



# Polymer Flow Module

User's Guide

# Polymer Flow Module User's Guide

© 1998–2023 COMSOL

Protected by patents listed on [www.comsol.com/patents](http://www.comsol.com/patents), or see Help>About COMSOL Multiphysics on the File menu in the COMSOL Desktop for less detailed lists of U.S. Patents that may apply. Patents pending.

This Documentation and the Programs described herein are furnished under the COMSOL Software License Agreement ([www.comsol.com/sla](http://www.comsol.com/sla)) 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 6.2

## Contact Information

Visit the Contact COMSOL page at [www.comsol.com/contact](http://www.comsol.com/contact) to submit general inquiries 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 on 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/product-update](http://www.comsol.com/product-update)
- COMSOL Blog: [www.comsol.com/blogs](http://www.comsol.com/blogs)
- Discussion Forum: [www.comsol.com/forum](http://www.comsol.com/forum)
- Events: [www.comsol.com/events](http://www.comsol.com/events)
- COMSOL Video Gallery: [www.comsol.com/videos](http://www.comsol.com/videos)
- Support Knowledge Base: [www.comsol.com/support/knowledgebase](http://www.comsol.com/support/knowledgebase)

Part number: CM025301

C o n t e n t s

Chapter 1: Introduction

<b>About the Polymer Flow Module</b>	<b>16</b>
The Polymer Flow Module . . . . .	16
The Polymer Flow Module Physics Interface Guide . . . . .	16
Common Physics Interface and Feature Settings and Nodes . . . . .	20
Where Do I Access the Documentation and Application Libraries? . . . .	20
 <b>Overview of the User's Guide</b>	 <b>24</b>

Chapter 2: Modeling Polymer Flow

<b>Newtonian and Non-Newtonian Fluids</b>	<b>28</b>
References . . . . .	30
 <b>The Polymer Flow Physics Interfaces</b>	 <b>31</b>
Selecting the Right Physics Interface. . . . .	31
Single-Phase Flow . . . . .	32
Porous Media Flow . . . . .	34
Multiphase Flow . . . . .	34
Fluid-Structure Interaction . . . . .	34
Chemical Species Transport. . . . .	35
Heat Transfer and Nonisothermal Flow . . . . .	35
Coupling to Other Physics Interfaces . . . . .	35

Chapter 3: Single-Phase Flow Interfaces

<b>The Creeping Flow and Laminar Flow Interfaces</b>	<b>38</b>
The Creeping Flow Interface . . . . .	38
The Laminar Flow Interface . . . . .	39
Domain, Boundary, Pair, and Point Nodes for Single-Phase Flow . . . .	45

Fluid Properties . . . . .	46
Volume Force . . . . .	52
Initial Values . . . . .	52
Wall . . . . .	53
Inlet. . . . .	57
Outlet . . . . .	60
Symmetry . . . . .	63
Open Boundary . . . . .	64
Boundary Stress . . . . .	65
Periodic Flow Condition . . . . .	66
Interior Wall . . . . .	68
Porous Interface . . . . .	70
Flow Continuity . . . . .	71
Pressure Point Constraint . . . . .	72
Point Mass Source . . . . .	72
Line Mass Source. . . . .	73
Gravity . . . . .	74
<b>The Potential Flow Interface</b>	<b>75</b>
The Incompressible Potential Flow Interface . . . . .	75
Domain, Boundary, Point, and Pair Nodes for the Incompressible	
Potential Flow Interface . . . . .	77
Fluid Properties . . . . .	77
Initial Values . . . . .	78
Wall . . . . .	78
Velocity . . . . .	78
Open Boundary . . . . .	78
Symmetry . . . . .	79
Potential Point Constraint . . . . .	79
Periodic Flow Condition . . . . .	79
Interior Wall . . . . .	81
Flow Continuity . . . . .	81
<b>The Rotating Machinery, Laminar Flow Interface</b>	<b>83</b>
Moving Mesh . . . . .	83
The Rotating Machinery, Laminar Flow Interface . . . . .	84
Domain, Boundary, Point, and Pair Nodes for the Rotating	
Machinery Interfaces . . . . .	86

