken as the one defined in the parent **Rigid Domain** node.
- For **Centroid of selected entities** select an **Entity level** — **Boundary**, **Edge**, or **Point**. The available choices depend on geometrical dimension. The initial conditions act at the centroid of the selected entities, which do not need to be related to rigid domain itself. As a special case, you can select a single point and thus prescribe the initial conditions at that point.



When chosen, a default **Center of Rotation: Boundary**, **Center of Rotation: Point**, or **Center of Rotation: Edge** subnode is automatically added.

- For **User defined**, enter the **Global coordinates of center of rotation**,  $\mathbf{X}_c$ , in the table.

Select the **Offset** check box to add an optional offset vector to the definition of the center of rotation. Enter values for the offset vector  $\mathbf{X}_{\text{offset}}$ .

The center of rotation used is the sum of the vector obtained from any of the input methods and the offset vector.

$$\mathbf{X}_P = \mathbf{X}_{P,\text{input}} + \mathbf{X}_{\text{offset}}$$

#### INITIAL VALUES: TRANSLATIONAL

- **Displacement at center of rotation  $\mathbf{u}$ .**
- **Velocity at center of rotation  $\partial\mathbf{u}/\partial t$ .**

#### INITIAL VALUES: ROTATIONAL

- *For 3D components only:* **Axis of rotation  $\Omega$**
- **Angle of rotation  $\phi$ .**
- **Angular velocity  $\omega$**  (3D components) and  $\partial\phi/\partial t$  (2D components)

#### LOCATION IN USER INTERFACE

This node is automatically added when **Locally defined** is selected in the **Initial Values** section in the parent **Rigid Domain** node. It cannot be added or removed manually.

## *Prismatic Joint*

---

Use the **Prismatic Joint** node for 2D and 3D components to create a joint between two components such that they are free to translate relative to each other along the joint axis.

The [Constraints](#), [Locking](#), [Spring and Damper](#), [Prescribed Motion](#), [Applied Force and Moment](#), and [Friction](#) subnodes can be added to modify the properties of the joint.

#### ATTACHMENT SELECTION

Select a **Source** and a **Destination** attachment from the lists. You can select any [Attachment](#), [Rigid Domain](#), [Spur Gear](#), [Helical Gear](#), [Bevel Gear](#), [Worm Gear](#), [Spur Rack](#), or [Helical Rack](#) that is defined for the model. In **Source** an additional option, **Fixed**, is also available. When **Fixed** is selected, the source side of the joint is fixed in space, and not connected to any modeled parts. If any [Base Motion](#) nodes are present in the model, these also appear in the list for the **Source** attachment. Selecting a base motion gives a behavior similar to using **Fixed**, with the exception that the base now can have a nonzero prescribed displacement, velocity, or acceleration.

In large models, the list of available attachments can become long. You can then create geometric filters to narrow down the search. When the **Use selection filter** check box is selected, two subnodes named **Source Filter** and **Destination Filter** are added to the joint node. In these subnodes you can make graphic selections of the objects whose attachments should be shown in the **Source** and a **Destination** lists.

### CENTER OF JOINT

The center of joint is the point about which displacements and rotations of source and destination boundaries are interpreted. Select an option from the list — **Centroid of source**, **Centroid of destination**, **Centroid of selected entities**, or **User defined**.

- For **Centroid of source** and **Centroid of destination**, the joint is located at the centroid of the selection giving the source or destination, respectively.
- For **Center of mass**, the center of joint is taken as the center of mass of the gear.
- For **Centroid of selected entities** select an **Entity level** — **Boundary**, **Edge**, or **Point**. The available choices depend on geometrical dimension. The center of joint is located at the centroid of the selected entities. As a special case, you can select a single point, and thus use that point as center of joint.



When chosen, a default **Center of Joint: Boundary**, **Center of Joint: Edge**, or **Center of Joint: Point** subnode is automatically added.

- For **User defined**, enter the **Global coordinates of center of rotation**,  $\mathbf{X}_c$ , in the table.

Select the **Offset** check box to add an optional offset vector to the definition of the center of rotation. Enter values for the offset vector  $\mathbf{X}_{\text{offset}}$ .

The center of rotation used is the sum of the vector obtained from any of the input methods and the offset vector.

$$\mathbf{X}_P = \mathbf{X}_{P,\text{input}} + \mathbf{X}_{\text{offset}}$$

### AXIS OF JOINT

The axis of joint is the axis along which the relative motion is allowed. Select an option from the list — **Specify direction**, **Select a parallel edge**, or **From selected coordinate system**.

- For **Specify direction** enter a value or expression for  $\mathbf{e}_0$ . The default is  $(1, 0, 0)$ . The direction is specified in the selected coordinate system.

- For **Select a parallel edge** an edge can be selected on the **Joint Axis** subnode, which is added automatically. The vector from the first to the last end of the edge is used to define the joint axis. Any edge in the model can be used. Select the **Reverse direction** check box to reverse the direction of the joint axis.
- For **From selected coordinate system** the joint axis is formed using the selected coordinate system. With **Axis to use** you select the axis (first, second, or third) that is parallel to the joint axis.

### JOINT ELASTICITY

Select an option from the list to determine whether the joint should be fully rigid or elastic — **Rigid joint** or **Elastic joint**. For **Elastic joint**, the properties of an elastic joint are given in the **Joint Elasticity** subnode, which is added automatically.

### JOINT FORCES AND MOMENTS

Select an option from the list to use the method of computation for the forces and moments — **Computed on destination**, **Computed on source**, **Do not compute**, **Computed using weak constraints**, or **Computed using penalty method**. For **Computed using penalty method**, enter the **Penalty factor**  $p_j$ . The default is  $(mbd.prj1.Eequ*(0.1*mbd.diag))$ . The exact naming of the variable depends on the tags of the nodes in the model tree.



- [Prismatic Joint Theory](#)
- [Computing Joint Forces and Moments](#)

### LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics**>**Joints**>**Prismatic Joint**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Global**>**Joints**>**Prismatic Joint**

### *Hinge Joint*

Use the **Hinge Joint** node for 2D and 3D components to create a joint between two components such that they are free to rotate relative to each other about the joint axis at the center of joint.

The [Constraints](#), [Locking](#), [Spring and Damper](#), [Prescribed Motion](#), [Applied Force](#) and [Moment](#) subnodes can be added to modify the properties of the joint.



See [Prismatic Joint](#) for all settings — **Attachment Selection**, **Center of Joint**, **Joint Elasticity**, **Joint Forces and Moments**, and **Sketch**. For 3D components, also see **Axis of Joint**.



[Hinge Joint Theory](#)

#### LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics**>**Joints**>**Hinge Joint**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Global**>**Joints**>**Hinge Joint**

#### *Cylindrical Joint*

Use the **Cylindrical Joint** node for 3D components to create a joint between two components such that they are free to translate relative to each other along the joint axis and also free to rotate relative to each other about the same axis.

The [Constraints](#), [Locking](#), [Spring and Damper](#), [Prescribed Motion](#), [Applied Force](#) and [Moment](#) subnodes can be added to modify the properties of the joint



See [Prismatic Joint](#) for all settings — **Attachment Selection**, **Center of Joint**, **Axis of Joint**, **Joint Elasticity**, **Joint Forces and Moments**, and **Sketch**.



[Cylindrical Joint Theory](#)

## LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics>Joints>Cylindrical Joint**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Global>Joints>Cylindrical Joint**

## *Screw Joint*

---

Use the **Screw Joint** node for 3D components to create a joint between two components such that they are free to rotate relative to each other about the joint axis and at the same time translate along the same axis. The translation and rotation are not independent; they are connected through the number of starts and the pitch of a screw thread.

The [Constraints](#), [Locking](#), [Spring and Damper](#), [Prescribed Motion](#), [Applied Force and Moment](#), and [Friction](#) subnodes can be added to modify the properties of the joint.

## THREAD PARAMETERS

Enter a value or expression for the **Pitch  $p$**  and **Start  $n$** .



See [Prismatic Joint](#) for these settings — **Attachment Selection**, **Center of Joint**, **Axis of Joint**, **Joint Elasticity**, **Joint Forces and Moments**, and **Sketch**.

---



[Screw Joint Theory](#)

---

## LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics>Joints>Screw Joint**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Global>Joints>Screw Joint**

## *Planar Joint*

---

Use the **Planar Joint** node for 3D components to create a joint between two components such that they are free to translate relative to each other in a plane perpendicular to the joint axis and also free to rotate relative to each other about the same axis.

The [Constraints](#), [Locking](#), [Spring and Damper](#), [Prescribed Motion](#), [Applied Force and Moment](#), and [Friction](#) subnodes can be added to modify the properties of the joint.



See [Prismatic Joint](#) for all settings — **Attachment Selection**, **Center of Joint**, **Axis of Joint**, **Joint Elasticity**, **Joint Forces and Moments**, and **Sketch**.

---



[Planar Joint Theory](#)

---

### **LOCATION IN USER INTERFACE**

*Context Menus*

**Multibody Dynamics**>**Joints**>**Planar Joint**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Global**>**Joints**>**Planar Joint**

## *Ball Joint*

---

Use the **Ball Joint** node for 3D components to create a joint between two components such that they are free to rotate relative to each other about all three axes at the center of joint.

The [Constraints](#), [Locking](#), [Spring and Damper](#), [Prescribed Motion](#), [Applied Force and Moment](#), and [Friction](#) subnodes can be added to modify the properties of the joint.

## AXES OF JOINT

Enter values or expressions for the **Source axis pointing toward center of joint**  $e_0$  and **Destination axis pointing away from center of joint**  $e_{d0}$ . The default is (1, 0, 0) for both axes.



See [Prismatic Joint](#) for these settings — **Attachment Selection**, **Center of Joint**, **Joint Elasticity**, **Joint Forces and Moments**, and **Sketch**.



[Ball Joint Theory](#)

## LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics**>**Joints**>**Ball Joint**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Global**>**Joints**>**Ball Joint**

## Slot Joint

Use the **Slot Joint** node for 3D components to create a joint between two components such that they are free to rotate relative to each other about all three axes at the center of joint and also free to translate relative to each other along the joint translational axis.

The [Constraints](#), [Locking](#), [Spring and Damper](#), [Prescribed Motion](#), and [Applied Force and Moment](#) subnodes can be added to modify the properties of the joint.

## AXES OF JOINT

Enter values or expressions for the **Source axis pointing toward center of joint**  $e_0$  and **Destination axis pointing away from center of joint**  $e_{d0}$ . The default is (1, 0, 0) for both axes. Select an option from the **Joint translation axis** list — **Destination axis** or **Source axis**.



See [Prismatic Joint](#) for these settings — **Attachment Selection**, **Center of Joint**, **Joint Elasticity**, **Joint Forces and Moments**, and **Sketch**.



### LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics>Joints>Slot Joint**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Global>Joints>Slot Joint**

### *Reduced Slot Joint*

Use the **Reduced Slot Joint** node for 2D and 3D components to create joints between two components such that they are free to rotate relative to each other about the joint rotational axis and also free to translate relative to each other along the joint translational axis.

The [Constraints](#), [Locking](#), [Spring and Damper](#), [Prescribed Motion](#), and [Applied Force and Moment](#) subnodes can be added to modify the properties of the joint.

### AXES OF JOINT



For 3D components, Select the **Joint translational axis — Attached on destination** or **Attached on source** then enter values or expressions for **Joint translation axis**  $e_{tr0}$  and **Joint rotational axis**  $e_{rot0}$ . The default is (1,0,0) and (0,1,0) for  $e_{tr0}$  and  $e_{rot0}$ , respectively.



For 2D components, select the **Joint translational axis — Attached on destination** or **Attached on source** then enter values or expressions for **Joint translation axis**  $e_{tr0}$ . The default is (1,0,0).



See [Prismatic Joint](#) for these settings — **Attachment Selection**, **Center of Joint**, **Joint Elasticity**, **Joint Forces and Moments**, and **Sketch**.



---

## LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics>Joints>Reduced Slot Joint**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Global>Joints>Reduced Slot Joint**

---

## *Clearance Joint*

Use the **Clearance Joint** for 2D and 3D components to create joints between two components such that they are free to rotate relative to each other about all three axes and free to translate relative to each other within the specified clearance limit.

### ATTACHMENT SELECTION

Select a **Source**. Select any **Attachment**, **Rigid Domain**, **Spur Gear**, **Helical Gear**, **Bevel Gear**, **Worm Gear**, **Spur Rack**, **Helical Rack**, or **Base Motion** that is defined for the model. The option **Fixed** is also available. When **Fixed** is selected, the source side of the joint is fixed in space, and not connected to any modeled parts. Selecting a base motion gives a behavior similar to using **Fixed**, with the exception that the base now can have a nonzero prescribed displacement, velocity or acceleration.

Select a **Connection point** — **Centroid of source**, **Centroid of selected entities**, or **User defined**.

- For **User defined** also enter the global coordinates of the connection point  $\mathbf{X}_s$ .
- For **Centroid of selected entities** select an **Entity level** — **Boundary**, **Edge** (3D components), or **Point**.



---

When chosen, a default **Source Point: Boundary**, **Source Point: Edge**, or **Source Point: Point** subnode is added.

---

If the source is **Fixed**, the only option available is **User defined**.

Select a **Destination**. Select any [Attachment](#), [Rigid Domain](#), [Spur Gear](#), [Helical Gear](#), [Bevel Gear](#), [Worm Gear](#), [Spur Rack](#), or [Helical Rack](#) that is defined for the model.

Select a **Connection point** — **Centroid of destination**, **Centroid of selected entities**, or **User defined**.

- For **User defined** also enter the global coordinates of the connection point  $\mathbf{X}_d$ .
- For **Centroid of selected entities** select an **Entity level** — **Boundary**, **Edge** (3D components), or **Point**.



When chosen, a default [Destination Point: Boundary](#), [Destination Point: Edge](#), or [Destination Point: Point](#) subnode is added.

In large models, the list of available attachments can become long. You can then create geometric filters to narrow down the search. When the **Use selection filter** check box is selected, two subnodes named [Source Filter](#) and [Destination Filter](#) are added to the joint node. In these subnodes you can make graphic selections of the objects whose attachments should be shown in the **Source** and a **Destination** lists.

#### CLEARANCE SETTINGS

Enter a **Clearance**  $c$ . The default is 0 which corresponds to two connection points at the same geometric position. This is equivalent to a ball joint (3D) or a hinge joint (2D).

Enter the **Penalty factor**  $p_j$ . The default is  $(mbd.crj1.Eequ*(0.1*mbd.diag))$ . The exact naming of the variable depends on the tags of the nodes in the model tree



[Clearance Joint Theory](#)

#### LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics>Joints>More Joints>Clearance Joint**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Global>More Joints>Clearance Joint**

## *Fixed Joint*

---

Use a **Fixed Joint** for 2D and 3D components to prevent any relative motion between the two components. The components connected by this joint are rigidly attached to each other and no additional degrees of freedom are added. The forces or moments required to restrict the relative motion between the two components can be computed.



See [Prismatic Joint](#) for all settings — **Attachment Selection**, **Joint Elasticity**, **Joint Forces and Moments**, **Center of Joint**, and **Axis of Joint**.



[Fixed Joint Theory](#)

### **LOCATION IN USER INTERFACE**

*Context Menus*

**Multibody Dynamics>Joints>More Joints>Fixed Joint**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Global>More Joints>Fixed Joint**

## *Distance Joint*

---

Use a **Distance Joint** for 2D and 3D components to maintain a specified distance between two components. The two components behave as if they are connected through a rigid massless link with ball joints at the ends. No additional degrees of freedom are added by this joint.

It is also possible to specify a change in distance by prescribing the extension. This can be interpreted as if, instead of a rigid link, there is telescopic link connecting the two components.

### **ATTACHMENT SELECTION**

Select a **Source**. Select any [Attachment](#), [Rigid Domain](#), [Spur Gear](#), [Helical Gear](#), [Bevel Gear](#), [Worm Gear](#), [Spur Rack](#), [Helical Rack](#), or [Base Motion](#) that is defined for the model. The option **Fixed** is also available. When **Fixed** is selected, the source side of the

joint is fixed in space, and not connected to any modeled parts. Selecting a base motion gives a behavior similar to using **Fixed**, with the exception that the base now can have a nonzero prescribed displacement, velocity or acceleration.

Select a **Connection point** — **Centroid of source**, **Centroid of selected entities**, or **User defined**.

- For **User defined** also enter the global coordinates of the connection point  $\mathbf{X}_s$ .
- For **Centroid of selected entities** select an **Entity level** — **Boundary**, **Edge** (3D components), or **Point**.



When chosen, a default **Source Point: Boundary**, **Source Point: Edge**, or **Source Point: Point** subnode is added.

The connection point is the exact location where the fictitious ball joint is connected to the source attachment. If the source is **Fixed**, the only option available is **User defined**.

Select a **Destination**. Select any **Attachment**, **Rigid Domain**, **Spur Gear**, **Helical Gear**, **Bevel Gear**, **Worm Gear**, **Spur Rack**, or **Helical Rack** that is defined for the model.

Select a **Connection point** — **Centroid of destination**, **Centroid of selected entities**, or **User defined**.

- For **User defined** also enter the global coordinates of the connection point  $\mathbf{X}_d$ .
- For **Centroid of selected entities** select an **Entity level** — **Boundary**, **Edge** (3D components), or **Point**.



When chosen, a default **Destination Point: Boundary**, **Destination Point: Edge**, or **Destination Point: Point** subnode is added.

In large models, the list of available attachments can become long. You can then create geometric filters to narrow down the search. When the **Use selection filter** check box is selected, two subnodes named **Source Filter** and **Destination Filter** are added to the joint node. In these subnodes you can make graphic selections of the objects whose attachments should be shown in the **Source** and a **Destination** lists.

## MODIFY DISTANCE

Enter an **Extension  $\Delta$** . The default is 0, in which case the distance is equal to the initial geometrical distance between the two connection points.



Distance Joint Theory

## LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics>Joints>More Joints>Distance Joint**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Global>More Joints>Distance Joint**

## *Universal Joint*

A universal joint, also known as Hooke's joint or Cardan joint, is commonly used to transmit rotational motion between shafts that have different orientations. You can use the **Universal Joint** node in 3D components to simulate this behavior. Physically, this joint is made up of pair of hinge joints, oriented perpendicular to each other, connected by a cross shaft or gimbal. The universal joint is an abstract joint where the gimbal is not modeled and both the shafts are directly connected to each other. In general, this joint is used for the purpose of transfer of mechanical power from one shaft to another shaft. The added degrees of freedom in this joint are the relative rotations about the two legs of the gimbal that form joint axis and second axis.



See **Prismatic Joint** for these settings — **Attachment selection**, **Center of Joint**, **Joint Elasticity**, **Joint Forces and Moments**, and **Sketch**.

## SHAFTS ORIENTATION

Select an **Input shaft orientation** — **Specify direction** or **Select a parallel edge**.