Stationary Free Surface . . . . .	86
Contact Angle . . . . .	88
<b>The Viscoelastic Flow Interface</b>	<b>89</b>
Domain, Boundary, Point, and Pair Nodes for the Viscoelastic Interface. . . . .	90
Fluid Properties . . . . .	91
<b>Theory for the Single-Phase Flow Interfaces</b>	<b>94</b>
General Single-Phase Flow Theory . . . . .	95
Compressible Flow . . . . .	97
Weakly Compressible Flow . . . . .	98
The Mach Number Limit . . . . .	98
Incompressible Flow . . . . .	99
The Reynolds Number. . . . .	99
Non-Newtonian Flow . . . . .	100
Gravity . . . . .	105
Theory for the Wall Boundary Condition . . . . .	108
Prescribing Inlet and Outlet Conditions . . . . .	114
Mass Flow . . . . .	115
Fully Developed Flow (Inlet) . . . . .	117
Fully Developed Flow (Outlet). . . . .	118
Normal Stress Boundary Condition. . . . .	120
Pressure Boundary Condition . . . . .	120
Porous Interface . . . . .	122
Mass Sources for Fluid Flow . . . . .	125
Numerical Stability — Stabilization Techniques for Fluid Flow . . . . .	127
Solvers for Laminar Flow. . . . .	129
Pseudo Time Stepping for Laminar Flow Models . . . . .	132
Nitsche Constraints. . . . .	133
Particle Tracing in Fluid Flow . . . . .	134
References for the Single-Phase Flow, Laminar Flow Interfaces . . . . .	135
<b>Theory for the Rotating Machinery, Laminar Flow Interface</b>	<b>137</b>
About Rotating Machinery . . . . .	137
Frozen Rotor . . . . .	138
Setting Up a Rotating Machinery Model . . . . .	140
References . . . . .	141

<b>Theory for the Viscoelastic Flow Interface</b>	<b>142</b>
General Viscoelastic Flow Theory . . . . .	142
Numerical Stability — Stabilization Techniques for Fluid Flow . . . . .	147
Solvers for Viscoelastic Flow . . . . .	147
Pseudo Time Stepping for Viscoelastic Flow Models . . . . .	147
References for the Viscoelastic Flow Interfaces . . . . .	148
 <b>Theory for the Potential Flow Interface</b>	 <b>149</b>
General Incompressible Potential Flow Theory . . . . .	149

## Chapter 4: Multiphase Flow Interfaces

<b>The Laminar Two-Phase Flow, Level Set and Laminar</b>	
<b>Two-Phase Flow, Phase Field Interfaces</b>	<b>152</b>
The Laminar Two-Phase Flow, Level Set Interface . . . . .	152
The Two-Phase Flow, Level Set Coupling Feature . . . . .	153
The Wetted Wall Coupling Feature. . . . .	157
The Interior Wetted Wall Coupling Feature . . . . .	160
The Laminar Two-Phase Flow, Phase Field Interface . . . . .	162
The Two-Phase Flow, Phase Field Coupling Feature. . . . .	163
Domain, Boundary, Point, and Pair Nodes for the Laminar Flow,	
Two-Phase, Level Set and Phase Field Interfaces. . . . .	167
 <b>The Two-Phase Flow, Level Set, Brinkman Equations Interface</b>	 <b>169</b>
 <b>The Laminar Three-Phase Flow, Phase Field Interface</b>	 <b>170</b>
The Laminar Three-Phase Flow, Phase Field Interface . . . . .	170
The Three-Phase Flow, Phase Field Coupling Feature . . . . .	171
Domain, Boundary, Point, and Pair Nodes for the Laminar	
Three-Phase Flow, Phase Field Interface. . . . .	173
 <b>The Laminar Two-Phase Flow, Moving Mesh Interface</b>	 <b>175</b>
Select Laminar Flow Properties . . . . .	176
Deforming Domain . . . . .	176
Fluid-Fluid Interface. . . . .	177

Free Surface. . . . .	178
Contact Angle. . . . .	180
<b>Theory for the Level Set and Phase Field Interfaces</b>	<b>182</b>
Level Set and Phase Field Equations . . . . .	182
Conservative and Nonconservative Formulations . . . . .	187
Phase Initialization . . . . .	187
Numerical Stabilization . . . . .	189
References for the Level Set and Phase Field Interfaces . . . . .	189
<b>Theory for the Three-Phase Flow, Phase Field Interface</b>	<b>191</b>
Governing Equations of the Three-Phase Flow, Phase Field Interface . . . . .	191
Reference for the Three-Phase Flow, Phase Field Interface . . . . .	194
<b>Theory for the Two-Phase Flow, Moving Mesh Interface</b>	<b>195</b>
Domain Level Fluid Flow . . . . .	195
About the Moving Mesh . . . . .	196
About the Fluid Interface Boundary Conditions . . . . .	197
Contact Angle Boundary Conditions . . . . .	201
References for the Two-Phase Flow, Moving Mesh Interface . . . . .	202

## Chapter 5: Porous Media and Subsurface Flow Interfaces

<b>The Brinkman Equations Interface</b>	<b>206</b>
Domain, Boundary, Point, and Pair Nodes for the Brinkman Equations Interface. . . . .	209
Porous Medium . . . . .	210
Fluid . . . . .	211
Porous Matrix . . . . .	212
Mass Source . . . . .	213
Volume Force . . . . .	213
Initial Values . . . . .	214
Fluid Properties . . . . .	214
Gravity . . . . .	215

<b>The Free and Porous Media Flow, Brinkman Interface</b>	<b>217</b>
Domain, Boundary, Point, and Pair Nodes for the Free and Porous Media Flow, Brinkman Interface . . . . .	219
Fluid Properties . . . . .	220
Porous Medium . . . . .	221
Fluid . . . . .	221
Porous Matrix . . . . .	222
Mass Source . . . . .	223
Volume Force . . . . .	223
Initial Values . . . . .	224
Gravity . . . . .	224
Wall . . . . .	224
 <b>Theory for the Brinkman Equations Interface</b>	 <b>226</b>
About the Brinkman Equations . . . . .	226
Brinkman Equations Theory. . . . .	227
References for the Brinkman Equations Interface. . . . .	228
 <b>Theory for the Free and Porous Media Flow, Brinkman Interface</b>	 <b>229</b>
Reference for the Free and Porous Media Flow, Brinkman Interface . . . . .	229

Chapter 6: Mathematics, Moving Interfaces

<b>The Level Set Interface</b>	<b>232</b>
The Level Set Interface . . . . .	232
The Level Set in Porous Media Interface . . . . .	233
Domain, Boundary, and Pair Nodes for the Level Set Interface . . . . .	234
Level Set Model . . . . .	234
Porous Medium . . . . .	235
Porous Matrix . . . . .	235
Initial Values . . . . .	236
Inlet. . . . .	236
No Flow . . . . .	237
Open Boundary . . . . .	237
Thin Barrier. . . . .	237



Continuity . . . . .	237
<b>The Phase Field in Fluids Interface</b>	<b>239</b>
Domain, Boundary, and Pair Nodes for the Phase Field in Fluids	
Interface. . . . .	240
Phase Field Model . . . . .	241
Initial Values . . . . .	242
Inlet. . . . .	243
Wetted Wall . . . . .	243
Interior Wetted Wall . . . . .	244
Continuity . . . . .	244
<b>The Ternary Phase Field Interface</b>	<b>245</b>
Domain, Boundary, and Pair Nodes for the Ternary Phase Field	
Interface. . . . .	246
Mixture Properties . . . . .	246
Initial Values . . . . .	248
Inlet. . . . .	248
Outlet . . . . .	248
Symmetry . . . . .	249
Wetted Wall . . . . .	249
<b>Theory for the Level Set Interface</b>	<b>251</b>
The Level Set Method . . . . .	251
Conservative and Nonconservative Form . . . . .	253
The Level Set Equation in Porous Domains . . . . .	253
Initializing the Level Set Function . . . . .	254
Variables For Geometric Properties of the Interface . . . . .	255
Reference for the Level Set Interface . . . . .	255
<b>Theory for the Phase Field in Fluids Interface</b>	<b>256</b>
About the Phase Field Method. . . . .	256
The Equations for the Phase Field Method . . . . .	256
Conservative and Nonconservative Forms . . . . .	258
Additional Sources of Free Energy . . . . .	259
Initializing the Phase Field Function . . . . .	259
Variables and Expressions . . . . .	260
Reference for the Phase Field in Fluids Interface . . . . .	261

<b>Theory for the Ternary Phase Field Interface</b>	<b>262</b>
About the Phase Field Method. . . . .	262
The Equations of the Ternary Phase Field Method . . . . .	262
Reference for the Ternary Phase Field Interface . . . . .	264

## Chapter 7: Chemical Species Transport Interfaces

<b>The Transport of Diluted Species Interface</b>	<b>266</b>
The Transport of Diluted Species in Porous Media Interface . . . . .	270
Domain, Boundary, and Pair Nodes for the Transport of Diluted Species Interface. . . . .	272
Prescribing Conditions on Fluid-Solid Interfaces . . . . .	274
Species Properties . . . . .	274
Transport Properties . . . . .	275
Solid . . . . .	277
Initial Values . . . . .	278
Mass-Based Concentrations. . . . .	278
Reactions. . . . .	279
No Flux . . . . .	280
Inflow . . . . .	281
Outflow . . . . .	282
Concentration . . . . .	282
Flux . . . . .	282
Symmetry . . . . .	284
Flux Discontinuity . . . . .	284
Partition Condition . . . . .	284
Periodic Condition . . . . .	285
Line Mass Source. . . . .	287
Point Mass Source . . . . .	288
Open Boundary . . . . .	288
Thin Diffusion Barrier . . . . .	289
Thin Impermeable Barrier . . . . .	289
Equilibrium Reaction . . . . .	289
Surface Reactions . . . . .	290
Surface Equilibrium Reaction . . . . .	291
Fast Irreversible Surface Reaction . . . . .	291