- For **Specify direction** enter a value or expression for  $\mathbf{e}_{i0}$ . The default is  $(1, 0, 0)$ . The direction is given in the coordinate system selected in the **Coordinate system selection** section.

- For **Select a parallel edge** an edge can be selected on the **Input Shaft Orientation** subnode, which is added automatically. The vector from the first to the last end of the edge is used to define the input shaft orientation. Any edge in the model can be used. Select the **Reverse direction** check box to reverse the direction of the shaft orientation.

Select an **Output shaft orientation** — **Specify direction** or **Select a parallel edge**.

- For **Specify direction** enter a value or expression for  $\mathbf{e}_{o0}$ . The default is (1, 1, 0). The direction is given in the coordinate system selected in the **Coordinate system selection** section.
- For **Select a parallel edge** an edge can be selected on the **Output Shaft Orientation** subnode, which is added automatically. The vector from the first to the last end of the edge is used to define the output shaft orientation. Any edge in the model can be used. Select the **Reverse direction** check box to reverse the direction of the shaft orientation.

#### **GIMBAL ORIENTATION**

Select an **Auxiliary axis** — **Perpendicular to the plane of shafts** or **User defined**. For **User defined** enter a value or expression for  $\mathbf{e}_{aux0}$  based on space dimension. The direction is given in the coordinate system selected in the **Coordinate system selection** section.

Select an **Axis connecting Gimbal and Input shaft** — **Same as auxiliary axis** or **Rotated auxiliary axis**. For **Rotated auxiliary axis** enter a value or expression for the **Rotation about input shaft**  $\theta_{g0}$ .



[Universal Joint Theory](#)

#### **LOCATION IN USER INTERFACE**

*Context Menus*

**Multibody Dynamics**>**Joints**>**More Joints**>**Universal Joint**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Global**>**More Joints**>**Universal Joint**

## Joint Elasticity

---

The **Joint Elasticity** subnode is automatically added to a joint node when **Elastic joint** is chosen under **Joint Elasticity**. You use it to specify elastic and viscous data for the constrained degrees of freedom in a joint.

The contents of the **Joint Elasticity** settings adapt to the type of joint used.



For specifying elastic or viscous properties for the degrees of freedom which are not constrained by the joint, see [Spring-Damper](#).

### COORDINATE SYSTEM SELECTION

Select a coordinate system in which the elasticity and viscosity matrices are described. The default is **Joint coordinate system**, in which case the directions are the same as those specified for the joint itself.

### ELASTIC DEGREES OF FREEDOM

Select the degrees of freedom that are to be treated as elastic. The default is all constrained degrees of freedom for this type of joint. If you deselect a certain degree of freedom, it reverts to being treated as rigidly constrained. The axis directions always refer to the directions of the joint coordinate system, irrespective of the settings in the **Coordinate System Selection** section.

Check boxes are shown only for the degrees of freedom which are constrained by the current joint type.

### SPRING

Specify the elastic properties of the joint. For joints constraining both translational and rotational degrees of freedom, select **Translational-rotational coupling** to enable input of stiffness terms coupling translation and rotation.

- In  $\mathbf{k}_u$ , enter the translational stiffness matrix.
- In  $\mathbf{k}_\theta$ , enter the rotational stiffness matrix.
- For **Translational-rotational coupling** also enter the coupling matrices  $\mathbf{k}_{u\theta}$  and  $\mathbf{k}_{\theta u}$ .

### VISCOUS DAMPING

In this section, you specify the viscous damping properties of the joint. For joints constraining both translational and rotational degrees of freedom, select

**Translational-rotational coupling** to enable input of viscous terms coupling translation and rotation.

- In  $\mathbf{c}_u$ , enter the translational damping matrix.
- In  $\mathbf{c}_\theta$ , enter the rotational damping matrix.
- For **Translational-rotational coupling** also enter the coupling matrices  $\mathbf{c}_{u\theta}$  and  $\mathbf{c}_{\theta u}$ .



The stiffness and damping matrices are always entered with the largest possible size (for example as 3x3 matrices in 3D). Values entered in directions that are not part of the degrees of freedom declared as elastic are ignored.



The rotational and coupling matrix inputs are not available for the **Ball Joint**, **Slot Joint**, **Hinge Joint** in 2D, and **Reduced Slot Joint** in 2D. These joints do not have any rotational constraints.



[Elasticity in Joints](#)

#### LOCATION IN USER INTERFACE

This node is automatically added when **Elastic joint** is selected in the parent joint node. It cannot be added or removed manually.

This node is also automatically added as a subnode to **Chain Drive**, when **Joint type** is set as **Elastic**. It cannot be added or removed manually.

#### *Constraints*

The **Constraints** subnode is used to impose a restriction on the relative motion between two components. You can specify an upper and lower bound of relative motion. This node can be added more than once if required, and the most restrictive value will then take precedence.

The **Constraints** subnode can be added to the **Prismatic Joint**, **Hinge Joint**, **Cylindrical Joint**, **Screw Joint**, **Planar Joint**, **Ball Joint**, **Slot Joint**, and **Reduced Slot Joint**. The contents of the settings sections will differ between different joint types. Multiple

sections can be present in a single joint, depending on the number of degrees of freedom it has.



The [Theory for Joint Constraints](#) section describes the theory for each of the different types of constraints: translational, rotational, inclination angle, and axial rotation.

## TRANSLATIONAL CONSTRAINTS

This section is available for the [Prismatic Joint](#), [Cylindrical Joint](#), [Screw Joint](#), [Planar Joint](#), [Slot Joint](#), and [Reduced Slot Joint](#) nodes.

- For the **Planar Joint** only, enter values or expressions for the **Initial constraint axis  $e_{c0}$** .
- Enter the **Maximum relative displacement  $u_{\max}$** .
- Enter the **Minimum relative displacement  $u_{\min}$** .
- Select **Constraint parameters** — **Automatic** or **User defined**. For **User defined** enter a value or expression for the **Penalty factor  $p_u$** . The default is  $((1[1/\text{ms}])^2) * (\text{mbd.prj1.ct1.me})$ . The exact naming of the variable depends on the tags of the nodes in the model tree.
- Enter a value or expression for the **Maximum allowable time step  $\Delta t_{\max}$** .

The default is

$((\pi)/(30)) * (\sqrt{(\text{mbd.prj1.ct1.me}) / (\text{mbd.prj1.ct1.p_u})})$ . The exact naming of the variables depends on the tags of the nodes in the model tree.

## ROTATIONAL CONSTRAINTS

This section is available for the [Hinge Joint](#), [Cylindrical Joint](#), [Screw Joint](#), [Planar Joint](#), and [Reduced Slot Joint](#).

- Enter the **Maximum relative rotation  $\theta_{\max}$** .
- Enter the **Minimum relative rotation  $\theta_{\min}$** .
- Select **Constraint parameters** — **Automatic** or **User defined**. For **User defined** enter a value or expression for the **Penalty factor  $p_\theta$** . The default is  $((1[1/\text{ms}])^2) * (\text{mbd.hgj1.ct1.Ie})$ . The exact naming of the variable depends on the tags of the nodes in the model tree.
- Enter a value or expression for the **Maximum allowable time step  $\Delta t_{\max}$** .

The default is

$((\pi)/(30)) * (\sqrt{(\text{mbd.hgj1.ct1.Ie}) / (\text{mbd.hgj1.ct1.p_th})})$ . The exact naming of the variables depends on the tags of the nodes in the model tree.

## INCLINATION ANGLE CONSTRAINT

This section is available for the [Ball Joint](#) and [Slot Joint](#).

- Select the **Initial reference axis** — **Initial source axis**, **Initial destination axis**, or **User defined**. For **User defined** enter values for  $e_{R0}$ .
- Enter a value or expression for the **Maximum inclination angle**  $\theta_{\max}$ .
- Select the **Constraint parameters** — **Automatic** or **User defined**. For **User defined** enter a value or expression for the **Penalty factor**  $p_\theta$ . The default is  $((1[1/ms])^2) * (mbd.blj1.ct1.Ie_th)$ . The exact naming of the variable depends on the tags of the nodes in the model tree.
- Enter a value or expression for the **Maximum allowable time step**  $\Delta t_{\max}$ .

The default is

$((\pi)/(30)) * (\sqrt((mbd.blj1.ct1.Ie_th)/(mbd.blj1.ct1.p_th)))$ . The exact naming of the variables depends on the tags of the nodes in the model tree.

## AXIAL ROTATION CONSTRAINTS

This section is available for the [Ball Joint](#) and [Slot Joint](#).

- Enter the **Maximum axial rotation**  $\psi_{\max}$ .
- Enter the **Minimum axial rotation**  $\psi_{\min}$ .
- Select the **Constraint parameters** — **Automatic** or **User defined**. For **User defined** enter a value or expression for the **Penalty factor**  $p_\psi$ . The default is  $((1[1/ms])^2) * (mbd.blj1.ct1.Ie_psi)$ . The exact naming of the variable depends on the tags of the nodes in the model tree.
- Enter a value or expression for the **Maximum allowable time step**  $\Delta t_{\max}$ .

The default is

$((\pi)/(30)) * (\sqrt((mbd.blj1.ct1.Ie_psi)/(mbd.blj1.ct1.p_psi)))$ .

The exact naming of the variables depends on the tags of the nodes in the model tree.

## LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics>Prismatic Joint>Constraints**

**Multibody Dynamics>Hinge Joint>Constraints**

**Multibody Dynamics>Cylindrical Joint>Constraints**

**Multibody Dynamics>Screw Joint>Constraints**

**Multibody Dynamics>Planar Joint>Constraints**

**Multibody Dynamics>Ball Joint>Constraints**  
**Multibody Dynamics>Slot Joint>Constraints**  
**Multibody Dynamics>Reduced Slot Joint>Constraints**

*Ribbon*

Physics tab with **Prismatic Joint**, **Hinge Joint**, **Cylindrical Joint**, **Screw Joint**, **Planar Joint**, **Ball Joint**, **Slot Joint**, or **Reduced Slot Joint** selected in the model tree:

**Attributes>Constraints**

*Locking*

---

Use the **Locking** subnode to put a lock condition on the relative motion between two components on a joint. Lock conditions are similar to **Constraints**, but after the displacement in the joint degree of freedom reaches the limiting value, it will be fixed to that value. You can specify the locking value for the relative motion in the form of upper and lower bound. This node can be added more than once if required, and the most restrictive value will then take precedence.

The **Locking** subnode can be added to the **Prismatic Joint**, **Hinge Joint**, **Cylindrical Joint**, **Screw Joint**, **Planar Joint**, **Ball Joint**, **Slot Joint**, and **Reduced Slot Joint**.

If multiple degrees of freedom are present in a joint, a lock condition can be put on each of the degrees of freedom. If all the degrees of freedom are locked, the components attached to the joint move as a single object. The contents of the settings sections will differ between different joint types.



The [Theory for Joint Locking](#) section describes the theory for each of the different types of locks: translational, rotational, inclination angle, and axial rotation.

---

**TRANSLATIONAL LOCKING**

This section is available for the **Prismatic Joint**, **Cylindrical Joint**, **Screw Joint**, **Planar Joint**, **Slot Joint**, and **Reduced Slot Joint** nodes.

- For the **Planar Joint** only, enter values or expressions for the **Initial locking axis**  $e_{l0}$ .
- Enter the **Maximum relative displacement**  $u_{\max}$ .
- Enter the **Minimum relative displacement**  $u_{\min}$ .
- Select the **Locking parameters** — **Automatic** or **User defined**. For **User defined** enter a value or expression for the **Penalty factor**  $p_{u1}$ , which supplies a penalty stiffness used

for enforcing the constraint. The default is

$((0.01) * (\text{mbd.prj1.lk1.Eequ}) * (\text{mbd.diag}))$ . Enter a value or expression for the **Penalty factor**  $p_{u2}$ , which supplies damping suppressing vibrations. The default is  $(\text{mbd.prj1.lk1.p\_u1}) * (10[\text{ms}])$ . The exact naming of the variables depends on the tags of the nodes in the model tree.

### ROTATIONAL LOCKING

This section is available for the [Hinge Joint](#), [Cylindrical Joint](#), [Screw Joint](#), [Planar Joint](#), and [Reduced Slot Joint](#).

- Enter the **Maximum relative rotation**  $\theta_{\max}$ .
- Enter the **Minimum relative rotation**  $\theta_{\min}$ .
- Select the **Locking parameters** — **Automatic** or **User defined**. For **User defined** enter a value or expression for the **Penalty factor**  $p_{\theta1}$ .

The default is  $((1e-4) * (\text{mbd.hgj1.lk1.Eequ}) * ((\text{mbd.diag})^{(3)}))$ . The exact naming of the variables depends on the tags of the nodes in the model tree. Enter a value or expression for the **Penalty factor**  $p_{\theta2}$ . The default is

$(\text{mbd.hgj1.lk1.p\_th1}) * (10[\text{ms}])$ . The exact naming of the variable depends on the tags of the nodes in the model tree.

### INCLINATION ANGLE LOCKING

This section is available for the [Ball Joint](#) and [Slot Joint](#).

- Select the **Initial reference axis** — **Initial source axis**, **Initial destination axis**, or **User defined**. For **User defined** enter values for  $\mathbf{e}_{R0}$ .
- Enter a value or expression for the **Maximum inclination angle**  $\theta_{\max}$ .
- Select the **Locking parameters** — **Automatic** or **User defined**. For **User defined** enter a value or expression for the **Penalty factor**  $p_{\theta1}$ .

The default is  $((1e-4) * (\text{mbd.blj1.lk1.Eequ}) * ((\text{mbd.diag})^{(3)}))$ . The exact naming of the variables depends on the tags of the nodes in the model tree. Enter a value or expression for the **Penalty factor**  $p_{\theta2}$ . The default is

$(\text{mbd.blj1.lk1.p\_th1}) * (10[\text{ms}])$ . The exact naming of the variable depends on the tags of the nodes in the model tree.

### AXIAL ROTATION LOCKING

This section is available for the [Ball Joint](#) and [Slot Joint](#).

- Enter the **Maximum axial rotation**  $\psi_{\max}$ .
- Enter the **Minimum axial rotation**  $\psi_{\min}$ .

- Select the **Locking parameters** — **Automatic** or **User defined**. For **User defined** enter a value or expression for the **Penalty factor**  $p_{\psi 1}$ .

The default is  $(1e-4) * (mbd.blj1.1k1.Eequ) * (mbd.diag)^{(3)}$ . The exact naming of the variables depends on the tags of the nodes in the model tree. Enter a value or expression for the **Penalty factor**  $p_{\psi 2}$ . The default is

$(mbd.blj1.1k1.p_psi1) * (10[ms])$ . The exact naming of the variable depends on the tags of the nodes in the model tree.

## LOCATION IN USER INTERFACE

### Context Menus

**Multibody Dynamics>Prismatic Joint>Locking**

**Multibody Dynamics>Hinge Joint>Locking**

**Multibody Dynamics>Cylindrical Joint>Locking**

**Multibody Dynamics>Screw Joint>Locking**

**Multibody Dynamics>Planar Joint>Locking**

**Multibody Dynamics>Ball Joint>Locking**

**Multibody Dynamics>Slot Joint>Locking**

**Multibody Dynamics>Reduced Slot Joint>Locking**

### Ribbon

Physics tab with **Prismatic Joint**, **Hinge Joint**, **Cylindrical Joint**, **Screw Joint**, **Planar Joint**, **Ball Joint**, **Slot Joint**, or **Reduced Slot Joint** selected in the model tree:

### Attributes>Locking

## Spring and Damper

Use the **Spring and Damper** subnode to apply elastic and viscous forces to restrict and control the relative motion on joint. You can also use a dashpot to model various kind of losses in the joint. This node can be added more than once if required, and the effects will then be superimposed. Each joint type has different settings sections.

The **Spring and Damper** subnode can be added to the **Prismatic Joint**, **Hinge Joint**, **Cylindrical Joint**, **Screw Joint**, **Planar Joint**, **Ball Joint**, **Slot Joint**, and **Reduced Slot Joint**. The contents of the settings sections will differ between different joint types



The [Theory for Springs and Dampers on Joints](#) section describes the theory for each of the different types of springs and dampers: translational, rotational, inclination angle, and axial rotation.

### SPRING AND DAMPER: TRANSLATIONAL

This section is available for the [Prismatic Joint](#), [Cylindrical Joint](#), [Screw Joint](#), [Planar Joint](#), [Slot Joint](#), and [Reduced Slot Joint](#) nodes.

Select a **Spring type** — **Spring constant** or **Force as function of extension**.

- For **Spring constant**, enter a value of  $k_u$ .
- For **Force as function of extension**, enter an expression for  $F_s$ . The expression can be a function of the spring extension. The built-in variable for the spring extension has the form `<physicsName>.<JointTag>.<SpringAndDamperTag>.us` (for example, `mbd.prj1.sd1.us`).

To add a predeformation of the spring, enter **Predeformation**  $u_0$ . To add viscous damping in a dynamic analysis, enter a value or expression for the **Damping coefficient**  $c_u$ .

For the **Planar Joint**, enter values or expressions for the **Initial spring axis**  $e_{s0}$ . Also select the **Initial damper axis** — **Same as spring axis** or **User defined**. For **User defined** enter values or expressions for  $e_{d0}$ .

### SPRING AND DAMPER: ROTATIONAL

This section is available for the [Hinge Joint](#), [Cylindrical Joint](#), [Screw Joint](#), [Planar Joint](#), and [Reduced Slot Joint](#) nodes.

Select a **Spring type** — **Spring constant** or **Moment as function of rotation**.

- For **Spring constant**, enter a value of  $k_\theta$ .
- For **Moment as function of rotation**, enter an expression for  $M_s$ . The expression can be a function of the spring rotation. The built-in variable for the spring rotation has the form `<physicsName>.<JointTag>.<SpringAndDamperTag>.ths` (for example, `mbd.hgj1.sd1.ths`).

To add a predeformation of the spring, enter **Predeformation**  $\theta_0$ . To add viscous damping in a dynamic analysis, enter a value or expression for the **Damping coefficient**  $c_\theta$ .

### SPRING AND DAMPER: INCLINATION ANGLE

This section is available for [Ball Joint](#) and [Slot Joint](#) nodes.

Select the **Initial reference axis** — **Initial source axis**, **Initial destination axis**, or **User defined**. For **User defined** enter values for  $e_{R0}$ .

Select a **Spring type** — **Spring constant** or **Moment as function of rotation**.

- For **Spring constant**, enter a value of  $k_\theta$ .
- For **Moment as function of rotation**, enter an expression for  $M_s$ . The expression can be a function of the spring rotation. The built-in variable for the spring rotation has the form `<physicsName>.<JointTag>.<SpringAndDamperTag>.ths` (for example, `mbd.blj1.sd1.ths`).

To add a predeformation of the spring, enter **Predeformation**  $\theta_0$ . To add viscous damping in a dynamic analysis, enter a value or expression for the **Damping coefficient**  $c_\theta$ .

#### **SPRING AND DAMPER: AXIAL ROTATION**

This section is available for [Ball Joint](#) and [Slot Joint](#) nodes.

Select a **Spring type** — **Spring constant** or **Moment as function of rotation**.

- For **Spring constant**, enter a value of  $k_\psi$ .
- For **Moment as function of rotation**, enter an expression for  $M_s$ . The expression can be a function of the spring rotation. The built-in variable for the spring rotation has the form `<physicsName>.<JointTag>.<SpringAndDamperTag>.psis` (for example, `mbd.blj1.sd1.psis`).

To add a predeformation of the spring, enter **Predeformation**  $\psi_0$ . To add viscous damping in a dynamic analysis, enter a value or expression for the **Damping coefficient**  $c_\psi$ .

#### **LOCATION IN USER INTERFACE**

*Context Menus*

[Multibody Dynamics>Prismatic Joint>Spring and Damper](#)  
[Multibody Dynamics>Hinge Joint>Spring and Damper](#)  
[Multibody Dynamics>Cylindrical Joint>Spring and Damper](#)  
[Multibody Dynamics>Screw Joint>Spring and Damper](#)  
[Multibody Dynamics>Planar Joint>Spring and Damper](#)  
[Multibody Dynamics>Ball Joint>Spring and Damper](#)  
[Multibody Dynamics>Slot Joint>Spring and Damper](#)  
[Multibody Dynamics>Reduced Slot Joint>Spring and Damper](#)

*Ribbon*

Physics tab with **Prismatic Joint**, **Hinge Joint**, **Cylindrical Joint**, **Screw Joint**, **Planar Joint**, **Ball Joint**, **Slot Joint**, or **Reduced Slot Joint** selected in the model tree:

### Prescribed Motion

---

Use the **Prescribed Motion** subnode to prescribe the relative motion on a joint. You can also prescribe the motion by giving the time derivatives of the joint degrees of freedom, for example, velocity for translation motion and angular velocity for rotational motion. Activation conditions can be used to prescribe the motion for only a certain duration. Axis directions are interpreted in the selected coordinate system.

The **Prescribed Motion** subnode can be added to the [Prismatic Joint](#), [Hinge Joint](#), [Cylindrical Joint](#), [Screw Joint](#), [Planar Joint](#), [Ball Joint](#), [Slot Joint](#), and [Reduced Slot Joint](#). The contents of the settings sections will differ between different joint types



The [Theory for Prescribed Motion on Joints](#) section describes the theory for each of the different types of prescribed motion: translational and rotational.

#### PREScribed TRANSLATIONAL MOTION

This section is available for the [Prismatic Joint](#), [Cylindrical Joint](#), [Screw Joint](#), [Planar Joint](#), [Slot Joint](#), and [Reduced Slot Joint](#) nodes.

Select an option from the **Prescribed motion through** list — **Displacement** or **Velocity**.

- For **Displacement** enter a value or expression for  $u_p$ .
- For **Velocity** enter a value or expression for  $v_p$ .

Select an **Activation condition** — **Always active** or **Conditionally active**. For **Conditionally active** enter a **Deactivation indicator expression**  $i_{up}$ . For the joints having more than one degree of freedom (Cylindrical Joint, Screw Joint, Planar Joint, Slot Joint, and Reduced Slot Joint), **Never active** is also an option.

For the [Planar Joint](#), enter values or expressions for the **Initial prescribed motion axis**  $e_{p0}$ .

#### PREScribed ROTATIONAL MOTION

This section is available for the [Hinge Joint](#), [Cylindrical Joint](#), [Screw Joint](#), [Planar Joint](#), [Ball Joint](#), [Slot Joint](#), and [Reduced Slot Joint](#) nodes.

For the [Ball Joint](#) and [Slot Joint](#), enter values or expressions for the **Initial prescribed motion axis**  $e_{p0}$ .

Select an option from the **Prescribed motion through** list — **Rotation** or **Angular velocity**.

- For **Rotation** enter a value or expression for  $\theta_p$ . In this case, the variable name is  $\phi_p$  for the Ball Joint and Slot Joint.
- For **Angular velocity** enter a value or expression for  $\omega_p$ .

Select an **Activation condition** — **Always active** or **Conditionally active**. For **Conditionally active** enter a **Deactivation indicator expression**  $i_{\theta_p}$ . (In this case, the variable name is  $i_{\phi_p}$  for the Ball Joint and Slot Joint). For the joints having more than one degree of freedom (Cylindrical Joint, Screw Joint, Planar Joint, Slot Joint, and Reduced Slot Joint), **Never active** is also an option.

#### **REACTION FORCE SETTINGS**

Select **Evaluate reaction forces** to compute the reaction force caused by the prescribed motion. The default is to not compute the reaction force. When selected, the prescribed motion is implemented as a weak constraint. If the prescribed motion is not active all through the study step, because **Activation condition** is **Conditionally active**, this setting is ignored. Reaction forces are never computed in that case.

Select **Apply reaction only on joint variables** to use a unidirectional constraint for enforcing the prescribed motion. The default is that bidirectional constraints are used. This setting is useful in a situation where a bidirectional constraint would give an unwanted coupling in the equations. This would happen if the prescribed value of the motion is a variable solved for in other equations.

If the motion is prescribed as a velocity, the reaction force is applied on the joint variables only (unidirectional constraint) irrespective of the setting of this control.

#### **LOCATION IN USER INTERFACE**

*Context Menus*

**Multibody Dynamics>Prismatic Joint>Prescribed Motion**

**Multibody Dynamics>Hinge Joint>Prescribed Motion**

**Multibody Dynamics>Cylindrical Joint>Prescribed Motion**

**Multibody Dynamics>Screw Joint>Prescribed Motion**

**Multibody Dynamics>Planar Joint>Prescribed Motion**

**Multibody Dynamics>Ball Joint>Prescribed Motion**

**Multibody Dynamics>Slot Joint>Prescribed Motion**

**Multibody Dynamics>Reduced Slot Joint>Prescribed Motion**

### Ribbon

Physics tab with **Prismatic Joint**, **Hinge Joint**, **Cylindrical Joint**, **Screw Joint**, **Planar Joint**, **Ball Joint**, **Slot Joint**, or **Reduced Slot Joint** selected in the model tree:

#### Attributes>Prescribed Motion

### *Applied Force and Moment*

---

Use the **Applied Force and Moment** subnode to apply forces and moments on either the source or destination attachment of a joint, or directly on the joint degrees of freedom.

The **Applied Force and Moment** subnode can be added to the **Prismatic Joint**, **Hinge Joint**, **Cylindrical Joint**, **Screw Joint**, **Planar Joint**, **Ball Joint**, **Slot Joint**, and **Reduced Slot Joint**. The contents of the settings sections will differ between different joint types.

#### APPLIED ON

Select **Source attachment**, **Destination attachment**, or **Joint** to apply force and moment on the selected attachments or directly on the joint degrees of freedom.

#### LOCATION

In this section you define the point of action of the force. This section is not available when the force is applied on **Joint**.

Select **Center of attachment**, **Center of joint**, or **User defined**. For **User defined** enter values or expressions for the **Location  $\mathbf{X}_p$** .

#### APPLIED FORCE AND MOMENT

For **Source attachment** or **Destination attachment**, enter values or expressions for the applied force **F** and applied moment **M** (3D components) or  **$M_z$** . The forces or moments that you prescribe are interpreted in the selected coordinate system.

When the load is applied on **Joint**, the input in this section depends on the joint type. For all joints, a scalar force or moment acts directly on the single translational or rotational degree of freedom in the joint, while vector inputs have directions as given by the selected coordinate system.

- For **Prismatic Joint**, enter the scalar force **F**.
- For **Hinge Joint**, enter the scalar moment **M**.
- For **Cylindrical Joint**, **Screw Joint** or **Reduced Slot Joint**, enter the scalar force **F** and moment **M**.

- For **Planar Joint**, enter the force vector  $\mathbf{F}$ , and the scalar moment  $M$ .
- For **Ball Joint**, enter the moment vector  $\mathbf{M}$ .
- For **Slot Joint**, enter the scalar force  $F$  and the moment vector  $\mathbf{M}$ .

---



Theory for Applied Forces and Moments on Joints

---

#### LOCATION IN USER INTERFACE

##### *Context Menus*

**Multibody Dynamics**>**Prismatic Joint**>**Applied Force and Moment**

**Multibody Dynamics**>**Hinge Joint**>**Applied Force and Moment**

**Multibody Dynamics**>**Cylindrical Joint**>**Applied Force and Moment**

**Multibody Dynamics**>**Screw Joint**>**Applied Force and Moment**

**Multibody Dynamics**>**Planar Joint**>**Applied Force and Moment**

**Multibody Dynamics**>**Ball Joint**>**Applied Force and Moment**

**Multibody Dynamics**>**Slot Joint**>**Applied Force and Moment**

**Multibody Dynamics**>**Reduced Slot Joint**>**Applied Force and Moment**

##### *Ribbon*

Physics tab with **Prismatic Joint**, **Hinge Joint**, **Cylindrical Joint**, **Screw Joint**, **Planar Joint**, **Ball Joint**, **Slot Joint**, or **Reduced Slot Joint** selected in the model tree:

**Attributes**>**Applied Force and Moment**

#### *Friction*

---

Use the **Friction** subnode to model friction losses in the joint.

The **Friction** subnode can be added to the **Prismatic Joint**, **Hinge Joint**, **Cylindrical Joint**, **Screw Joint**, **Planar Joint**, and **Ball Joint**.

---



Friction can only be applied to the joints in a time-dependent analysis.

---

#### **FRICITION**

Enter a **Friction coefficient**  $\mu$ .

Enter a **Characteristic slip velocity**  $v_0$ . The default is `mbd.diag*(1e-3[1/s])`, where the `mbd.diag` variable contains the diagonal of the bounding box of the geometry. The characteristic slip velocity should be small when compared to actual velocities in the joint. It is the velocity with which the joint slides even under sticking conditions.

- For the [Hinge Joint](#), enter a **Hinge radius**  $r$ .
- For the [Cylindrical Joint](#), enter a **Cylinder radius**  $r$ .
- For the [Screw Joint](#), enter a **Screw radius**  $r$ .
- For the [Ball Joint](#), enter a **Ball radius**  $r$ .

The radius is used for computing the friction moment in the joints that have a rotational degree of freedom.

For the [Planar Joint](#), select a **Friction model** — **Total force at center of joint** or **Distributed force on selected boundaries**. By distributing the friction forces, it is possible to obtain a moment caused by friction for a rotational relative motion. When **Distributed force on selected boundaries** is selected, a [Contact Area](#) subnode is added by default. In that subnode you make the selection of the boundaries on which the friction forces should be distributed.

#### **NORMAL FORCE**

In this section you supply details about how to compute the force acting on the joint, to which the friction loss is proportional.



For a detailed discussion of the effect of the settings for the normal force for each joint type, see [Theory for Friction in Joints](#)

---

The **Contribution from joint forces** check box is selected by default, which implies that the friction forces are computed based on normal forces determined from the forces acting the joint. If you click to clear the check box the friction force is independent of the forces acting on the joint, and the following parameter selection is not required.

Select an option from the **Normal force defined through** list — **Attachment reaction forces**, **Joint forces and moments**, or for the [Prismatic Joint](#), **Joint force in specified direction**.



---

For the [Ball Joint](#), no further selection is required.

---



The alternative **Attachment reaction forces** is not available if the joint is attached to a rigid domain or to an attachment in a Shell or Beam interface.

For **Joint forces and moments** enter values for the **Joint moment contribution factors**  $\alpha_{11}$  (for the [Prismatic Joint](#) only),  $\alpha_{12}$ , and  $\alpha_{13}$ .

- For the [Prismatic Joint](#), and if **Joint force in specified direction** is selected, enter values for each space coordinate for the **Direction of surface normal**  $e_{n0}$ .

By selecting **Contribution from interference fit** check box, you can define the normal forces by pressure or total force from an assumed interference fit.

Select an option from the **Normal force defined through** list — **Pressure** or **Total force**.

- For **Pressure** enter a value for  $p$ . Also enter a **Contact area A**.
- For **Total force** enter a value for  $F_n$ .

#### ADVANCED

Enter values or expressions for the:

- Additional sliding resistance**  $F_{f,r}$ .
- Maximum friction force**  $F_{f,max}$ . The default is **Inf**, which means that no upper limit is imposed on the friction force.



#### Theory for Friction in Joints

#### LOCATION IN USER INTERFACE

##### Context Menus

**Multibody Dynamics>Prismatic Joint>Friction**

**Multibody Dynamics>Hinge Joint>Friction**

**Multibody Dynamics>Cylindrical Joint>Friction**

**Multibody Dynamics>Screw Joint>Friction**

**Multibody Dynamics>Planar Joint>Friction**

**Multibody Dynamics>Ball Joint>Friction**

##### Ribbon

Physics tab with **Prismatic Joint**, **Hinge Joint**, **Cylindrical Joint**, **Screw Joint**, **Planar Joint**, or **Ball Joint** selected in the model tree:

## Attributes>Friction

### Contact Area

---

Select the boundaries to be used for distributing the friction forces in a [Planar Joint](#) when **Friction model** is chosen as **Distributed force on selected boundaries**.

#### LOCATION IN USER INTERFACE

This node is automatically added when **Distributed force on selected boundaries** is selected in the parent joint node. It cannot be added or removed manually.

### Joint Axis

---

Use the **Joint Axis** node to select any edge in the geometry which is parallel to the axis of the joint.

This setting is available for the [Prismatic Joint](#), [Hinge Joint](#), [Cylindrical Joint](#), [Screw Joint](#), and [Planar Joint](#).



### Joint Theory

---

#### LOCATION IN USER INTERFACE

This node is automatically added when **Select a parallel edge** is selected in the parent joint node. It cannot be added or removed manually.

### Input Shaft Orientation

---

Select any edge in the geometry which is parallel to the ingoing shaft in a [Universal Joint](#).



### Universal Joint Theory

---

#### LOCATION IN USER INTERFACE

This node is automatically added when **Select a parallel edge** is selected under **Input Shaft Orientation** in the parent **Universal Joint** node. It cannot be added or removed manually.

## *Output Shaft Orientation*

---

Select any edge in the geometry which is parallel to the outgoing shaft in a [Universal Joint](#).

---



[Universal Joint Theory](#)

---

### **LOCATION IN USER INTERFACE**

This node is automatically added when **Select a parallel edge** is selected under **Output Shaft Orientation** in the parent **Universal Joint** node. It cannot be added or removed manually.

## *Center of Joint Nodes*

---



[Positioning the Joint Center](#)

---

### **CENTER OF JOINT: BOUNDARY**

Select a set of boundaries whose centroid represents the center of the joint.

### **CENTER OF JOINT: EDGE**

Select a set of edges whose centroid represents the center of the joint.

### **CENTER OF JOINT: POINT**

Select a set of points whose centroid represents the center of the joint.

### **LOCATION IN USER INTERFACE**

These nodes are automatically added when **Centroid of selected entities** is chosen in the parent joint node. The type of node generated depends in the setting of **Entity level**. They cannot be added or removed manually.

## *Source Point Nodes*

---

### **SOURCE POINT: BOUNDARY**

Select a set of boundaries whose centroid represents the source connection point.

**SOURCE POINT: EDGE**

Select a set of edges whose centroid represents the source connection point.

**SOURCE POINT: POINT**

Select a set of points whose centroid represents the source connection point.

**LOCATION IN USER INTERFACE**

When **Centroid of selected entities** is chosen under **Connection point** for the source attachment in the parent **Clearance Joint** or **Distance Joint**, a subnode will be generated based on the setting of **Entity level**. They cannot be added or removed manually.

This node is also available when **Centroid of selected entities** is chosen under **Center** in the **Source** section in the parent **Rigid Body Contact**.

---

*Destination Point Nodes***DESTINATION POINT: BOUNDARY**

Select a set of boundaries whose centroid represents the destination connection point.

**DESTINATION POINT: EDGE**

Select a set of edges whose centroid represents the destination connection point.

**DESTINATION POINT: POINT**

Select a set of points whose centroid represents the destination connection point.

**LOCATION IN USER INTERFACE**

When **Centroid of selected entities** is chosen under **Connection point** for the destination attachment in the parent **Clearance Joint** or **Distance Joint**, a subnode will be generated based on the setting of **Entity level**. They cannot be added or removed manually.

This node is also available when **Centroid of selected entities** is chosen under **Center** in the **Destination** section in the parent **Rigid Body Contact**.

## *Source Filter*

---

When you select domains in this node, the **Source** list in the parent node will only contain attachments and rigid domains that are part of the selection. The selection rules are:

- If any part of an active rigid domain is within the selection, that rigid domain will appear in the list.
- If any boundary of an active attachment is adjacent to the selected domains, that attachment will appear in the list.
- If the selection is such that no attachments or rigid domains satisfy the rules above, the **Source** list in the parent node will only contain the entry **None** (if the parent node is a joint), or **Select a connection point** (if the parent node is a **Spring-Damper**).
- If the selection is such that some attachments or rigid domains satisfy the rules above, the **None** entry in the **Source** list in the parent node will be removed, and one of the attachments or rigid domains will be selected as default. This means that if the selection is such that the **Source** list has only one unique item, this item is automatically selected.
- If the selection is empty, the behavior is as if **Use selection filter** were not selected.

### **LOCATION IN USER INTERFACE**

This node is automatically created when the **Use selection filter** check box is selected in the **Attachment Selection** section in the parent node. The parent can be any joint node, or a **Spring-Damper** node. The **Source Filter** node cannot be added or removed manually.

## *Destination Filter*

---

When you select domains in this node, the **Destination** list in the parent node will only contain attachments and rigid domains which are part of the selection. The selection rules are described in [Source Filter](#).

### **LOCATION IN USER INTERFACE**

This node is automatically created when the **Use selection filter** check box is selected in the **Attachment Selection** section in the parent node. The parent can be any joint node, or a **Spring-Damper** node. The **Destination Filter** node cannot be added or removed manually.

## Base Motion

---

Sometimes the components described by a multibody dynamics assembly are attached to a common “ground”, which is not fixed in space but has a uniform movement. Use **Base Motion** (➡️➡️) to specify the motion (displacement, velocity, or acceleration) of the environment to which the assembly is connected. The **Base Motion** node can then be referenced in the selection of source attachment for the joints and springs that connect to the base.

### BASE MOTION

Select a **Base motion type** — **Displacement**, **Velocity**, or **Acceleration**.

Enter a value based on the selection:

- For **Displacement**, enter  $\mathbf{u}_b$ .
- For **Velocity**, enter  $\mathbf{v}_b$ .
- For **Acceleration**, enter  $\mathbf{a}_b$ .

The values are given in the coordinate system selected in the **Coordinate system selection** section.

### INITIAL BASE DISPLACEMENT

This section is available when **Velocity** or **Acceleration** is selected as the **Base motion type**. Then enter a value for  $\mathbf{u}_{b0}$  to prescribe a nonzero initial value for the base displacement. The values are given in the coordinate system selected in the **Coordinate system selection** section.