Porous Electrode Coupling . . . . .	291
Reaction Coefficients . . . . .	292
Electrode Surface Coupling . . . . .	292
Porous Medium . . . . .	293
Fluid . . . . .	293
Porous Matrix . . . . .	295
Dispersion . . . . .	296
Unsaturated Porous Medium . . . . .	296
Liquid . . . . .	297
Gas . . . . .	299
Adsorption . . . . .	300
Volatilization . . . . .	302
Species Source. . . . .	302
Hygroscopic Swelling . . . . .	303
Out-of-Plane Flux . . . . .	304
Fracture . . . . .	306
Fluid (Fracture) . . . . .	306
Fracture Material. . . . .	307
<b>Theory for the Transport of Diluted Species Interface</b>	<b>308</b>
Mass Balance Equation. . . . .	309
Equilibrium Reaction Theory . . . . .	310
Convective Term Formulation. . . . .	312
Solving a Diffusion Equation Only . . . . .	313
Mass Sources for Species Transport . . . . .	313
Adding Transport Through Migration . . . . .	315
Supporting Electrolytes . . . . .	316
Crosswind Diffusion . . . . .	317
Danckwerts Inflow Boundary Condition . . . . .	318
Mass Balance Equation for Transport of Diluted Species in Porous Media . . . . .	319
Convection in Porous Media . . . . .	320
Diffusion in Porous Media . . . . .	322
Dispersion . . . . .	323
Adsorption . . . . .	325
Reactions. . . . .	327
Mass Transport in Fractures . . . . .	328
References . . . . .	329

Chapter 8: Heat Transfer and Nonisothermal Flow  
Interfaces

<b>Modeling Heat Transfer in the Polymer Flow Module</b>	<b>332</b>
Specifying Material Properties for Non-Newtonian Materials . . . . .	332
Selecting the Right Physics Interface. . . . .	332
<b>Boundary Features in the Heat Transfer Interfaces</b>	<b>334</b>
Inflow . . . . .	334
Open Boundary . . . . .	335
<b>The Nonisothermal Flow, Laminar Flow and Viscoelastic Flow Interfaces</b>	<b>337</b>
Advantages of Using the Multiphysics Interfaces . . . . .	337
The Nonisothermal Flow, Laminar Flow and Viscoelastic Flow Interfaces . . . . .	337
<b>Theory for the Nonisothermal Viscoelastic Flow</b>	<b>340</b>
Nonisothermal Viscoelastic Flow Theory. . . . .	340
References for the Nonisothermal Viscoelastic Flow Interfaces . . . . .	341

Chapter 9: Chemorheology and Curing Kinetics

<b>The Curing Reaction Interface</b>	<b>344</b>
The Curing Reaction Interface. . . . .	344
Domain, Boundary, and Global Nodes for the Curing Reaction Interface. . . . .	345
Curing Kinetics . . . . .	346
No Flux . . . . .	347
Conversion . . . . .	348
Initial Values . . . . .	348
Open Boundary . . . . .	348
Symmetry . . . . .	348
Chemorheology . . . . .	348
Curing Reaction Heating . . . . .	350

<b>Theory for the Curing Reaction Interface</b>	<b>352</b>
Kinetics Models . . . . .	352
Pseudo Time Stepping . . . . .	353
Chemorheological Models . . . . .	354
Reference for the Curing Reaction Interface . . . . .	356

## Chapter 10: Parameter Estimation

<b>Index</b>	<b>359</b>
--------------	------------



# Introduction

This guide describes the Polymer Flow Module, an optional add-on package for COMSOL Multiphysics® designed to model and simulate the flow of Newtonian and non-Newtonian fluids.

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

In this chapter:

- [About the Polymer Flow Module](#)
- [Overview of the User's Guide](#)

# About the Polymer Flow Module

In this section:

- [The Polymer Flow Module](#)
- [The Polymer Flow Module Physics Interface Guide](#)
- [Common Physics Interface and Feature Settings and Nodes](#)
- [Where Do I Access the Documentation and Application Libraries?](#)

## *The Polymer Flow Module*

---

Non-Newtonian fluids are found in a great variety of processes in the polymer, food, pharmaceutical, cosmetics, household, and fine chemicals industries. Examples of these fluids are coatings, paints, yogurt, ketchup, colloidal suspensions, aqueous suspensions of drugs, lotions, creams, shampoo, and suspensions of peptides and proteins, to mention a few. Modeling and simulation can be used to design and optimize processes where these fluids are involved.

The Polymer Flow Module is intended for the modeling and simulation of non-Newtonian fluid flow with viscoelastic, thixotropic, shear thickening, or shear thinning properties. Dependencies of fluid properties on temperature and composition can be accounted for in order to model cure and polymerization. Manufacturing processes often involve free surfaces, or surfaces between Newtonian and non-Newtonian fluids: extrusion processes, coating flows, and injection molding. The Polymer Flow Module includes the necessary Multiphase Flow functionality for modeling these flows.

For studies of mixing processes, a Rotating Machinery, Fluid Flow interface has also been included.

Fully coupled and time-dependent fluid-structure interactions can be modeled with the Polymer Flow Module in combination with other modules in COMSOL Multiphysics®.

## *The Polymer Flow Module Physics Interface Guide*

---

The Polymer Flow Module extends the functionality of the physics interfaces of the base package for COMSOL Multiphysics. The details of the physics interfaces and study types for the Polymer Flow Module are listed in the table. The functionality of


















the COMSOL Multiphysics base package is described in the *COMSOL Multiphysics Reference Manual*.




















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](#).