### INITIAL BASE VELOCITY

This section is available when **Acceleration** is selected as the **Base motion type**. Then enter a value for  $\mathbf{v}_{b0}$  to prescribe a nonzero initial value for the base velocity. The values are given in the coordinate system selected in the **Coordinate system selection** section.

### LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics>Base Motion**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Global>Multibody Dynamics>Base Motion**

## Rigid Body Contact

---

Use the **Rigid Body Contact** node (

If friction is to be included, add a [Friction \(Rigid Body Contact\)](#) subnode.



When a **Rigid Body Contact** node is present in your model, stationary and time dependent studies are geometrically nonlinear. The **Include geometric nonlinearity** check box in the study step **Settings** window is selected and cannot be cleared.

### SKETCH

This section shows a sketch of the rigid body contact for different settings in the **Destination** section. The sketch highlights the source center, the source radius, the destination center, the destination radius, the distance between source center and destination contact point, and the gap.

### SOURCE

In this section, you select the rigid source.

Specify the **Shape** of the source. The currently available choices are **Spherical** in 3D, and **Circular** in 2D.

From the **Source** list, select a **Rigid Domain** node available in the Multibody Dynamics interface. The default value is **None**. Enter the radius of the **Source** rigid body using the **Radius** input.

Select the **Center** — **Centroid of source**, **Centroid of selected entities**, or **User defined**.

- For **User defined** enter the global coordinates of the center  $\mathbf{X}_s$ .
- For **Centroid of selected entities** select an **Entity level** — **Boundary**, **Edge** (3D components), or **Point**. When selected, a **Source Point: Boundary**, **Source Point: Edge**, or **Source Point: Point** subnode is added.



- **Source Point: Boundary**
- **Source Point: Edge**
- **Source Point: Point**

## DESTINATION

In this section, you set the properties of the destination part of the contact.

Specify the **Shape** — **Spherical** (3D), **Circular** (2D) or **Arbitrary**.

Choose **Spherical** to model structural contact between two rigid bodies of spherical shape in 3D. Similarly, by setting **Shape** to **Circular** in 2D, you can model the structural contact between two circular rigid bodies.

If the **Shape** is set to **Spherical** (3D) or **Circular** (2D), a **Destination** list is available. Select a **Rigid Domain**, different from the one selected as **Source**. The default value is **None**. Enter the radius of the **Destination** rigid body using the **Radius** input.

Select the **Center** — **Centroid of destination**, **Centroid of selected entities**, or **User defined**.

- For **User defined** also enter the global coordinates of the center  $\mathbf{X}_d$ .
- For **Centroid of selected entities** select an **Entity level** — **Boundary**, **Edge** (3D components), or **Point**. When selected, a **Destination Point: Boundary**, **Destination Point: Edge**, or **Destination Point: Point** subnode is added.



- **Destination Point: Boundary**
- **Destination Point: Edge**
- **Destination Point: Point**

Select **Use inside boundaries for contact** if the location of the source is inside the destination sphere. In this case, the inner boundaries of the destination are used for the gap distance and contact force calculations.

If the **Shape** is set to **Arbitrary**, you can select a set of destination boundaries for the contact in the **Boundary Selection, Destination** section. These boundaries need not necessarily be part of any **Rigid Domain** nodes. The boundary selection can be specified for 3D and 2D models.

### CONTACT SETTINGS

Under **Formulation**, you select the algorithm used for computing the contact. Select **Penalty** or **Penalty, dynamic** (default).

The **Penalty, dynamic** is intended for use in time-dependent studies to model dynamic contact.

#### *Settings for the Penalty Method*

Select the type of **Penalty factor control** — **Automatic** or **User defined**.

- **Automatic** provides predefined values for the penalty factor. This value is multiplied by a **Penalty factor multiplier**  $f_p$  which is the input having default value as 1.
- For **User defined**, enter a **Penalty factor**  $p_n$ . The default value is  $(1e10[N/m^2]) * mbd.diag$ , where **mbd.diag** is the diagonal of the bounding box of the geometry.

#### *Settings for the Penalty, Dynamic Method*

Select the type of **Penalty factor control** — **Automatic**; **Viscous only** or **User defined**. The settings are the same as for the standard penalty method, except for the **Viscous only** option. By selecting **Viscous only**, the stiffness terms of the penalty contact are omitted from the formulation.

Select the type of **Viscous penalty factor control** — **Automatic** or **User defined**. For **Automatic**, enter a value for the **Characteristic time**  $\tau_n$ . This value can be used as a multiplier for the viscous penalty factor, but should as a rule-of-thumb be in the same order of magnitude as the duration of the contact event. When **User defined** is selected, enter the **Viscous penalty factor**  $p_{nv}$ . The default value is  $(1e10[N/m^2]) * mbd.diag * 1[ms]$ , where **mbd.diag** is the diagonal of the bounding box of the geometry.

When **Formulation** is set to **Penalty, dynamic**, select **Compute viscous contact dissipation** to compute and store the energy dissipated by the viscous contact. This adds a global dependent variable with an ODE for energy integration purpose.

---



## Rigid Body Contact Theory

---

### LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics>Rigid Body Contact**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Global>Multibody Dynamics>Rigid Body Contact**

### *Friction (Rigid Body Contact)*

---

By adding a **Friction** subnode to **Rigid Body Contact** node, you can model frictional forces and losses between the two contacting rigid bodies. There can only be one **Friction** subnode under a **Rigid Body Contact** node.

---



Friction can only be applied to a rigid body contact in a time-dependent study type.

---

### FRICITION

- Enter a **Friction coefficient**  $\mu$ .
- Enter a **Characteristic slip velocity**  $v_0$ . The default is `mbd.diag*(1e-3[1/s])`, where `mbd.diag` is the diagonal of the bounding box of the geometry. The characteristic slip velocity should be small compared to the actual slip velocity at the contact point. This is the velocity with which the contact surfaces slide, even under sticking conditions.

### ADVANCED

Enter values or expressions for the:

- **Additional sliding resistance**  $F_{fr}$ .

- **Maximum friction force**  $F_{f,\max}$ . The default is **Inf**, which means that no upper limit is imposed on the friction force.
- Select **Compute frictional dissipation** to compute and store the energy dissipated by the friction. This adds a global dependent variable with an ODE for energy integration purpose.



## Theory of Friction in Rigid Body Contact

---

### LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics>Rigid Body Contact>Friction**

*Ribbon*

Physics tab with **Rigid Body Contact** selected in the model tree:

**Attributes>Friction**

### *Cam-Follower*

---

Use the **Cam-Follower** ( ) node to force a point of one moving body to maintain contact with a curve or surface of another moving body. This connection node is generic in nature and allows any point to follow a curve or surface, including a typical radial or cylindrical cam-follower connection used in various mechanisms.

The cam-follower connection models a simplified contact and establishes a higher kinematic pair between a cam and its follower.

### **BOUNDARY SELECTION, CAM**

Select a set of boundaries which is being followed by a point called 'follower'. This option is available only when **Selection type** is set to **Boundary** in the **Cam** section. The boundary selection can be specified for 3D and 2D models.

### **EDGE SELECTION, CAM**

Select a set of edges which is being followed by a point called "follower". This option is available only when **Selection type** is set to **Edge** in the **Cam** section. The edge selection can be specified only for 3D models.

### **DOMAIN SELECTION, FOLLOWER**

Select a set of domains such that the centroid of the selected entities follows the cam boundaries or edges with a certain optional offset. This option is available only when **Connection point** is set to **Centroid of selected entities** and **Entity level** is set to **Domain** in the **Follower** section.

### **BOUNDARY SELECTION, FOLLOWER**

Select a set of boundaries such that the centroid of the selected entities follows the cam boundaries or edges with a certain optional offset. This option is available only when **Connection point** is set to **Centroid of selected entities** and **Entity level** is set to **Boundary** in the **Follower** section.

### **EDGE SELECTION, FOLLOWER**

Select a set of edges such that the centroid of the selected entities follows the cam boundaries or edges with a certain optional offset. This option is available only when **Connection point** is set to **Centroid of selected entities** and **Entity level** is set to **Edge** in the **Follower** section. The edge selection can be specified only for 3D models.

### **POINT SELECTION, FOLLOWER**

Select a set of points such that the centroid of the selected entities follows the cam boundaries or edges with a certain optional offset. This option is available only when **Connection point** is set to **Centroid of selected entities** and **Entity level** is set to **Point** in the **Follower** section.

### **POINT SELECTION, FOLLOWER (SINGLETON)**

Select a single point, such that the selected point follows the cam boundaries or edges with a certain optional offset. This option is available when **Connection point** is set to **Geometric Point** in the **Follower** section.

## **CAM**

The cam selection can be defined on both rigid and elastic bodies. Thus, cam boundaries and edges can have any kind of rigid body motion as well as deformation. When defining a cam selection, it is also be possible to use boundaries or edges that are defined outside the physics interface.

Select the **Selection type** — **Boundary** or **Edge**.

Enter a value of offset based on the selection:

- For **Boundary**, enter **Follower point offset from cam boundary**,  $X_{\text{offset}}$ , as a scalar value.
- For **Edge**, enter **Follower point offset from cam edge**,  $X_{\text{offset}}$ , as a vector.

### **FOLLOWER**

A follower is defined through a point which follows the cam boundaries or edges with a certain optional offset. The follower point can be a part of rigid or elastic bodies. If there is a need for multiple points to follow the cam boundaries or edges, multiple instances of the **Cam-Follower** feature should be used.

Select the **Connection point** — **Centroid of selected entities** or **Geometric point**.

When **Centroid of selected entities** is selected as a **Connection point**, specify the **Entity level** as **Domain**, **Boundary**, **Edge**, or **Point**.

### **CONNECTION FORCE**

Select the **Connection force** — **Do not compute**, **Computed using weak constraints**, or **Computed using penalty method** in order to disable or enable the computation of the cam-follower connection force.

In case **Computed using penalty method** is selected, specify a value of **Penalty factor** ( $p_c$ ) used for the connection constraint.

### **ACTIVATION CONDITION**

Select **Connection state** — **Always active**, or **Conditionally active**.

When **Conditionally active** is selected, specify a value of the **Connection tolerance** ( $\Delta$ ) used in the connection constraint. This option can be used to deactivate the connection between the two bodies based on their spatial positions and the value of the connection tolerance.



Cam-Follower Theory

---

### **LOCATION IN USER INTERFACE**

*Context Menus*

**Multibody Dynamics>Cam-Follower**

### Ribbon

Physics tab with **Multibody Dynamics** selected:

**Global>Multibody Dynamics>Cam-Follower**

### Spur Gear

---

Use the **Spur Gear** node to model a spur gear 2D and 3D. The gear is assumed to be rigid. Select one or more domains to make them a part of the gear.

A spur gear can be connected to the following gear types:

- Another **Spur Gear**, or a **Helical Gear** through the **Gear Pair** node.
- A **Worm Gear** through the **Worm and Wheel** node.
- A **Spur Rack**, or a **Helical Rack** through the **Rack and Pinion** node.

Any two gears or rack can be connected to each other only if they satisfy the [Gear Pair Compatibility Criteria](#).

The gear has the following subnodes to constraint the displacement and rotation, to apply forces and moments, and to add mass and moment of inertia:

---

These subnodes are described in the *Structural Mechanics Module User's Guide*:

?

- [Fixed Constraint \(Rigid Domain\)](#)
- [Prescribed Displacement/Rotation](#)
- [Applied Force \(Rigid Domain\)](#)
- [Applied Moment \(Rigid Domain\)](#)
- [Mass and Moment of Inertia \(Rigid Domain\)](#)

---

### SKETCH

This section shows a sketch of a spur gear. Some of the inputs are also highlighted in the sketch.

### GEAR PROPERTIES

Specify the properties of a spur gear in this section.

- Select the **Gear mesh** — **External** or **Internal**.
- Enter **Number of teeth  $n$** .

- Enter **Pitch diameter**  $d_p$ .
- Enter **Pressure angle**  $\alpha$ .

### GEAR AXIS

The gear axis is defined as the axis of rotation of the gear. The gear axis is defined as the axis of rotation of the gear. To define the axis, choose **Specify direction** or **Select a parallel edge**.

- For **Specify direction**, enter a value or expression for  $\mathbf{e}_g$ . The default is  $(0, 0, 1)$ . The direction is specified in the selected coordinate system.
- For **Select a parallel edge**, an edge can be selected on the **Gear Axis** subnode, which is added automatically. The vector from the first to last end of the edge is used to define the gear axis. Any edge in the model can be used. Select the **Reverse direction** check box to reverse the direction of the gear axis.

This section is only present in 3D. In 2D, the gear axis is assumed to be the out-of-plane direction.

### DENSITY

The default **Density**  $\rho$  is taken **From material**. In this case, the material assignment for the domain supplies the mass density. For **User defined** enter another value or expression.



The density is needed for dynamic analysis. It is also used when computing mass forces for gravitational or rotating frame loads, and when computing mass properties ([Computing Mass Properties](#)).

### CENTER OF ROTATION

Select a **Center of Rotation** — **Center of mass**, **Centroid of selected entities**, or **User defined**. The center of rotation is the point about which the rotation of the gear is interpreted.

- For **Center of mass**, the center of rotation is taken as the center of mass of the gear.
- For **Centroid of selected entities** select an **Entity level** — **Boundary**, **Edge**, or **Point**. The available choices depend on geometrical dimension. The center of rotation is located at the centroid of the selected entities, which do not need to be related to

rigid domain itself. As a special case, you can select a single point and thus use that point as center of rotation.



When chosen, a default [Center of Rotation: Boundary](#), [Center of Rotation: Edge](#), or [Center of Rotation: Point](#) subnode is automatically added.

- For **User defined**, enter the **Global coordinates of center of rotation**,  $\mathbf{X}_c$ , in the table.

Select the **Offset** check box to add an optional offset vector to the definition of the center of rotation. Enter values for the offset vector  $\mathbf{X}_{\text{offset}}$ .

The center of rotation used is the sum of the vector obtained from any of the input methods and the offset vector.

$$\mathbf{X}_P = \mathbf{X}_{P,\text{input}} + \mathbf{X}_{\text{offset}}$$

#### INITIAL VALUES

Select **From physics interface node** or **Locally defined**.

- When **From physics interface node** is selected, initial values for rigid body displacement, rotation, and velocities are inherited from the physics interface level.
- When **Locally defined** is selected, an **Initial Values** subnode is automatically added, in which you can initialize the rigid domain degrees of freedom.

Select a **Consistent initialization — Default** or **Force initial values** to enforce that the exact values given in this node should be maintained through the consistent initialization process. The default when the given initial values are inconsistent is to make them adapt to each other by adjusting all initial values. For **Force initial values** click to select the applicable check box or boxes — **Translation along first axis**, **Translation along second**

**axis**, **Translation along third axis**, and **Total rotation**. Only the selected degrees of freedom have the initial values fixed during the consistent initialization process..

---

	<ul style="list-style-type: none"><li>• The optional subnode is described in <a href="#">Initial Values (Rigid Domain)</a>.</li><li>• For details about the input expressions at the physics interface level, see <a href="#">Initial Values</a>.</li><li>• See <a href="#">Rigid Domain Model</a> in the <i>Structural Mechanics Module User's Guide</i> for more information.</li><li>• For details about the initialization process, see <a href="#">Initialization in Time-Dependent Studies</a>.</li></ul>
---	---

---



## Spur Gear Theory

---

### LOCATION IN USER INTERFACE

Context Menus

**Multibody Dynamics**>**Gears**>**Spur Gear**

Ribbon

Physics tab with **Multibody Dynamics** selected:

**Domains**>**Gears**>**Spur Gear**

### *Helical Gear*

---

Use the **Helical Gear** node to model a helical gear in 2D and 3D. The gear is assumed to be rigid. Select one or more domains to make them a part of the gear.

A helical gear can be connected to the following gear types:

- Another **Helical Gear**, or a **Spur Gear** through the **Gear Pair** node.
- A **Worm Gear** through the **Worm and Wheel** node.
- A **Helical Rack**, or a **Spur Rack** through the **Rack and Pinion** node.

Any two gears or rack can be connected to each other only if they satisfy the **Gear Pair Compatibility Criteria**.

## GEAR PROPERTIES

Specify the properties of a helical gear in this section.

- Select the **Gear mesh** — **External** or **Internal**.
- Enter **Number of teeth**  $n$ .
- Enter **Pitch diameter**  $d_p$ .
- Enter **Pressure angle**  $\alpha$ .
- Enter **Helix angle**  $\beta$ .



See [Spur Gear](#) for the list of subnodes and the following settings — **Sketch**, **Gear Axis**, **Density**, **Center of Rotation**, and **Initial Values**.



[Helical Gear Theory](#)

## LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics**>**Gears**>**Helical Gear**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Domains**>**Gears**>**Helical Gear**

## Bevel Gear

Use the **Bevel Gear** node to model a bevel gear in 3D. The gear is assumed to be rigid. Select one or more domains to make them a part of the gear.

A bevel gear can be connected to another [Bevel Gear](#) through the [Gear Pair](#) node. Two bevel gears can be connected to each other only if they satisfy the [Gear Pair Compatibility Criteria](#).

## GEAR PROPERTIES

Specify the properties of a bevel gear in this section.

- Enter **Number of teeth**  $n$ .
- Enter **Pitch diameter**  $d_p$ .

- Enter **Pressure angle  $\alpha$ .**
- Enter **Cone angle  $\gamma$ .**



See [Spur Gear](#) for the list of subnodes and the following settings — **Sketch**, **Gear Axis**, **Density**, **Center of Rotation**, and **Initial Values**.



[Bevel Gear Theory](#)

## LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics**>**Gears**>**Bevel Gear**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Domains**>**Gears**>**Bevel Gear**

## *Worm Gear*

Use this node to model a **Worm Gear** in 3D. The gear is assumed to be rigid. Select one or more domains to make them a part of a gear.

A worm gear can be connected to a [Helical Gear](#), or a [Spur Gear](#) through the [Worm and Wheel](#) node. A worm can be connected to a wheel only if they satisfy the [Gear Pair Compatibility Criteria](#).

## GEAR PROPERTIES

Specify the properties of a worm gear in this section.

- Enter **Number of starts  $n$ .**
- Enter **Pitch diameter  $d_p$ .**
- Enter **Pressure angle  $\alpha$ .**

- Enter **Lead angle**  $\phi$ .
- Select the **Handedness** — **Right** or **Left**.



See [Spur Gear](#) for the list of subnodes and the following settings — **Sketch**, **Gear Axis**, **Density**, **Center of Rotation**, and **Initial Values**.



The default value of gear axis for a worm gear is (1, 0, 0).



[Worm Gear Theory](#)

#### LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics**>**Gears**>**Worm Gear**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Domains**>**Gears**>**Worm Gear**

#### *Spur Rack*

Use this node to model a **Spur Rack** in 2D and 3D. The rack is assumed to be rigid. Select one or more domains to make them a part of the rack.

A spur rack can be connected to a [Spur Gear](#), or a [Helical Gear](#) through the [Rack and Pinion](#) node. A rack can be connected to a pinion only if they satisfy the [Gear Pair Compatibility Criteria](#).

#### RACK PROPERTIES

Specify the properties of a rack in this section.

- Enter **Pitch**  $p$ .
- Enter **Pressure angle**  $\alpha$ .

## RACK AXIS

The rack axis is the axis of translation of the rack. To define the axis, choose — **Specify direction**, or **Select a parallel edge**.

- For **Specify direction** enter a value or expression for  $\mathbf{e}_r$ . The default is  $(1, 0, 0)$ . The direction is specified in the selected coordinate system.
- For **Select a parallel edge** an edge can be selected on the **Rack Axis** subnode, which is added automatically. The vector from the first to the last end of the edge is used to define the rack axis. Any edge in the model can be used. Select the **Reverse direction** check box to reverse the direction of the rack axis.



See **Spur Gear** for the list of subnodes and the following settings — **Sketch**, **Density**, **Center of Rotation**, and **Initial Values**.

---



**Spur Rack Theory**

---

## LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics**>**Gears**>**Spur Rack**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Domains**>**Gears**>**Spur Rack**

## *Helical Rack*

---

Use this node to model a **Helical Rack** in 2D and 3D. The rack is assumed to be rigid. Select one or more domains to make them a part of the rack.

A helical rack can be connected to a **Spur Gear**, or a **Helical Gear** through the **Rack and Pinion** node. A rack can be connected to a pinion only if they satisfy the **Gear Pair Compatibility Criteria**.

## RACK PROPERTIES

Specify the properties of a rack in this section.

- Enter **Pitch  $p$** .

- Enter **Pressure angle  $\alpha$ .**
- Enter **Helix angle  $\beta$ .**



See [Spur Rack](#) for the list of subnodes and the following settings — **Sketch**, **Rack Axis**, **Density**, **Center of Rotation**, and **Initial Values**.

---



[Helical Rack Theory](#)

---

#### LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics**>**Gears**>**Helical Rack**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Domains**>**Gears**>**Helical Rack**

#### *Gear Axis*

---

Use the **Gear Axis** node to select any edge in the geometry that is parallel to the axis of rotation of the gear.

This setting is available for the [Spur Gear](#), [Helical Gear](#), [Bevel Gear](#), and [Worm Gear](#) nodes.

#### LOCATION IN USER INTERFACE

This node is automatically added when **Select a parallel edge** is selected in the parent gear node. It cannot be added or removed manually.

#### *Rack Axis*

---

Use the **Rack Axis** node to select any edge in the geometry which is parallel to the axis of translation of the rack.

This setting is available for the [Spur Rack](#) and [Helical Rack](#).

## LOCATION IN USER INTERFACE

This node is automatically added when **Select a parallel edge** is selected in the parent rack node. It cannot be added or removed manually.

### *Gear Pair*

---

Use the **Gear Pair** node for 2D and 3D components to create a connection between gears of the types [Spur Gear](#), [Helical Gear](#), or [Bevel Gear](#) in such a way that, at the contact point, they have no relative motion along the line of action. The remaining displacements and rotations of both the gears are independent of each other.

If the **Line contact** option is chosen in the **Contact model**, one additional constraint is added at the contact point to restrict the relative rotation about a line joining the two gear centers.

When **Friction** is included for a gear pair, friction forces in the plane perpendicular to the line of action are added on both the gears.

### **SKETCH**

This section shows a sketch of a gear pair. The sketch highlights the coordinate system attached to both the gears, the coordinate system attached to the gear tooth, the line of action, and other relevant input parameters.

### **GEAR SELECTION**

Select **Wheel** and **Pinion** from the list of gears. You can select any [Spur Gear](#), [Helical Gear](#), or [Bevel Gear](#) node that is defined for the model.

### **LINE OF ACTION**

Define the line of action by specifying **Obtained by rotation of tangent in — Clockwise direction**, or **Counterclockwise direction**.

Select the **Clockwise direction** option in the following cases:

- **Wheel** is the driver and the **Wheel** has positive rotation about its own axis.
- **Pinion** is the driver and the **Wheel** has negative rotation about its own axis.

Select the **Counterclockwise direction** option in the following cases:

- **Pinion** is the driver and the **Wheel** has positive rotation about its own axis.
- **Wheel** is the driver and the **Wheel** has negative rotation about its own axis.

The gear pair sketch in the **Sketch** section is updated based on the selection and shows the line of action for both cases.

### CONTACT MODEL

Define the contact model for a gear pair in this section.

- Select the **Configuration** — **Parallel or intersecting** or **Neither parallel nor intersecting**. The **Configuration** is always set to **Parallel or intersecting** for spur and bevel gears. For helical gears, you can select the **Configuration** as **Parallel or intersecting** (for parallel axis helical gears) or **Neither parallel nor intersecting** (for crossed axis helical gears).
- For **Parallel or intersecting**, select the **Contact model** — **Point contact** or **Line contact**. For **Neither parallel nor intersecting** the **Contact model** is always set to **Point contact**.
- For **Parallel or intersecting**, enter **Contact point offset from pinion center**  $z_{pn}$ . For **Neither parallel nor intersecting** it is automatically computed.
- For **Line contact**, enter **Working width of pinion**  $w_{pn}$ .



In 2D, there are no settings for selecting the **Contact Model**. The **Configuration** is always **Parallel** and the **Contact model** is always set to the **Point contact**.

### GEAR PAIR PROPERTIES

Define the advanced properties for a gear pair in this section.

- Select **Include gear elasticity** to enter the gear mesh stiffness, mesh damping, and the contact ratio. These properties can be entered in the **Gear Elasticity** subnode, which is added automatically. By default, this option is not selected.
- Select **Include transmission error** to enter the static transmission error for individual gears or for the gear pair. These can be entered in the **Transmission Error** subnode, which is added automatically. By default, this option is not selected.
- Select **Include backlash** to enter backlash in a gear pair. This can be entered in the **Backlash** subnode, which is added automatically. By default, this option is not selected.
- Select **Include friction** to add friction at the contact point. Enter friction properties in the **Friction (Gear)** subnode, which is added automatically. By default, this option is not selected.

## CONTACT FORCE COMPUTATION

Select an option from the list to use the method of computation for the forces at the contact point — **Computed using weak constraints** or **Computed using penalty method**. Alternatively, you can also choose **Do not compute**.

For **Computed using penalty method**, enter **Penalty factor**  $p_c$ . The default value is  $((1[1/ms])^2)*mbd.grp1.me$ . The exact naming of the variable depends on the tags of the nodes in the model tree.



Theory for Gear Pairs

## LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics**>**Gear Pairs**>**Gear Pair**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Global**>**Gear Pairs**>**Gear Pair**

## Worm and Wheel

Use the **Worm and Wheel** node for 3D components to create a connection between a **Worm Gear** and a **Helical Gear** or a **Spur Gear** in such a way that at the contact point they have no relative motion along the line of action. The remaining displacements and rotations of both the gears are independent of each other.

When **Friction** is included on a **Worm and Wheel** node, friction forces in the plane perpendicular to the line of action are added on both the gears.

## GEAR SELECTION

Select **Worm** and **Wheel** from the list of gears. You can select any **Worm Gear** in the **Worm** field and any **Helical Gear** or **Spur Gear** in the **Wheel** field.



See **Gear Pair** for the following settings — **Sketch**, **Line of Action**, **Gear Pair Properties**, and **Contact Force Computation**.

---

	<p>In this node, there are no settings for <b>Contact Model</b>. The <b>Configuration</b> is always <b>Neither parallel nor intersecting</b> and the <b>Contact model</b> is always set to the <b>Point contact</b>.</p>
	<p>In this node, the <b>Include friction</b> option is selected by default, and a <b>Friction</b> subnode is thus added.</p>
	<p><b>Theory for Worm and Wheel</b></p>

---

#### LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics**>**Gear Pairs**>**Worm and Wheel**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Global**>**Gear Pairs**>**Worm and Wheel**

#### *Rack and Pinion*

Use the **Rack and Pinion** node for 2D and 3D components to create a connection between a **Spur Rack** and a **Spur Gear**, or between a **Helical Rack** and a **Helical Gear** in such a way that at the contact point they have no relative motion along the line of action. The remaining displacements and rotations of the rack and pinion are independent of each other.

If the **Line contact** option is chosen in the **Contact model**, one additional constraint is added at the contact point to restrict the relative rotation about a line joining the rack and pinion centers.

When **Friction** is included on a rack and pinion node, friction forces in the plane perpendicular to the line of action are added on the rack and pinion.

## GEAR SELECTION

Select **Rack** and **Pinion** from the list of racks and gears. You can select any [Spur Rack](#), or [Helical Rack](#) in the **Rack** field and any [Spur Gear](#), or [Helical Gear](#) in the **Pinion** field that is defined for the model.

---

	See <a href="#">Gear Pair</a> for the following settings — <b>Sketch</b> , <b>Line of Action</b> , <b>Contact Model</b> , <b>Gear Pair Properties</b> , and <b>Contact Force Computation</b> .
	In this node, there is no setting for <b>Configuration</b> in the <b>Contact Model</b> section since it is possible to have either <b>Point Contact</b> or <b>Line Contact</b> independent of the configuration,
	<a href="#">Theory for Rack and Pinion</a>

---

## LOCATION IN USER INTERFACE

*Context Menus*

**Multibody Dynamics**>**Gear Pairs**>**Rack and Pinion**

*Ribbon*

Physics tab with **Multibody Dynamics** selected:

**Global**>**Gear Pairs**>**Rack and Pinion**

## *Gear Elasticity*

---

The **Gear Elasticity** subnode is automatically added to a [Gear Pair](#), [Worm and Wheel](#), or [Rack and Pinion](#) node when **Include gear elasticity** is selected under **Gear Properties**. In this node, you can enter the elastic properties of a gear pair, such as gear mesh stiffness, mesh damping, and contact ratio.

The contents of the **Gear Elasticity** settings adapt to the type of gear pair where it is used.

## SKETCH

This section shows a sketch of the contact point of gear pair in the gear tooth normal plane. The sketch highlights the line of action along which stiffness and damping are interpreted.

## MESH STIFFNESS

Select the mesh stiffness method **Specify — Tooth stiffness of individual gears** or **Total stiffness of gear pair**.

For **Tooth stiffness of individual gears**, enter the following:

- When the parent node is **Gear Pair**, enter **Wheel tooth stiffness as function of mesh cycle**,  $k_{t,wh}$  and **Pinion tooth stiffness as function of mesh cycle**,  $k_{t,pn}$ . The default expressions are  $(0[N/m]) * mbd.grp1.thm_wh$  and  $(0[N/m]) * mbd.grp1.thm_pn$ , respectively. The exact naming of the variables depends on the tags of the nodes in the model tree.
- When the parent node is **Worm and Wheel**, enter **Worm tooth stiffness as function of mesh cycle**,  $k_{t,wm}$ , and **Wheel tooth stiffness as function of mesh cycle**,  $k_{t,wh}$ . The default expressions are  $(0[N/m]) * mbd.grp1.thm_wm$  and  $(0[N/m]) * mbd.grp1.thm_wh$ , respectively. The exact naming of the variables depends on the tags of the nodes in the model tree.
- When the parent node is **Rack and Pinion**, enter **Rack tooth stiffness as function of mesh cycle**,  $k_{t,rk}$ , and **Pinion tooth stiffness as function of mesh cycle**,  $k_{t,pn}$ . The default expressions are  $(0[N/m^2]) * mbd.grp1.um_rk$  and  $(0[N/m]) * mbd.grp1.thm_pn$ , respectively. The exact naming of the variables depends on the tags of the nodes in the model tree.
- Specify **Contact ratio in mesh cycle** — **Constant** or **Varying**.
- For **Constant** contact ratio, select **Contact ratio** — **1**, **2**, or **3**.
- For **Varying** contact ratio, select **Maximum contact ratio** — **2**, or **3**. Also enter the **Next tooth engagement position in mesh cycle**,  $\zeta$ . The value of  $\zeta$  must vary between 1/2 and 1 when **Maximum contact ratio** is set to **2** and between 1/3 to 1/2 when **Maximum contact ratio** is set to **3**.

For **Total stiffness of gear pair**, enter the following:

- In case the parent node is **Gear Pair**, select the **Total stiffness as function of — Wheel mesh cycle**, **Wheel full revolution**, **Pinion mesh cycle**, or **Pinion full revolution**.
- In case the parent node is **Worm and Wheel**, select the **Total stiffness as function of — Worm mesh cycle**, **Worm full revolution**, **Wheel mesh cycle**, or **Wheel full revolution**.

- In case the parent node is [Rack and Pinion](#), select the **Total stiffness as function of — Rack mesh cycle**, **Pinion mesh cycle**, or **Pinion full revolution**.
- Enter **Total stiffness**,  $k_g$ . The default expression of total stiffness depends on the selection in previous input; for example, it is  $(0[N/m]) * \text{mbd.grp1.thm_wh}$  when **Wheel mesh cycle** is selected. The exact naming of the variable depends on the tags of the nodes in the model tree.

#### MESH DAMPING

Enter **Mesh damping**,  $c_g$ . The mesh damping can be specified as proportional to the mesh stiffness. The default expression is  $(0[s]) * \text{mbd.grp1.kg}$ . The exact naming of the variable depends on the tags of the nodes in the model tree.



Theory for Gear Elasticity

#### LOCATION IN USER INTERFACE

This node is automatically added when **Include gear elasticity** is selected in the parent [Gear Pair](#), [Worm and Wheel](#), and [Rack and Pinion](#) nodes. It cannot be added or removed manually.

#### Transmission Error

The **Transmission Error** subnode is automatically added to a [Gear Pair](#), [Worm and Wheel](#), or [Rack and Pinion](#) node when **Include transmission error** is selected under **Gear Properties**. The static transmission error for individual gears or for a gear pair can be entered in this node.

The contents of the **Transmission Error** settings adapt to the type of gear pair where it is used.

#### SKETCH

This section shows a sketch of the contact point of a gear pair in the gear tooth normal plane. The sketch highlights the line of action along which transmission error is interpreted.

#### TRANSMISSION ERROR

Select the transmission error method **Specify — Transmission error of individual gears**, or **Transmission error of gear pair**.

For **Transmission error of individual gears** enter the followings:

- When the parent node is **Gear Pair**, select the **Wheel transmission error as function of — Mesh cycle**, or **Full revolution** and enter **Wheel transmission error**,  $e_{wh}$ . Similarly, select the **Pinion transmission error as function of — Mesh cycle**, or **Full revolution** and enter **Pinion transmission error**,  $e_{pn}$ .
- When the parent node is **Worm and Wheel**, select the **Worm transmission error as function of — Mesh cycle**, or **Full revolution** and enter **Worm transmission error**,  $e_{wm}$ . Similarly, select the **Wheel transmission error as function of — Mesh cycle**, or **Full revolution** and enter **Wheel transmission error**,  $e_{wh}$ .
- When the parent node is **Rack and Pinion**, enter **Rack transmission error**  $e_{rk}$  as a function of **Mesh cycle**. Similarly select the **Pinion transmission error as function of — Mesh cycle** or **Full revolution** and enter **Pinion transmission error**,  $e_{pn}$ .

The default expression for transmission errors depends on the selection in previous input (for example, the default expression of **Wheel transmission error** is  $(0[m]) * mbd.grp1.thm_wh$  when **Mesh cycle** is selected). The exact naming of the variable depends on the tags of the nodes in the model tree.

For **Transmission error of gear pair** enter the following:

- When the parent node is **Gear Pair**, select the **Transmission error as function of — Wheel mesh cycle**, **Wheel full revolution**, **Pinion mesh cycle**, or **Pinion full revolution**.
- When the parent node is **Worm and Wheel**, select the **Transmission error as function of — Worm mesh cycle**, **Worm full revolution**, **Wheel mesh cycle**, or **Wheel full revolution**.
- When the parent node is **Rack and Pinion**, select the **Transmission error as function of — Rack mesh cycle**, **Pinion mesh cycle**, or **Pinion full revolution**.

Enter **Total transmission error**,  $e_t$ . The default expression of total transmission error depends on the selection in previous input (for example, it is  $(0[m]) * mbd.grp1.thm_wh$  when **Wheel mesh cycle** is selected). The exact naming of the variable depends on the tags of the nodes in the model tree.



## LOCATION IN USER INTERFACE

This node is automatically added when **Include transmission error** is selected in the parent **Gear Pair**, **Worm and Wheel**, and **Rack and Pinion** nodes. It cannot be added or removed manually.

### Backlash

---

The **Backlash** subnode is automatically added to a **Gear Pair**, **Worm and Wheel**, or **Rack and Pinion** node when **Include backlash** is selected under **Gear Properties**.



Backlash can only be specified on a gear pair in a time-dependent analysis.

---

## SKETCH

This section shows a sketch of the contact point of gear pair in the gear tooth normal plane. The sketch highlights the direction in which backlash is interpreted.

## BACKLASH

- Enter **Backlash**  $b_1$ .
- Enter **Penalty factor**  $p_b$ . The default expression is  $((1[1/ms])^2)*mbd.grp1.Ie$ . The exact naming of the variable depends on the tags of the nodes in the model tree.

## ADVANCED

- Select **Manual control on time stepping** to restrict the maximum time step taken by the solver when there is finite penetration.
- Enter **Maximum allowable time step**  $\Delta t_{max}$ . The default expression is  $(\pi/30)*(sqrt(mbd.grp1.Ie/mbd.grp1.pb))$ . The exact naming of the variables depends on the tags of the nodes in the model tree.



Theory for Backlash

---

## LOCATION IN USER INTERFACE

This node is automatically added when **Include backlash** is selected in the parent **Gear Pair**, **Worm and Wheel**, and **Rack and Pinion** nodes. It cannot be added or removed manually.

## Friction (Gear)

---

The **Friction** subnode is automatically added to a [Gear Pair](#), [Worm and Wheel](#), or [Rack and Pinion](#) node when **Include friction** is selected under **Gear Properties**. Use this node to model friction losses on a gear pair.

---



Friction can only be applied to a gear pair in a time-dependent analysis.

---

### FRICITION

- Enter a **Friction coefficient**  $\mu$ .
- Enter a **Characteristic slip velocity**  $v_0$ . The default is `mbd.diag*(1e-3[1/s])`, where the `mbd.diag` variable contains the diagonal of the bounding box of the geometry. The characteristic slip velocity should be small when compared to the actual slip velocity at the contact point. It is the velocity with which the contact surfaces slide, even under sticking conditions.

### NORMAL FORCE

- Specify **Normal force defined through — Contact force** or **User defined force**. For **Contact force** option, it is required that contact force is computed on the parent **Gear Pair** node.
- For the **User defined force** option, enter **Normal force**  $N$ .

### ADVANCED

Enter values or expressions for the:

- **Additional sliding resistance**  $F_{f,r}$ .
- **Maximum friction force**  $F_{f,max}$ . The default is `Inf`, which means that no upper limit is imposed on the friction force.