PHYSICS INTERFACE	ICON	TAG	SPACE DIMENSION	AVAILABLE STUDY TYPE
<b>Chemical Species Transport</b>				
Transport of Diluted Species <sup>1</sup>		tds	all dimensions	stationary; time dependent
<b>Reacting Flow</b>				
Laminar Flow, Diluted Species <sup>1</sup>		—	3D, 2D, 2D axisymmetric	stationary; time dependent
<b>Fluid Flow</b>				
<b>Single-Phase Flow</b>				
Creeping Flow		spf	3D, 2D, 2D axisymmetric	stationary; time dependent
Laminar Flow		spf	3D, 2D, 2D axisymmetric	stationary; time dependent
Viscoelastic Flow		vef	3D, 2D, 2D axisymmetric	stationary; time dependent
<b>Rotating Machinery, Fluid Flow</b>				
Rotating Machinery, Laminar Flow		spf	3D, 2D	frozen rotor; time dependent

PHYSICS INTERFACE	ICON	TAG	SPACE DIMENSION	AVAILABLE STUDY TYPE
 <i>Potential Flow</i>				
Incompressible Potential Flow		ipf	3D, 2D, 2D axisymmetric	stationary
 <b>Multiphase Flow</b>				
 <i>Two-Phase Flow, Moving Mesh</i>				
Laminar Two-Phase Flow, Moving Mesh		—	3D, 2D, 2D axisymmetric	time dependent
 <i>Two-Phase Flow, Level Set</i>				
Laminar Two-Phase Flow, Level Set		—	3D, 2D, 2D axisymmetric	time dependent with phase initialization
Two-Phase Flow, Level Set, Brinkman Equations		—	3D, 2D, 2D axisymmetric	time dependent with initialization
 <i>Two-Phase Flow, Phase Field</i>				
Laminar Two-Phase Flow, Phase Field		—	3D, 2D, 2D axisymmetric	time dependent with phase initialization
 <i>Three-Phase Flow, Phase Field</i>				
Laminar, Three-Phase Flow, Phase Field		—	3D, 2D, 2D axisymmetric	time dependent with phase initialization
 <b>Porous Media and Subsurface Flow</b>				
Brinkman Equations		br	3D, 2D, 2D axisymmetric	stationary; time dependent
Free and Porous Media Flow, Brinkman		fp	3D, 2D, 2D axisymmetric	stationary; time dependent


PHYSICS INTERFACE	ICON	TAG	SPACE DIMENSION	AVAILABLE STUDY TYPE
 <b>Nonisothermal Flow</b>				
Laminar Flow <sup>(2)</sup>		—	3D, 2D, 2D axisymmetric	stationary; time dependent; Stationary, one-way NITF; time dependent, one-way NITF
Viscoelastic Flow		—	3D, 2D, 2D axisymmetric	stationary; time dependent; stationary, one-way NITF; time dependent, one-way NITF
 <b>Fluid-Structure Interaction</b>				
Fluid–Solid Interaction, Fixed Geometry <sup>2</sup>		—	3D, 2D, 2D axisymmetric	stationary; stationary, one way; time dependent; time dependent, one way
Fluid–Solid Interaction, Viscoelastic Flow, Fixed Geometry <sup>2</sup>		—	3D, 2D, 2D axisymmetric	time dependent; time dependent with phase initialization
Fluid–Solid Interaction, Two-Phase Flow, Phase Field, Fixed Geometry <sup>2</sup>		—	3D, 2D, 2D axisymmetric	time dependent; time dependent with phase initialization
 <b>Heat Transfer</b>				
Heat Transfer in Fluids <sup>1</sup>		ht	all dimensions	stationary; time dependent
Heat Transfer in Solids and Fluids <sup>1</sup>		ht	all dimensions	stationary; time dependent
 <b>Curing</b>				
Curing Reaction		creq	3D, 2D, 2D axisymmetric, 1D	stationary; time dependent
 <b>Moving Interface</b>				
Level Set		ls	all dimensions	time dependent with phase initialization

PHYSICS INTERFACE	ICON	TAG	SPACE DIMENSION	AVAILABLE STUDY TYPE
Level Set in Porous Media		ls	all dimensions	time dependent with phase initialization
Phase Field in Fluids		pf	all dimensions	time dependent; time dependent with phase initialization
Ternary Phase Field		terpf	3D, 2D, 2D axisymmetric	time dependent
<sup>1</sup> This physics interface is included with the core COMSOL Multiphysics software but has added functionality for this module.				
<sup>2</sup> This physics interface is a predefined multiphysics coupling that automatically adds all the physics interfaces and coupling features required.				

### *Common Physics Interface and Feature Settings and Nodes*

There are several common settings and sections available for the physics interfaces and feature nodes. Some of these sections also have similar settings or are implemented in the same way no matter the physics interface or feature being used. There are also some physics feature nodes that display in COMSOL Multiphysics.

In each module’s documentation, only unique or extra information is included; standard information and procedures are centralized in the *COMSOL Multiphysics Reference Manual*.