---



[Theory for Gear Friction](#)

---

### LOCATION IN USER INTERFACE

This node is automatically added when **Include friction** is selected in the parent [Gear Pair](#), [Worm and Wheel](#), and [Rack and Pinion](#) nodes. It cannot be added or removed manually.

## Chain Drive

---

Use the **Chain Drive** (  ) node to model a roller chain sprocket assembly in 2D or 3D. This node determines the interaction of the roller chain sprocket assembly, and automatically generates a set of physics nodes that are necessary to describe its behavior.

The geometry of the roller chain sprocket assembly can be created from the COMSOL **Part Library**. These parts include a set of domain and boundary selections used for the automatic generation of physics nodes. It is also possible to create or import your own geometry. However, in this case, you need to manually create the domain and boundary selections required by the **Chain Drive** node.



See the [Chain Drive](#) and [Chain Geometries](#) sections for more details about chain modeling, geometry creation, and selection settings.

---

### CREATE LINKS AND JOINTS

Click the **Create Links and Joints** button (  ) to automatically generate the physics nodes required to model the chain drive. Clicking this button makes the following changes in the Multibody Dynamics interface:

- Creates a number of **Rigid Domain** nodes collected under a node group named **cdrl: Link (Rigid Domains)**. This node group is only generated if **Link type** is set to **Rigid** or **Rigid with elastic bushing**.



**cdrl** is the default name for the first instance of the **Chain Drive** node and can be changed if needed.

---

- Creates a number of **Attachment** nodes collected under a node group named **cdrl: Link (Attachments)**.
- Creates a number of **Hinge Joint** nodes collected under a node group named **cdrl: Link (Hinge Joints)**. A **Joint Elasticity** subnode is added to all the joints if **Joint type** is set to **Elastic**.
- Creates a node group named **cdrl: Sprocket** containing two **Rigid Domain** nodes, two **Attachment** nodes and two **Hinge Joint** nodes. These attachment and joint nodes are

only generated if the **Include sprocket mounting joints** check box is selected. The rigid domains are only created if **Sprocket type** is set to **Rigid**.

- Creates a node group named **cdrl: Bushing** containing a **Linear Elastic Material** node. Its selection input includes all bushing domains. This node group is only generated if **Link type** is set to **Rigid with elastic bushing**.
- Creates a node group named **cdrl: Contact** containing a **Contact** node to model the contact between the sprockets and the chain links. This node group is only generated if **Contact method** is set to **Mesh based**.
- Creates a **Contact Pair** named **Contact: Sprocket-Roller** under **Definitions**. The source and destination selections of this **Contact Pair** are set automatically. This node is only generated if **Contact method** is set to **Mesh based**.



Only click the **Create Links and Joints** button after setting appropriate values for all parameters in the **Chain Drive** node. The parameters are described in the following sections.



If you change one or more parameters of the **Chain Drive** node after the automatic creation of physics nodes, the settings of the associated physics nodes also require an update. Click the **Create Links and Joints** button again in order to update the settings of the existing physics nodes. A warning message is added under the **Chain Drive** node to notify when such an update is required.



If any selection or related parameters of the **Chain Drive** node are modified after the automatic creation of physics nodes, all associated physics nodes need to be recreated. Click the **Create Links and Joints** button again in order to do this. Since all physics nodes are deleted and created again during this operation, the update may take a while. A warning message is added under the **Chain Drive** node to notify when such an update is required.

#### DOMAIN SELECTION, LINK

Select the domains of all chain links of the roller chain sprocket assembly here. If you use a geometry from the **COMSOL Part Library**, a built-in domain selection named **Links** is automatically selected. This selection is used to create a **Rigid Domain** node

for each link plate. This selection input is only available when **Link type** is set to **Rigid** or **Rigid with elastic bushing**.

#### **DOMAIN SELECTION, SPROCKET**

Select the domains of both sprockets of the roller chain sprocket assembly here. If you use a geometry from the COMSOL **Part Library**, a built-in domain selection named **Sprockets** is automatically selected. This selection is used to create a **Rigid Domain** node for each of the two sprockets. This selection input is only available when **Sprocket type** is set to **Rigid**.

#### **DOMAIN SELECTION, BUSHING**

Select the domains of the bushings of the roller chain sprocket assembly here. Bushings are the components present between the roller plate and the pin plate of a chain link. If you use a geometry from the COMSOL **Part Library**, a built-in domain selection named **Bushings** is automatically selected. This selection is used to create a **Linear Elastic Material** node for the bushing domains. This selection input is only available when **Link type** is set to **Rigid with elastic bushing**.

#### **BOUNDARY SELECTION, PIN**

For 3D models, select the outer cylindrical surfaces of the pin plates here. If you use a geometry from the COMSOL **Part Library**, a built-in boundary selection named **Pin Outer Boundaries** is automatically selected.

For 2D models, select the inner boundaries of the pin plates here. If you use a geometry from the COMSOL **Part Library**, a built-in boundary selection named **Pin Inner Boundaries** is automatically selected.

This selection is used to create an **Attachment** node on the outer boundary of each pin plate. This selection input is always available.

#### **BOUNDARY SELECTION, ROLLER INNER**

For 3D models, select the inner cylindrical surfaces of the roller plates here. For 2D models, select the inner boundaries of the roller plates. If you use a geometry from the COMSOL **Part Library**, a built-in boundary selection named **Roller Inner Boundaries** is automatically selected. This selection is at the same geometrical position as that of the **Pin Outer Boundaries** selection.

This selection is used to create an **Attachment** node on the inner boundaries of each roller plate. This selection input is always available.

A **Hinge Joint** is created for each roller plate and pin plate pair, where the **Attachment** nodes belonging to the pair are set as source and destination of the hinge.

#### **BOUNDARY SELECTION, ROLLER OUTER**

For 3D models, select the outer cylindrical surfaces of the roller plates here. For 2D models, select the outer boundaries of roller plates. This selection is used to model the contact between the chain links and the sprockets. If you use a geometry from the COMSOL **Part Library**, a built-in boundary selection called **Roller Outer Boundaries** is automatically selected.

#### **BOUNDARY SELECTION, SPROCKET OUTER**

Select the outer curved boundaries of the sprockets here. This selection is used to model the contact between the chain links and the sprockets. If you use a geometry from the COMSOL **Part Library**, a built-in boundary selection called **Sprocket Outer Boundaries** is automatically selected.

When **Contact method** is set to **Mesh based**, this selection and the roller outer boundary selection are used to create a **Contact Pair** node between the chain links and the sprockets. If **Contact method** is set to **Roller center based**, these selections are used for a penalty based rigid contact formulation.

#### **BOUNDARY SELECTION, SPROCKET INNER**

Select the inner boundaries of sprockets here. This selection is used to mount each sprocket on a shaft. If you use a geometry from the COMSOL **Part Library**, a built-in boundary selection called **Sprocket Inner Boundaries** is automatically selected.

This selection is used to create an **Attachment** and a **Hinge Joint** node on the inner boundary of each sprocket. It is only available when the **Include sprocket mounting joints** check box is selected.

#### **SELECTION SETTINGS**

For the automatic creation of physics nodes, the **Chain Drive** node requires a set of domain and boundary selections. In the **Chain selections** list, chose whether the selections are to be defined by **From part instance** or **User defined**.

When **Chain selections** is set to **From part instance**, the **Part instance** list contains all roller chain sprocket assemblies present in the **Geometry** node. You can choose any one of them. The selections in the chosen part are automatically identified and taken as the input to the selections required by the **Chain Drive** node. Since the selection inputs are

automatically set, they are in a non-editable state. However, you can edit them by selecting the **Manual control of selections** check box.

When **Chain selections** is set to **User defined**, you have to manually create a set of appropriate domain and boundary selections, and associate each selection with the correct selection input in the **Chain Drive** node.

### CHAIN SETTINGS

Specify the name of chain drive in the **Name** field. By default, the name is `cdr1` for the first instance created; it can, however, be any string. To easily identify the physics nodes created from a specific **Chain Drive** node, the **Name** string is used as an identifier for all node groups and physics nodes it creates.

Chain links can be rigid or elastic. You can also model the effect of elastic bushings between the rigid link plates. Determine the behavior by selecting a **Link type** — **Rigid**, **Rigid with elastic bushing** or **Elastic**.

If **Link type** is set to **Rigid**, a **Rigid Domain** node is automatically created for each link plate. If **Link type** is set to **Rigid with elastic bushing**, a **Linear Elastic Material** node is additionally created for the bushing domains between every roller and pin plate.

Select a **Sprocket type** — **Rigid** or **Elastic**. If **Sprocket type** is set to **Rigid**, a **Rigid Domain** node is automatically created for each sprocket.

If **Link type** or **Sprocket type** is set to **Elastic**, the **Chain Drive** node does not create any specific material model node. For these cases, the default **Linear Elastic Material** node is used for modeling the elastic bodies.

Select a **Contact method** — **Roller center based** or **Mesh based**. The contact method is used to model the contact condition between the chain links and the sprockets. The **Roller center based** contact method is available when **Link type** is **Rigid** or **Rigid with elastic bushing**. If the **Roller center based** contact method is selected, specify a **Penalty factor** ( $p_c$ ) used in the contact condition.

When either the links or the sprockets, or both, are elastic bodies, the **Mesh based** contact method is available. When the **Contact method** is set to **Mesh based**, a **Contact Pair** node is created under **Definitions**, and a **Contact** node with the **Penalty** formulation is created in the physics.

## JOINT SETTINGS

For 3D models, select a **Sprocket Axis** — **From selected part**, **Specify direction**, or **Select a parallel edge**. Relative rotation between chain links is only allowed about the sprocket axis.

- **From selected part** is available only when **Chain selection** is set to **From part instance**. This automatically takes the value of the sprocket axis from the geometry part selected in the **Part instance** input.
- For **Specify direction**, enter a value or expression for  $\mathbf{e}_0$ . The default is  $(0, 0, 1)$ .
- For **Select a parallel edge**, select an edge in the automatically added **Sprocket Axis** subnode. Any edge in the model can be used to specify the sprocket axis. Select the **Reverse direction** check box to reverse the direction of the sprocket axis.

For all above methods, the sprocket axis is also used to define the direction vector  $\mathbf{e}_0$  of each **Hinge Joint** node created by the **Chain Drive** node.

Select an **Attachment type** — **Rigid** or **Flexible**. The selected **Attachment type** determines the **Connection type** in all **Attachment** nodes created by the **Chain Drive** node.

Select a **Joint type** — **Rigid** or **Elastic**. Select an option from the list to determine whether the **Hinge Joint** nodes created by the **Chain Drive** node should be fully rigid or elastic. When **Rigid** is selected, the **Joint Elasticity** input in every **Hinge Joint** node is automatically set to **Rigid joint**. When **Elastic** is selected, a **Joint Elasticity** subnode is added to the **Chain Drive** node. The elastic properties given in this node are used in the **Joint Elasticity** subnode created automatically under every **Hinge Joint** node.

Select the **Include rotational damping** check box to add a **Spring and Damper** subnode with a rotational **Damping coefficient c** to every **Hinge Joint** node.

Select the **Include sprocket mounting joints** check box to automatically create an **Attachment** and a **Hinge Joint** node with a **Fixed** source for each sprocket.

## ADVANCED

Select the **Allow attachments and rigid domains in other joints** check box to announce the **Rigid Domain** and **Attachment** nodes related to the chain link in the list of **Source** and **Destination** in other **Joint** nodes. The attachments related to the sprockets are always announced to other joints.

## LOCATION IN USER INTERFACE

Context Menus

**Multibody Dynamics>Chain Drive**

## Ribbon

Physics tab with **Multibody Dynamics** selected:

**Global>Multibody Dynamics>Chain Drive**

## Sprocket Axis

Use the **Sprocket Axis** node to select any edge in the geometry that is parallel to the axis of the sprocket. This node is available for the [Chain Drive](#) node in 3D.



Refer [Joint Theory](#) for more details.

## LOCATION IN USER INTERFACE

This node is automatically added when **Select a parallel edge** is selected in the parent [Chain Drive](#) node. It cannot be added or removed manually.

## Radial Roller Bearing

Add a **Radial Roller Bearing** node to model a surface of the component supported by a roller bearing. The selected boundaries should form a cylindrical surface representing the surface connected with the inner race of the bearing. The forces on the rollers are computed using the Hertz contact theory. The built-in variable for the roller force distribution has the form `<physicsName>.<RollerBearingTag>.fb[x,y,z]` (for example, `mbd.rrb1.fbx`).

### BEARING ORIENTATION

Use the settings in this section to define the local *y* direction of the bearing.

Subsequent vector and matrix inputs are specified in the local directions specified here.

Enter the **Bearing orientation vector defining the local *y* direction**. The direction given will be adjusted so that it is orthogonal to the rotor axis. The default value is the global *y*-axis.

You can further modify the *y*-axis orientation by entering an angle for the **Rotation of the orientation vector around the bearing axis**. The direction inferred from the previous setting will be rotated counterclockwise around the rotor axis.

## GEOMETRIC PROPERTIES

Select a **Bearing Type** — **Deep groove ball bearing**, **Angular contact ball bearing**, **Self aligning ball bearing**, **Spherical roller bearing**, **Cylindrical roller bearing**, or **Tapered roller bearing**. Then go to the relevant section below to continue defining the properties.

### *Deep Groove Ball Bearing*

- Select either **Single row** or **Double row** to specify if the bearing consists of single or double rows of rollers.
- For a **Double row** bearing, enter the **Axial distance between the roller centers**.
- Enter the remaining geometric properties **Number of balls**; **Ball diameter**; **Radial clearance**; **Pitch diameter**; **Contour radius, inner race**; and **Contour radius, outer race** to define the bearing geometry.

### *Angular Contact Ball Bearing*

- Select either **Single row** or **Double row** to specify if the bearing consists of single or double rows of rollers.
- For a **Double row** bearing, enter the **Axial distance between the roller centers**.
- Enter the remaining geometric properties **Number of balls**; **Ball diameter**; **Radial clearance**; **Pitch diameter**; **Contour radius, inner race**; **Contour radius, outer race**; and **Initial contact angle** to define the bearing geometry.

### *Self Aligning Ball Bearing*

- Select either **Single row** or **Double row** to specify if the bearing consists of single or double rows of rollers.
- Enter the geometric properties **Number of balls**; **Ball diameter**; **Radial clearance**; **Pitch diameter**; **Contour radius, inner race**; and **Contour radius, outer race**.
- For a **Double row** bearing, enter also **Initial contact angle**.

### *Spherical Roller Bearing*

- Select either **Single row** or **Double row** to specify if the bearing consists of single or double rows of rollers.
- For a **Double row** bearing, enter the **Axial distance between the roller centers** and **Initial contact angle**.
- Enter the remaining geometric properties **Number of rollers**; **Roller diameter**; **Radial clearance**; **Pitch diameter**; **Contour radius, inner race**; and **Contour radius, outer race** to define the bearing geometry.

### Cylindrical Roller Bearing

- Select either of **Single row** or **Double row** to specify if the bearing consists of single or double rows of rollers.
- For a **Double row** bearing, enter the **Axial distance between the roller centers**.
- Enter the remaining geometric properties **Number of rollers**, **Effective length of the roller**, **Radial clearance**, and **Pitch diameter** to define the bearing geometry.

### Tapered Roller Bearing

- Select either of **Single row** or **Double row** to specify if the bearing consists of single or double rows of rollers.
- For **Single row** bearing, enter the **Axial offset of roller center**.
- For **Double row** bearing enter the **Axial distance between the roller centers**.
- Enter the remaining geometric properties **Number of rollers**, **Roller diameter**, **Effective length of the roller**, **Inner race clearance**, **Outer race clearance**, **Flange clearance**, **Pitch diameter**, **Cap radius**, **Semi cone angle**, **Roller axis orientation**, **Cap angle**, and **Flange contact angle** to define the bearing geometry.

## MATERIAL PROPERTIES

- Enter the material properties **Young's modulus**, **inner race** and **Poisson's ratio**, **inner race** to define the elastic properties of the inner race in the bearing.
- Select **Outer race material** — **Same as inner race** or **User defined**. If **User defined** is selected, enter the values of **Young's modulus**, **outer race** and **Poisson's ratio**, **outer race**.
- Finally select **Roller material** — **Same as inner race** or **User defined**. If **User defined** is selected, enter the values of the **Young's modulus**, **roller/ball** and **Poisson's ratio**, **roller/ball**.

## FOUNDATION PROPERTIES

Select one of the **Fixed**, **Moving**, or **Flexible** foundation options. In the **Moving** and **Flexible** cases, the subnodes **Moving Foundation** or **Flexible Foundation** are automatically added.



To know the details of contact force computation in different roller bearings, see the theory for [Radial Roller Bearing](#) in the *Rotordynamics Module's User's Guide*.

## LOCATION IN USER INTERFACE

Context Menus

**Multibody Dynamics>Radial Roller Bearing**

Ribbon

Physics tab with **Multibody Dynamics** selected:

**Boundaries>Multibody Dynamics>Radial Roller Bearing**

## *Flexible Foundation*

---

The **Flexible Foundation** node is added automatically when you set **Foundation properties** to **Flexible** on the parent **Radial Roller Bearing** node. In this node, you describe the flexibility of the foundation on which the bearing is mounted.

### FOUNDATION PROPERTIES

Enter the values of the translational and rotational stiffness  $k_u$  and  $k_\theta$  of the foundation. Select the **Translational-rotational coupling** check box to also incorporate the effect of coupled stiffness between the translational and rotational motion of the foundation. If selected, enter the values of the coupling stiffnesses,  $k_{u\theta}$  and  $k_{\theta u}$ .

If you want to perform a dynamic analysis, you can optionally also enter the values of the translational and rotational damping,  $c_u$  and  $c_\theta$ , of the foundation. Select the **Translational-rotational coupling** check box to also incorporate the effect of coupled damping constants between the translational and angular velocity of the foundation. If selected, enter the values of the coupling damping constants  $c_{u\theta}$  and  $c_{\theta u}$ .

## LOCATION IN USER INTERFACE

This node is automatically added when **Foundation Properties** is **Flexible** in the parent bearing node. It cannot be added or removed manually.

## *Moving Foundation*

---

The **Moving Foundation** node is added automatically when you set **Foundation properties** to **Moving** on the parent **Radial Roller Bearing** node. In this node, you prescribe the motion of the foundation on which the bearing is mounted.

### FOUNDATION PROPERTIES

Enter the values of the **Foundation displacement**,  $\mathbf{u}_f$ , and **Foundation rotation**,  $\theta_f$ , at the center of the bearing.

#### **LOCATION IN USER INTERFACE**

This node is automatically added when **Foundation Properties** is **Moving** in the parent bearing node. It cannot be added or removed manually.



## Lumped Mechanical System

This chapter describes the Lumped Mechanical System interface, which is found under the **Structural Mechanics** branch (  ) when adding a physics interface.