	In the <i>COMSOL Multiphysics Reference Manual</i> see <a href="#">Table 2-4</a> for links to common sections and <a href="#">Table 2-5</a> to common feature nodes. You can also search for information: press F1 to open the <b>Help</b> window or Ctrl+F1 to open the <b>Documentation</b> window.
---	---

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

A number of online 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 on 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*.

Win

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.

## Opening the Documentation Window

	<p>To open the <b>Documentation</b> window:</p> <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>
---	--

## 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.

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

## Opening the Application Libraries Window

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

	<p>From the <b>File</b> menu, select <b>Application Libraries</b>.</p> <p>To include the latest versions of model examples, from the <b>File&gt;Help</b> menu select (  ) <b>Update COMSOL Application Libraries</b>.</p>
 	<p>From the <b>File</b> or <b>Windows</b> menu, select <b>Application Libraries</b>.</p> <p>To include the latest versions of model examples, from the <b>Help</b> menu select (  ) <b>Update COMSOL Application Libraries</b>.</p>

## 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 will be 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/product-update">www.comsol.com/product-update</a>
COMSOL Blog	<a href="http://www.comsol.com/blogs">www.comsol.com/blogs</a>
Discussion Forum	<a href="http://www.comsol.com/forum">www.comsol.com/forum</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/videos">www.comsol.com/videos</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 *Polymer Flow Module User's Guide* gets you started with modeling using COMSOL Multiphysics. The information in this guide is specific to this module. Instructions on how to use COMSOL 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.

---

## MODELING POLYMER FLOW

This chapter gives a brief introduction to [Newtonian and Non-Newtonian Fluids](#) and a summary of [The Polymer Flow Physics Interfaces](#).

## THE FLUID FLOW BRANCH

There are several fluid flow interfaces available. The various types of momentum transport that you can simulate includes single-phase flow and multiphase flow of Newtonian and non-Newtonian fluids. Every section describes the applicable physics interfaces in detail and concludes with the underlying interface theory.

### *Single-Phase Flow*

[Single-Phase Flow Interfaces](#) describes the Creeping Flow, Laminar Flow, Viscoelastic Flow, Incompressible Potential Flow and Rotating Machinery, Laminar Flow interfaces.

### *Multiphase Flow and Mathematics, Moving Interfaces*

[Multiphase Flow Interfaces](#) describes the Two-Phase Flow, Level Set and Phase Field interfaces; the Three-Phase Flow, Phase Field interface; and the Two-Phase Flow, Moving Mesh interface. Some multiphase flow is described using the Phase Field and Level Set interfaces found under [Mathematics, Moving Interfaces](#) branch. In this module, these physics are already integrated into the relevant physics interfaces.

### *Porous Media and Subsurface Flow*

The [Porous Media and Subsurface Flow Interfaces](#) chapter describes the **Brinkman Equations**, and **Free and Porous Media Flow, Brinkman** interfaces.



## THE CHEMICAL SPECIES TRANSPORT BRANCH

This branch contains physics interfaces for chemical species transport and the predefined Reacting Flow multiphysics coupling.

### *Chemical Species Transport*

[Chemical Species Transport Interfaces](#) describes the Transport of Diluted Species physics interface and the Reacting Flow, Diluted Species multiphysics interface.

## THE HEAT TRANSFER BRANCH

This branch contains physics interfaces for heat transfer in fluids and solids.

### *Heat Transfer*

[Heat Transfer and Nonisothermal Flow Interfaces](#) describes the added functionality for the Heat Transfer and Nonisothermal Flow interfaces in the Polymer Flow Module.

### *Curing*

[Chemorheology and Curing Kinetics](#) describes the Curing Reaction interface and the Curing Reaction Heating multiphysics coupling.



# Modeling Polymer Flow

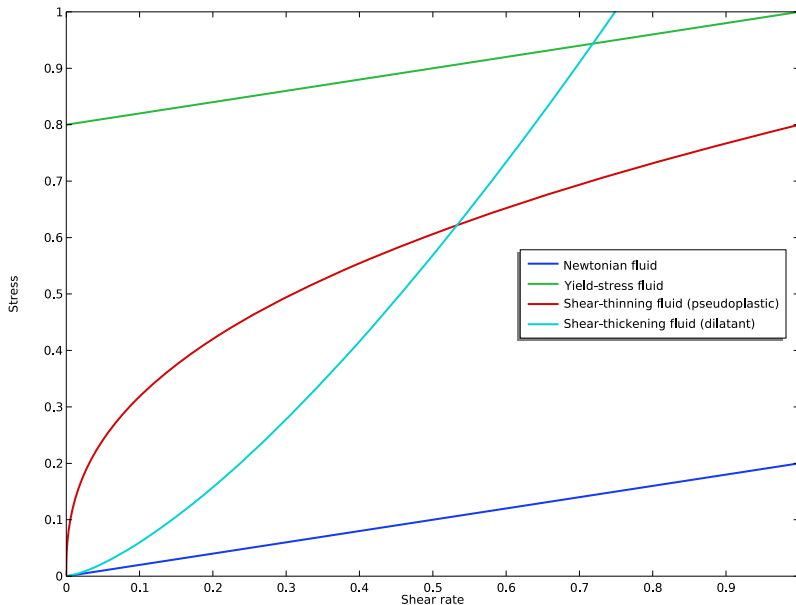
This chapter gives an overview of the physics interfaces available for modeling polymer flows and provides guidance on choosing the appropriate physics interface for a specific problem.