In this chapter:

- [The Lumped Mechanical System Interface](#)
- [Theory for the Lumped Mechanical System Interface](#)

# The Lumped Mechanical System Interface

The **Lumped Mechanical System (lms)** interface (  ), found under the **Structural Mechanics** branch (  ) when adding a physics interface, is intended to model displacements and forces in a mechanical system using a circuit representation with lumped components such as masses, springs, and dampers. In addition to the displacement and force sources in the system, terminal nodes are used to define forces, displacements, velocities, accelerations, or impedances as boundary conditions. The physics interface supports stationary, frequency-domain, eigenfrequency and time-domain analysis. The models created with the Lumped Mechanical System interface can also include connections to distributed finite element models.

When the **Lumped Mechanical System** interface is added, one default node is also added to the **Model Builder — Fixed Node** (a boundary condition which acts as “ground”). Then, from the **Physics** toolbar, add more features in the model builder. You can also right-click **Lumped Mechanical System** to select physics features from the context menu.

## SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the name string must be unique. Only letters, numbers, and underscores (`_`) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `lms`.

## SYSTEM SETTINGS

Specify the following in this section:

- **Create unique nodes for new components**
- **Check for hanging nodes before solving**

When **Create unique nodes for new components** is selected, newly added components will be assigned unused node names. The components will be disconnected from the rest of the system and the nodes should be updated to reflect the actual system

connections. When this setting is deselected, new components will be connected to the lowest-numbered nodes, starting from 0.

When **Check for hanging nodes before solving** is selected, a check for hanging (not connected) nodes is performed before solving the equations. An error message is displayed if one or more hanging nodes are present in the system.

### GRAVITY

Select **Include gravity** and specify **Acceleration due to gravity** ( $g$ ) in order to add gravity force on all the mass components. It is possible to exclude the gravity force contribution from a selected mass component at the physics feature level.

### DEPENDENT VARIABLES

Every feature of the physics interface adds certain ODE dependent variables in the form of displacement, force, velocity, and others. All the ODE dependent variables are grouped in the following ODE fields:

- **Displacements** (default field name: `displacements`)
- **Forces** (default field name: `forces`)
- **Velocities** (default field name: `velocities`)
- **Other** (default field name: `other`)

### DISCRETIZATION

All the dependent variables in this physics interface are global and do not need any spatial discretization.

## *Nodes for the Lumped Mechanical System Interface*

---

The **Lumped Mechanical System** has only global nodes. They are available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or by right-clicking then main physics interface node to access the context menu (all users).



Nodes in the lumped mechanical system should not be confused with nodes in the Model Builder tree of the COMSOL Multiphysics software. The system node names are not restricted to numerical values but can contain alphanumeric characters.



In the *COMSOL Multiphysics Reference Manual* see [Table 2-4](#) for links to common sections and [Table 2-5](#) to common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

The following nodes are available from the **Physics** toolbar, **Global** menu or from by right-clicking the **Lumped Mechanical System** interface node.

- Damper
- Displacement Source
- External Source
- Force Source
- Impedance
- Mass
- Spring
- Subsystem Definition
- Subsystem Instance

The following nodes are available from the **Physics** toolbar, **Global** menu or from by selecting from the **Terminals** subnode after right-clicking the **Lumped Mechanical System** interface node.

- Acceleration Node
- Displacement Node
- Fixed Node
- Force Node
- Free Node
- Impedance Node
- Velocity Node

The following nodes are available from the **Physics** toolbar, **Multiphysics Couplings** menu or from by right-clicking the **Multiphysics** node.

- Lumped-Structure Connection

## Mass

The **Mass** (■) feature connects a lumped mass between two nodes in the mechanical system.

### NAME

Enter a component **Name** for the mass. The default name (for the first instance) is M1.

### **NODE CONNECTIONS**

Set the two **Node names** for the connecting nodes for the mass.

### **COMPONENT PARAMETERS**

Specify the value of the **Mass**,  $m$ .

### **INITIAL VALUES**

Enter the values of **Initial displacement** ( $u$ ) and **Initial velocity** ( $\frac{\partial u}{\partial t}$ ). The initial values are used in a time dependent analysis.

### **GRAVITY**

Select the **Exclude contribution** check box in order to exclude the contribution of gravity force.

### **RESULTS**

Select appropriate options in the **Add the following to default results** in order to include the following quantities in the default plots or evaluations:

- **Force**
- **Displacement**
- **Velocity**
- **Acceleration**



Theory for Mass

---

*Context Menus*

**Lumped Mechanical System>Mass**

*Ribbon*

Physics tab with **Lumped Mechanical System** selected:

**Global>Lumped Mechanical System>Mass**

*Spring*

---

The **Spring** (  ) feature connects a linear spring between two nodes in the mechanical system.

## NAME

Enter a component **Name** for the spring. The default name (for the first instance) is K1.

## NODE CONNECTIONS

Set the two **Node names** for the connecting nodes for the spring.

## COMPONENT PARAMETERS

Specify the value of the **Spring Constant**,  $k$ , and the **Predeformation**,  $u_0$ .

## RESULTS

Select appropriate options in the **Add the following to default results** in order to include the following in the default plots or evaluations:

- **Force**
- **Displacement**



## Theory for Spring

---

*Context Menus*

**Lumped Mechanical System>Spring**

*Ribbon*

Physics tab with **Lumped Mechanical System** selected:

**Global>Lumped Mechanical System>Spring**

*Damper*

---

The **Damper** (  ) feature connects a viscous damper between two nodes in the mechanical system.

## NAME

Enter a component **Name** for the damper. The default name (for the first instance) is C1.

## NODE CONNECTIONS

Set the two **Node names** for the connecting nodes for the damper.

## COMPONENT PARAMETERS

Specify the value of the **Damping coefficient**,  $c$ .

## INITIAL VALUES

Enter the values of **Initial displacement**,  $u$ . The initial values are used in a time dependent analysis.

## RESULTS

Select appropriate options in the **Add the following to default results** in order to include the following in the default plots or evaluations:

- **Force**
- **Displacement**
- **Velocity**



Theory for Damper

---

*Context Menus*

**Lumped Mechanical System**>**Damper**

*Ribbon*

Physics tab with **Lumped Mechanical System** selected:

**Global**>**Lumped Mechanical System**>**Damper**

*Impedance*

---

The **Impedance** ( ) feature connects a component, defined through an impedance matrix, between two nodes in the mechanical system.

## NAME

Enter a component **Name** for the impedance. The default name (for the first instance) is  $Z1$ .

## NODE CONNECTIONS

Set the two **Node names** for the connecting nodes for the impedance.

## COMPONENT PARAMETERS

Specify the values of the elements of the **Impedance** matrix, **Z**.

## RESULTS

Select appropriate options in the **Add the following to default results** in order to include the following in the default plots or evaluations:

- **Force**
- **Displacement**
- **Velocity**



[Theory for Impedance](#)

---

*Context Menus*

**Lumped Mechanical System>Impedance**

*Ribbon*

Physics tab with **Lumped Mechanical System** selected:

**Global>Lumped Mechanical System>Impedance**

*Displacement Source*

---

The **Displacement Source** (  ) feature connects a displacement source between two nodes in the mechanical system.

### NAME

Enter a component **Name** for the displacement source. The default name (for the first instance) is U1.

### NODE CONNECTIONS

Set the two **Node names** for the connecting nodes for the displacement source.

## COMPONENT PARAMETERS

Specify the values of **Displacement**,  $u_{src}$ , and **Phase**,  $\phi$ .

## RESULTS

Select appropriate options in the **Add the following to default results** in order to include the following in the default plots or evaluations:

- **Force**
- **Displacement**



Theory for Displacement Source

---

*Context Menus*

**Lumped Mechanical System>Displacement Source**

*Ribbon*

Physics tab with **Lumped Mechanical System** selected:

**Global>Lumped Mechanical System>Displacement Source**

*Force Source*

---

The **Force Source** (–F–) feature connects a force source between two nodes in the mechanical system.

### NAME

Enter a component **Name** for the force source. The default name (for the first instance) is F1.

### NODE CONNECTIONS

Set the two **Node names** for the connecting nodes for the force source.

### COMPONENT PARAMETERS

Specify the values of **Force**,  $f_{src}$ , and **Phase**,  $\phi$ .

## RESULTS

Select appropriate options in the **Add the following to default results** in order to include the following in the default plots or evaluations:

- **Force**
- **Displacement**



### Theory for Force Source

---

*Context Menus*

**Lumped Mechanical System>Force Source**

*Ribbon*

Physics tab with **Lumped Mechanical System** selected:

**Global>Lumped Mechanical System>Force Source**

*External Source*

---

The **External Source** (  ) feature connects an external source between two nodes in the mechanical system.

The equations for the external source can be defined within or outside the physics interface. An external source can be used to connect a distributed model of a component to a lumped model of a system. The displacement of both nodes of the distributed component model can be specified as input to this feature. The reaction forces of both nodes of the lumped system model can be applied to the distributed component model. The **Lumped-Structure Connection** multiphysics coupling can be used to facilitate the connection between a distributed model and a lumped model.

#### NAME

Enter a component **Name** for the external source. The default name (for the first instance) is E1.

#### NODE CONNECTIONS

Set the two **Node names** for the connecting nodes for the external source.

## COMPONENT PARAMETERS

Specify **Input displacement** — **From multiphysics coupling** or **User defined**. If set to **From multiphysics coupling**, the input displacement is taken as the displacement of the connection points of the **Lumped-Structure Connection** multiphysics coupling. If **Input displacement** is set to **User defined**, specify the values of **Displacement, port-1**,  $u_{p10}$ , and **Displacement, port-2**,  $u_{p20}$ .

## RESULTS

Select appropriate options in the **Add the following to default results** in order to include the following in the default plots or evaluations:

- **Force**
- **Displacement**



Theory for External Source

Lumped-Structure Connection

*Context Menus*

**Lumped Mechanical System>External Source**

*Ribbon*

Physics tab with **Lumped Mechanical System** selected:

**Global>Lumped Mechanical System>External Source**

*Subsystem Definition*

The **Subsystem Definition** (  ) feature is used to define subsystems, which can be inserted as components into the main system using a **Subsystem Instance** node. Create the subsystem by adding subnodes to the **Subsystem Definition** node.

It is possible to add all types of two port and single port components within a subsystem definition. It also allows to add subsystem instance within a subsystem definition in order to create nested subsystems.

## NAME

Enter a component **Name** for the subsystem definition. The default name (for the first instance) is **sub1**.

## **NODE CONNECTIONS**

Define the number of connecting nodes by adding more nodes or deleting existing nodes and set the Node names at which the subsystem connects to the main system or to other subsystems. The order in which the **Node names** are defined is the order in which they are referenced by a **Subsystem Instance** node.

The components constituting the subsystem should be connected only to the subsystem's nodes and to themselves.

## **INPUT PARAMETERS**

This section is used to make a parametrized subsystem. This is an optional feature and enabled only with the **Advanced Physics Options** checked. It is possible to add required number of parameters by setting **Parameters** and **Default value**. These parameters are then available to be used within a subsystem and their actual values are provided by the **Subsystem Instance** node while linking this subsystem node to the main system.

*Context Menus*

**Lumped Mechanical System>Subsystem Definition**

*Ribbon*

Physics tab with **Lumped Mechanical System** selected:

**Global>Lumped Mechanical System>Subsystem Definition**

## *Subsystem Instance*

---

The **Subsystem Instance** (  ) feature represents an instance of a subsystem defined by a **Subsystem Definition** feature.

### **NAME**

Enter a component **Name** for the subsystem instance. The default name (for the first instance) is X1.

## **NODE CONNECTIONS**

Select the **Name of subsystem link** from the list of defined subsystems in the model and set the **Node names** at which the subsystem instance connects to the main system or to another subsystem.

## **INPUT PARAMETERS**

Set the values of the parameters in the **Expression** column to be used by a subsystem. This is an optional feature and enabled only with the **Advanced Physics Options** checked.

The parameters are shown in the list only when a subsystem chosen in the **Name of the subsystem link** defines any parameter. The values of parameters provided at this node override the default values of the parameters specified at the **Subsystem Definition** node.

*Context Menus*

**Lumped Mechanical System>Subsystem Instance**

*Ribbon*

Physics tab with **Lumped Mechanical System** selected:

**Global>Lumped Mechanical System>Subsystem Instance**

*Fixed Node*

The **Fixed Node** (  ) feature adds a fixed node to the system. This is the default node in the Lumped Mechanical System interface. This node is used in order to fix one or more ports of a component.

---



- Disable the **Fixed Node** if none of the nodes of the system is connected to the ground.
- It is also possible to use a single instance of this node to fix the ports of multiple components.

---

#### **NAME**

Enter a component **Name** for the fixed node. The default name (for the first instance) is **fix1**.

#### **NODE CONNECTIONS**

Set the **Node name** for the fixed node.



#### **LOCATION IN USER INTERFACE**

*Context Menus*

**Lumped Mechanical System>Terminals>Fixed Node**

### *Ribbon*

Physics tab with **Lumped Mechanical System** selected:

**Global>Terminals>Fixed Node**

### *Free Node*

---

The **Free Node** (⊕) feature adds a free node to the system. This node is used in order to make one or more ports of a component free.

#### **NAME**

Enter a component **Name** for the free node. The default name (for the first instance) is `fr1`.

#### **NODE CONNECTIONS**

Set the **Node name** for the free node.



Theory for Free node

---

### **LOCATION IN USER INTERFACE**

#### *Context Menus*

**Lumped Mechanical System>Terminals>Free Node**

### *Ribbon*

Physics tab with **Lumped Mechanical System** selected:

**Global>Terminals>Free Node**

### *Displacement Node*

---

The **Displacement Node** (⊖) feature adds a displacement node to the system. This node is used in order to prescribe the displacement of one or more ports of a component. Using the **Lumped-Structure Connection** multiphysics coupling, a displacement node can be used to connect a distributed model of a component to the lumped model of a system. The displacement of the node of the distributed component model can be specified as the input to this feature. The reaction force of the node of the lumped system model can be applied to the distributed component model.

## NAME

Enter a component **Name** for the displacement node. The default name (for the first instance) is **disp1**.

## NODE CONNECTIONS

Set the **Node name** for the displacement node.

## TERMINAL PARAMETERS

Specify **Input displacement** — **From multiphysics coupling** or **User defined**. If set to **From multiphysics coupling**, the input displacement is taken as the displacement of the connection point of the **Lumped-Structure Connection** multiphysics coupling. In this case, phase is set as 0. If **Input displacement** is set to **User defined**, specify the values of **Displacement**,  $u_{p10}$ , and **Phase**,  $\phi$ .



Theory for Displacement Node

Lumped-Structure Connection

## LOCATION IN USER INTERFACE

*Context Menus*

**Lumped Mechanical System**>**Terminals**>**Displacement Node**

*Ribbon*

Physics tab with **Lumped Mechanical System** selected:

**Global**>**Terminals**>**Displacement Node**

## Velocity Node

The **Velocity Node** (掣) feature adds a velocity node to the system. This node is used in order to prescribe the velocity of one or more ports of a component.

## NAME

Enter a component **Name** for the velocity node. The default name (for the first instance) is **vel1**.

## NODE CONNECTIONS

Set the **Node name** for the velocity node.

## TERMINAL PARAMETERS

Specify the value of **Velocity**,  $v_{p1}$ .

Also select **Displacement in stationary study** — **Free** or **Constrained**.

---



Theory for Velocity Node

## LOCATION IN USER INTERFACE

*Context Menus*

**Lumped Mechanical System**>**Terminals**>**Velocity Node**

*Ribbon*

Physics tab with **Lumped Mechanical System** selected:

**Global**>**Terminals**>**Velocity Node**

## Acceleration Node

---

The **Acceleration Node** (H) feature adds an acceleration node to the system. This node is used in order to prescribe the acceleration of one or more ports of a component.

## NAME

Enter a component **Name** for the acceleration node. The default name (for the first instance) is `acc1`.

## NODE CONNECTIONS

Set the **Node name** for the acceleration node.

## TERMINAL PARAMETERS

Specify the value of **Acceleration**,  $a_{p1}$ .

Also select **Displacement in stationary study** — **Free** or **Constrained**.

---



Theory for Acceleration Node

## LOCATION IN USER INTERFACE

*Context Menus*

**Lumped Mechanical System>Terminals>Acceleration Node**

*Ribbon*

Physics tab with **Lumped Mechanical System** selected:

**Global>Terminals>Acceleration Node**

## *Force Node*

---

The **Force Node** (  ) feature adds a force node to the system. This node is used in order to apply the force at one or more ports of a component.

### **NAME**

Enter a component **Name** for the force node. The default name (for the first instance) is **frc1**.

### **NODE CONNECTIONS**

Set the **Node name** for the force node.

### **TERMINAL PARAMETERS**

Specify the values of **Force**,  $f_{p10}$ , and **Phase**,  $\phi$ .



Theory for Force Node

---

## LOCATION IN USER INTERFACE

*Context Menus*

**Lumped Mechanical System>Terminals>Force Node**

*Ribbon*

Physics tab with **Lumped Mechanical System** selected:

**Global>Terminals>Force Node**

## *Impedance Node*

---

The **Impedance Node** (  ) feature adds an impedance node to the system. This node is used in order to define the impedance of one or more ports of a component.

### **NAME**

Enter a component **Name** for the impedance node. The default name (for the first instance) is `imp1`.

### **NODE CONNECTIONS**

Set the **Node name** for the impedance node.

### **TERMINAL PARAMETERS**

Specify the value of **Impedance, Z**.



[Theory for Impedance Node](#)

---

### **LOCATION IN USER INTERFACE**

*Context Menus*

**Lumped Mechanical System**>**Terminals**>**Impedance Node**

*Ribbon*

Physics tab with **Lumped Mechanical System** selected:

**Global**>**Terminals**>**Impedance Node**

---

### *Lumped-Structure Connection*

It is possible to couple the Lumped Mechanical System interface to several other structural mechanics interfaces through a predefined multiphysics coupling. Add a **Lumped-Structure Connection** multiphysics node(  ) to connect finite element models built using structural mechanics interfaces, to the Lumped Mechanical System interface.

The displacements obtained from the finite element model at the connection points are transferred to the lumped model. Similarly, the reaction forces obtained from the lumped model are applied to the finite element model.

## NAME

Enter a **Name** for the lumped-structure connection. The default name (for the first instance) is **1sc1**.

## COUPLED INTERFACES

Select **Lumped mechanical system** — **Lumped Mechanical System**.

Select **Structure** — **Solid Mechanics**, **Multibody Dynamics**, **Shell**, **Layered Shell**, **Membrane**, **Plate** (2D), **Beam**, **Truss**, **Solid Rotor**, **Beam Rotor** or **Pipe Mechanics**.

## CONNECTION SETTINGS

**Connect to** — Set to **External Source** or **Displacement Node** depending upon whether an **External Source** or a **Displacement Node** node is used to connect the lumped system to the finite element model. If you select **External Source**, corresponding nodes of the selected **Lumped Mechanical System** are listed. If you select **Displacement Node**, corresponding nodes of the selected **Lumped Mechanical System** are listed. Note that for an **External Source** or **Displacement Node** to be listed in the **Lumped-Structure Connection** multiphysics coupling node, you need to set the **Input displacement** of the respective node to **From multiphysics coupling**.