In this chapter:

- [Newtonian and Non-Newtonian Fluids](#)
- [The Polymer Flow Physics Interfaces](#)

# Newtonian and Non-Newtonian Fluids

A fluid may be characterized according to its response under the action of a shear stress. A Newtonian fluid has a linear relationship between shear stress and shear rate with the line passing through the origin. The constant of proportionality is referred to as the viscosity of the fluid and can depend on temperature, pressure, and composition. For a non-Newtonian fluid, the curve for shear stress versus shear rate is nonlinear or does not pass through the origin. If the curve is shifted away from the origin, the fluid has a yield stress. If the curve bends toward the shear-rate axis, the fluid is said to be shear thinning (pseudoplastic), but if it instead bends toward the shear-stress axis, it is said to be shear thickening (dilatant). The relationship may also contain time derivatives of the shear rate to model memory effects (thixotropy) and even parallel viscous and elastic responses. We refer to the latter as viscoelastic non-Newtonian fluids and all others as inelastic non-Newtonian fluids. For the inelastic non-Newtonian models, it is possible to define an apparent viscosity from a generalized Newtonian relationship between the deviatoric stress tensor and the strain-rate tensor.



*Figure 2-1: Conceptual behavior of shear stress versus shear rate for Newtonian, yield-stress, shear-thinning, and shear-thickening fluids.*

The apparent viscosity will then be a function of the first invariant of the strain-rate tensor. In simple shear flows, this invariant simplifies to the shear rate.

The inelastic non-Newtonian models in the Polymer Flow Module are listed below:

- Bingham–Papanastasiou (Viscoplastic)
- Casson–Papanastasiou (Viscoplastic)
- Power Law
- Carreau
- Carreau–Yasuda
- Cross
- Cross-Williamson
- Ellis
- Herschel–Bulkley–Papanastasiou
- Robertson–Stiff–Papanastasiou
- DeKee–Turcotte–Papanastasiou
- Houska thixotropy (Thixotropic)

The Bingham–Papanastasiou and Casson–Papanastasiou models are yield-stress fluids with a linear behavior at high shear rates. The Power Law model can be used to study both shear-thinning and shear-thickening behavior. The Carreau, Carreau–Yasuda, and Cross models are appropriate for shear-thinning fluids when there is a significant deviation from the Power Law model at very high and very low shear rates. Similarly, the Cross-Williamson and Ellis models can be used to study shear-thinning behavior when the deviations from the Power Law model are significant only at low shear rates. Herschel–Bulkley–Papanastasiou and Robertson–Stiff–Papanastasiou are power-law fluids with a yield-stress. The DeKee–Turcotte–Papanastasiou model is a yield-stress fluid with an exponential shear-thinning behavior at high shear rates. For the Houska thixotropy model, the apparent viscosity decreases with the time of shearing. Its structure breaks down under shear and rebuilds at rest. Thixotropic fluids display a hysteresis loop in the shear-rate shear-stress plane. The Houska model also includes a yield stress.

The viscoelastic models in the Polymer Flow Module are:

- Oldroyd-B
- Giesekus
- FENE-P

- FENE-CR
- LPTT
- EPTT
- Rolie–Poly

The simplest one is the Oldroyd-B model which has a constant viscosity and a constant first normal-stress coefficient. This is conceptually visualized as a Hookean spring in series with a dashpot in



To ensure material-frame indifference, an upper-convected time derivative is used to describe the transport of the additional stress due to the spring and dashpot. Multiple branches with different relaxation times and viscosities can be added in order to include multimodal responses. In the Oldroyd-B model, the springs are infinitely extensible, which can lead to unphysical behavior. The Giesekus model adds a quadratic stress term, thereby capturing power-law behavior in the viscous and normal stress components. The FENE-P and FENE-CR (Finitely Extensible Nonlinear Elastic with Peterlin and Chilcott–Rallison closure, respectively) models include nonlinear-spring behavior. Extension hardening and shear thinning can be studied with the LPTT and EPTT (Linear and Exponential Phan–Thien/Tanner) models. The Rouse Linear Entangled Polymers (Rolie–Poly) model is based on tube theory. The model incorporates the molecular motion mechanisms of reptation in the tubes, the stretching of polymer chains, and convective constraint release (CCR) due to the release of entanglements.