When **Connect to** is set to **External Source**, the displacements of two connection points are required to connect the selected **Lumped mechanical system** and **Structure**. Select them in the **Point Selection, Port-1** and **Point Selection, Port-2** sections, respectively. The displacements of these points are taken as the **Input displacement** of the ports of the selected **External Source** node. If multiple points are selected in these sections, the **Input displacement** of each port is the average displacement of all points in the respective selections.

When **Connect to** is set to **Displacement Node**, the displacement of a single connection point is required. This is selected in the **Point Selection, Terminal** section. The displacement of this point is taken as the **Input displacement** of the terminal of the selected displacement node. In this case, **phase** of the displacement node is set to 0. If multiple points are selected in this section, the **Input displacement** is the average of all points in the selection.

Select the **Direction** — **X-axis**, **Y-axis**, **Z-axis** (3D) or **User defined** to specify the connection point displacement component that will be used in the **Displacement Node** or **External Source** nodes. In 2D axisymmetry, the allowed values are **R-axis**, **Z-axis** or **User defined**. For **User defined**, enter the direction vector of the connection point displacement.

## LOCATION IN USER INTERFACE

*Context Menus*

**Multiphysics>Lumped-Structure Connection**

*Ribbon*

Physics tab with **Lumped Mechanical System** selected:

**Multiphysics Couplings>Lumped-Structure Connection**

# Theory for the Lumped Mechanical System Interface

This section introduces you to the theory of various components of the Lumped Mechanical System interface.

In this section:

- [Dependent Variables](#)
- [Theory for Two-Port Components](#)
- [Theory for Single-Port Components](#)

## *Dependent Variables*

---

Every feature of the physics interface adds certain ODE dependent variables in the form of displacement, force, velocity, and others. All the ODE dependent variables are grouped in the following ODE fields:

- Displacements
- Forces
- Velocities
- Other

## *Theory for Two-Port Components*

---

The equations for all the two port components are described below:

- [Theory for Mass](#)
- [Theory for Spring](#)
- [Theory for Damper](#)
- [Theory for Impedance](#)
- [Theory for Displacement Source](#)
- [Theory for Force Source](#)
- [Theory for External Source](#)

## THEORY FOR MASS

A mass component creates a difference in the forces between the two connecting nodes.

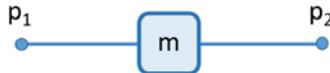


Figure 5-1: A mass component connected between the two nodes  $p_1$  and  $p_2$ .

The following equations are used to relate force ( $f_{p1}$ ) and displacement ( $u_{p1}$ ) of one node to the force ( $f_{p2}$ ) and displacement ( $u_{p2}$ ) of another node:

$$\begin{aligned}f_{p1} + f_{p2} &= f \\u_{p2} - u_{p1} &= 0 \\u_{p2} &= u\end{aligned}$$

where  $f$  and  $u$  are the force and displacement variables in the component. They are related by the following equations, where  $m$  is the component mass:

*Frequency domain and eigenfrequency studies*

$$f = -(i\omega)^2 mu$$

*Time dependent studies*

$$f = -m \frac{\partial^2 u}{\partial t^2}$$

*Stationary studies*

$$f = 0$$

An additional gravity force is added in the component force when the gravity contribution is included:

$$f = -mg$$

## THEORY FOR SPRING

A spring component creates a difference in the displacements between the two connecting nodes which is proportional to the force.

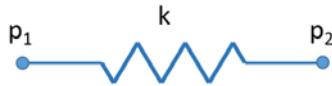


Figure 5-2: A spring component connected between the two nodes  $p_1$  and  $p_2$ .

The following equations are used to relate force ( $f_{p1}$ ) and displacement ( $u_{p1}$ ) of one node to the force ( $f_{p2}$ ) and displacement ( $u_{p2}$ ) of another node:

$$\begin{aligned} f_{p1} + f_{p2} &= 0 \\ u_{p2} - u_{p1} &= u \\ f_{p2} &= f \end{aligned}$$

where  $f$  and  $u$  are the force and displacement variables in the component. They are related by the following equation:

$$f = -k(u - u_0)$$

where  $k$  is the spring constant and  $u_0$  is the predeformation.

#### THEORY FOR DAMPER

A damper component is similar to the spring component and creates a difference in the displacements of the two connecting nodes. The relative velocity is proportional to the force.

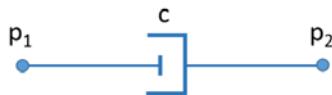


Figure 5-3: A damper component connected between the two nodes  $p_1$  and  $p_2$ .

The following equations are used to relate force ( $f_{p1}$ ) and displacement ( $u_{p1}$ ) of one node to the force ( $f_{p2}$ ) and displacement ( $u_{p2}$ ) of another node:

$$\begin{aligned} f_{p1} + f_{p2} &= 0 \\ u_{p2} - u_{p1} &= u \\ f_{p2} &= f \end{aligned}$$

where  $f$  and  $u$  are the force and displacement variables in the component. They are related by the following equations, where  $c$  is the damping coefficient:

*Frequency domain and eigenfrequency studies*

$$f = -(i\omega)cu$$

*Time dependent studies*

$$f = -c\frac{\partial u}{\partial t}$$

*Stationary studies*

$$f = 0$$

### THEORY FOR IMPEDANCE

An impedance component creates a difference in the displacements as well as forces between the two connecting nodes.

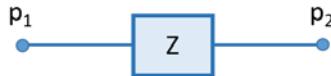


Figure 5-4: An impedance component connected between the two nodes  $p_1$  and  $p_2$ .

The following equations are used to relate force ( $f_{p1}$ ) and displacement ( $u_{p1}$ ) of one node to the force ( $f_{p2}$ ) and displacement ( $u_{p2}$ ) of another node:

$$f_{p1} = f_1$$

$$u_{p1} = u_1$$

$$f_{p2} = f_2$$

$$u_{p2} = u_2$$

where  $f_1, f_2$  and  $u_1, u_2$  are the force and displacement variables in the component. They are related by the following equations:

*Frequency domain and eigenfrequency studies*

$$f_1 = -(i\omega)(Z_{11}u_1 + Z_{12}u_2)$$

$$f_2 = -(i\omega)(Z_{21}u_1 + Z_{22}u_2)$$

$$f_1 = f_2$$

$$u_1 = u_2$$

Here  $Z_{11}$ ,  $Z_{12}$ ,  $Z_{21}$ ,  $Z_{22}$  are the components of the impedance matrix.

### THEORY FOR DISPLACEMENT SOURCE

A displacement source creates a difference in the displacements of the two connecting nodes.

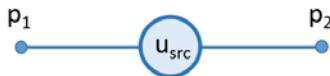


Figure 5-5: A displacement source connected between the two nodes  $p_1$  and  $p_2$ .

The following equations are used to relate force ( $f_{p1}$ ) and displacement ( $u_{p1}$ ) of one node to the force ( $f_{p2}$ ) and displacement ( $u_{p2}$ ) of another node:

$$\begin{aligned} f_{p1} + f_{p2} &= 0 \\ u_{p2} - u_{p1} &= u \\ f_{p2} &= f \end{aligned}$$

where  $f$  and  $u$  are the force and displacement variables in the component. The following equation prescribes the component displacement,  $u$ :

Frequency domain and eigenfrequency studies

$$u = u_{src} e^{i\phi}$$

Time dependent and stationary studies

$$u = u_{src}$$

Here  $u_{src}$  is the prescribed displacement value and  $\phi$  is the phase angle.

### THEORY FOR FORCE SOURCE

A force source creates a difference in the forces of the two connecting nodes.

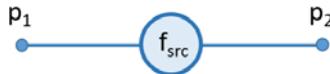


Figure 5-6: A force source connected between the two nodes  $p_1$  and  $p_2$ .

The following equations are used to relate force ( $f_{p1}$ ) and displacement ( $u_{p1}$ ) of one node to the force ( $f_{p2}$ ) and displacement ( $u_{p2}$ ) of another node:

$$\begin{aligned} f_{p1} + f_{p2} &= f \\ u_{p2} - u_{p1} &= 0 \\ u_{p2} &= u \end{aligned}$$

where  $f$  and  $u$  are the force and displacement variables in the component. The following equation prescribes the component force,  $f$ :

Frequency domain and eigenfrequency studies

$$f = f_{\text{src}} e^{i\phi}$$

Time dependent and stationary studies

$$f = f_{\text{src}}$$

Here,  $f_{\text{src}}$  is the prescribed force value and  $\phi$  is the phase angle.

#### THEORY FOR EXTERNAL SOURCE

An external source is similar to the impedance component and creates a difference in the displacements as well as forces of the two connecting nodes.



Figure 5-7: An external source connected between the two nodes  $p_1$  and  $p_2$ .

The following equations are used to relate force ( $f_{p1}$ ) and displacement ( $u_{p1}$ ) of one node to the force ( $f_{p2}$ ) and displacement ( $u_{p2}$ ) of another node:

$$\begin{aligned}f_{p1} &= f_1 \\u_{p1} &= u_1 \\f_{p2} &= f_2 \\u_{p2} &= u_2\end{aligned}$$

where  $f_1, f_2$  and  $u_1, u_2$  are the force and displacement variables in the component. The following equation prescribes the component displacement ( $u_1, u_2$ )

$$\begin{aligned}u_1 &= u_{p10} \\u_2 &= u_{p20}\end{aligned}$$

where  $u_{p10}, u_{p20}$  are the displacement values obtained from the component level model.

The component force ( $f_1, f_2$ ) obtained from the external source node of the system level model can be applied in the component level model as a feedback in order to complete the coupling between the two.

### *Theory for Single-Port Components*

---

The equations for all the single-port components, which are also known as the terminals, are described below:

- [Theory for Fixed Node](#)
- [Theory for Displacement Node](#)
- [Theory for Velocity Node](#)
- [Theory for Acceleration Node](#)
- [Theory for Free node](#)
- [Theory for Force Node](#)
- [Theory for Impedance Node](#)

#### **THEORY FOR FIXED NODE**



Figure 5-8: A fixed node connected at node  $p1$ .

The following equation prescribes the node displacement,  $u_{p1}$ :

$$u_{p1} = 0$$

#### THEORY FOR DISPLACEMENT NODE

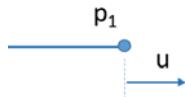


Figure 5-9: A displacement node connected at node  $p_1$ .

The following equation prescribes the node displacement,  $u_{p1}$ :

Frequency domain and eigenfrequency studies

$$u_{p1} = u_{p10} e^{i\phi}$$

Time dependent and stationary studies

$$u_{p1} = u_{p10}$$

Here  $u_{p10}$  is the prescribed displacement value and  $\phi$  is the phase angle.

#### THEORY FOR VELOCITY NODE

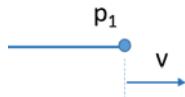


Figure 5-10: A velocity node connected at node  $p_1$ .

The following equation prescribes the node displacement,  $u_{p1}$ , given that  $v_{p1}$  is the prescribed velocity value:

Frequency domain and eigenfrequency studies

$$(i\omega)u_{p1} = v_{p1}$$

For time dependent studies

$$\frac{\partial u_{p1}}{\partial t} = v_{p1}$$

### Stationary studies

If the displacement is set to free:

$$f_{p1} = 0$$

If the displacement is set to constrained:

$$u_{p1} = 0$$

## THEORY FOR ACCELERATION NODE

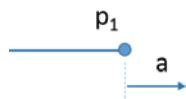


Figure 5-11: An acceleration node connected at node  $p1$ .

The following equation prescribes the node displacement,  $u_{p1}$ , given that  $a_{p1}$  is the prescribed acceleration value:

Frequency domain and eigenfrequency studies

$$(i\omega)^2 u_{p1} = a_{p1}$$

Time dependent studies

$$\frac{\partial^2 u_{p1}}{\partial t^2} = a_{p1}$$

### Stationary studies

If the displacement is set to free:

$$f_{p1} = 0$$

If the displacement is set to constrained:

$$u_{p1} = 0$$

### THEORY FOR FREE NODE

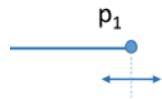


Figure 5-12: A free node connected at node  $p_1$ .

The following equation prescribes the nodal force,  $f_{p1}$ :

$$f_{p1} = 0$$

### THEORY FOR FORCE NODE

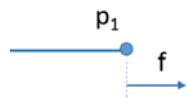


Figure 5-13: A force node connected at node  $p_1$ .

The following equation prescribes the nodal force,  $f_{p1}$ :

Frequency domain and eigenfrequency studies

$$f_{p1} = f_{p10}e^{i\phi}$$

Time dependent and stationary studies

$$f_{p1} = f_{p10}$$

Here  $f_{p10}$  is the prescribed force value and  $\phi$  is the phase angle.

### THEORY FOR IMPEDANCE NODE

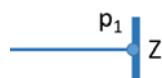


Figure 5-14: An impedance node connected at node  $p_1$ .

The following equation relates the nodal force ( $f_{p1}$ ) and node displacement ( $u_{p1}$ ):

*Frequency domain and eigenfrequency studies*

$$f_{p1} = -(i\omega)Zu_{p1}$$

*Time dependent and stationary studies*

$$f_{p1} = 0$$

Here  $Z$  is the prescribed impedance value.



# Index

- A** acceleration node (node) 304–305
- Application Libraries window 15
- applied force and moment (node) 242
  - theory 31, 138
- attachment (node) 211
- attachments
  - fixed 25
  - source and destination 25
  - theory 46
- axial rotation
  - constraints, theory 130
  - locking, theory 133
  - springs and dampers, theory 135
- B** backlash 55
- backlash (node) 275
  - theory 184
- ball joint (node) 222
  - theory 41, 115
- base motion (node) 250
- bevel gear 65
- bevel gear (node) 262
  - theory 158
- boundary nodes 204
- C** cam-follower (node) 255
- Cardan joint 44, 229
  - theory 44, 122
- center of joint 28
  - boundary (node) 247
  - edge (node) 247
  - point (node) 247
- center of rotation
  - attachments 47
  - boundary (node) 215
  - edge (node) 216
  - point (node) 216
- chain drive (node) 277
- clearance joint (node) 225
  - theory 43, 119
- constraints (node) 232
  - theory 28, 125
- contact area (node) 246
- coordinate systems
  - attachments 27
  - ball joints 115
  - cylindrical joints 113
  - hinge joints 110
  - joints 28, 106
  - reduced slot joints 119
  - slot joints 117
- Coulomb's law 31
- cylindrical joint (node) 220
  - theory 37, 112
- D** damper (node) 294
- deactivation condition 137
- degrees of freedom 25
  - for different joints 26
  - initial values 33
- destination attachment 25
- destination filter (node) 249
- destination point
  - boundary (node) 248
  - edge (node) 248
  - point (node) 248
- displacement node (node) 302
- displacement source (node) 296
- displacement vector 108
- distance joint (node) 227
  - theory 44, 121
- documentation 14
- domain nodes 204

- E**
  - edge nodes 204
  - effective mass 127
  - emailing COMSOL 16
  - external source (node) 298
- F**
  - fixed attachment 25
  - fixed joint (node) 227
    - theory 43, 121
  - fixed node (node) 301
  - flexible foundation (node) 286
  - fluid-multibody interaction 97, 99
  - force source (node) 297
  - free node (node) 302
  - friction 56
  - friction (node) 243
    - theory 31, 139
  - friction- gear (node)
    - theory 186
  - frictional coefficient 186
  - friction-gear (node) 276
  - friction-rigid body contact (node) 254
    - theory 194
- G**
  - gear axis (node) 266
  - gear elasticity (node) 271
    - theory 176
  - gear geometries 57
  - gear pair (node) 267
    - theory 163
  - gear tooth 58
  - geometric nonlinearity 95
- H**
  - helical gear 63
  - helical gear (node) 261
    - theory 156
  - helical rack 68
  - helical rack (node) 265
    - theory 162
  - hinge joint (node) 219
    - theory 37, 110
- I**
  - Hooke's joint 44, 229
  - theory 44, 122
- J**
  - impedance (node) 295
  - impedance node (node) 306
  - inclination angle
    - constraints, theory 129
    - locking, theory 133
    - springs and dampers, theory 135
  - initial values (node) 208
    - rigid domains 216
  - initial values, joints 33
  - initially rigid (node) 210
  - input shaft orientation (node) 246
  - internet resources 13
- K**
  - knowledge base, COMSOL 17
- L**
  - locking (node) 235
    - theory 29, 131
  - lumped mechanical system interface 290
  - lumped-structure connection (node) 306
- M**
  - mass (node) 292
  - moving foundation (node) 286
  - MPH-files 15
  - multibody dynamics interface 198
  - multiphysics coupling
    - fluid-multibody interaction 97, 99
- O**
  - output shaft orientation (node) 247
- P**
  - pair nodes 204
  - penalty factor 126
  - penalty method 148
  - planar joint (node) 222

- theory 40, 114
- plane stress and strain 199
- point nodes 204
- prescribed motion (node) 240
  - theory 30, 136
- prismatic joint (node) 217
  - theory 36, 106
- Q** quaternion representation, of rigid connector 104
  - quaternion vector 108
- R** rack and pinion (node) 270
  - theory 173
- rack axis (node) 266
- rack tooth 59
- radial roller bearing (node)
  - solid rotor 283
- reduced slot joint (node) 224
  - theory 42, 118
- refpnt variable 200
- regularization factor 32, 186
- revolute joint 37, 110
- rigid body contact (node) 251
  - theory 189
- rigid domain (node) 213
- rotational
  - constraints, theory 128
  - locking, theory 132
  - prescribed motion, theory 137
  - springs and dampers, theory 135
- S** screw joint (node) 221
  - theory 39, 113
- sliding joint 36, 106
- slot joint (node) 223
  - theory 41, 116
- source attachment 25
- source filter (node) 249
- source point
- boundary (node) 247
- edge (node) 248
- point (node) 248
- spring (node) 293
- spring and damper (node) 237
  - theory 29, 134
- spring-damper (node)
  - theory 50
- sprocket axis (node) 283
- spur gear 60
- spur gear (node) 258
  - theory 154
- spur rack 61, 67
- spur rack (node) 264
  - theory 160
- study steps, geometric nonlinearity and 95
- subsystem definition (node) 299
- subsystem instance (node) 300
- T** technical support, COMSOL 16
- thickness 199
- translational
  - constraints, theory 126
  - locking, theory 131
  - prescribed motion, theory 136
  - springs and dampers, theory 134
- translational joint 36, 106
- transmission error (node) 273
  - theory 183
- U** universal joint
  - theory 44, 122
- universal joint (node) 229
- V** variables
  - refpnt 200
  - velocity node (node) 303
- W** weak constraints 33, 147
- websites, COMSOL 17

worm and wheel (node) 269

theory 171

worm gear 66

worm gear (node) 263

theory 159