When applying a viscoelastic model, it is important to keep in mind that the mathematical formulation may become ill-posed at sharp corners. One remedy is to use a fillet operation to smooth the geometry at such points. As always when working with complex fluid-flow simulations, it is advisable to start with a simple model, and then switch to a more complicated model if the simple model does not capture the expected flow phenomena. In most industrial applications, inelastic non-Newtonian models produce sufficiently accurate results at a reasonable computational cost.

## References

---

1. R.P. Chhabra and J.F. Richardson, *Non-Newtonian Flow in the Process Industries*, Butterworth-Heinemann, 2004.

# The Polymer Flow Physics Interfaces

In this section:

- [Selecting the Right Physics Interface](#)
- [Single-Phase Flow](#)
- [Porous Media Flow](#)
- [Multiphase Flow](#)
- [Fluid-Structure Interaction](#)
- [Chemical Species Transport](#)
- [Heat Transfer and Nonisothermal Flow](#)
- [Coupling to Other Physics Interfaces](#)

## *Selecting the Right Physics Interface*


---


The Polymer Flow Module has a range of physics interfaces to use for fluid flow in a variety of circumstances. Often the selection of a particular physics interface implies certain assumptions about the equations of flow. If it is known in advance which assumptions are valid, then the appropriate physics interface can be added. However, when the flow type is unclear from the outset, or if it is difficult to reach a solution easily, starting with a simplified model and adding complexity subsequently is often a good approach. Using this approach, the results of a simulation are tested against the underlying assumptions and against experimental results, and the simulation can then be refined if necessary. For complex models it is often beneficial to take this approach even when the fluid flow is well characterized from the outset because a model with simplifying assumptions can be easier to solve initially. The solution process can then be fine-tuned for the more complex problem. Typically, [The Laminar Flow Interface](#) is a good starting point for problems involving Newtonian and inelastic non-Newtonian fluids, whereas [The Viscoelastic Flow Interface](#) must be chosen for problems involving viscoelastic non-Newtonian fluids.


The following sections describe the options available for simulating [Single-Phase Flow](#), [Multiphase Flow](#), [Chemical Species Transport](#), and [Heat Transfer and Nonisothermal Flow](#).

## Single-Phase Flow

---

The **Fluid Flow>Single-Phase Flow** branch () when adding a physics interface includes the Creeping Flow, Viscoelastic Flow and Rotating Machinery, Laminar Flow interfaces.


The **Creeping Flow Interface** () approximates the Navier–Stokes equations for very low Reynolds numbers. This is often referred to as Stokes flow and is applicable when viscous effects are dominant, such as in very small channels or microfluidics devices. The Creeping Flow interface also allows for simulation of inelastic non-Newtonian fluids.


The **Laminar Flow Interface** () is primarily applied to flows at low to intermediate Reynolds numbers. This physics interface solves the Navier–Stokes equations for incompressible, weakly compressible, and compressible flows (up to Mach 0.3). The Laminar Flow interface also allows for simulation of inelastic non-Newtonian fluids.



By selecting the **Neglect Inertial Form (Stokes Flow)** check box, in the **Settings** window for **Laminar Flow**, the Laminar Flow interface is converted into a Creeping Flow interface.

---

The **Rotating Machinery, Laminar Flow Interface** () combines the Laminar Flow interface and a Rotating Domain and is applicable to fluid-flow problems where one or more of the boundaries rotate (for example, in mixers and around propellers). The physics interface supports incompressible, weakly compressible, and compressible (Mach < 0.3) laminar flows of Newtonian and inelastic non-Newtonian fluids.

The **Viscoelastic Flow Interface** () is used to simulate incompressible isothermal flow of viscoelastic fluids. It solves the continuity equation, the momentum equation, and a constitutive equation that defines the elastic stresses. There are several predefined models for the elastic stresses: Oldroyd-B, FENE-P, Giesekus, LPTT, and EPTT.



The Settings window for the Laminar Flow interface is shown in [Figure 2-2](#).

*Figure 2-2: The Settings window for Laminar Flow. Use it to model incompressible, weakly compressible, or compressible ( $Ma < 0.3$ ) flow, as well as Stokes flow.*

### COMPRESSIBILITY

For compressible flow it is important that the density and any mass balances are well defined throughout the domain. Choosing to model incompressible flow simplifies the equations to be solved and decreases solution times. Most gas flows should be modeled as compressible flows; however, liquid flows can usually be treated as incompressible.

### NEGLECT INERTIAL TERM (STOKES FLOW)



For low Reynolds numbers, the inertial term in the Navier–Stokes equation may be neglected. When applicable, this approximation improves the convergence rate.

## INCLUDE GRAVITY

Gravity can be included when required. It can be neglected in models dominated by other effects.




## *Porous Media Flow*


---


The Brinkman Equations interface () is available for modeling flow in porous medium. The Free and Porous Media Flow, Brinkman () interface is used for modeling the single-phase flow where free flow is connected to porous media.

## *Multiphase Flow*

---


The Two-Phase Flow, Level Set () ; Two-Phase Flow, Phase Field () ; and Two Phase Flow, Moving Mesh () interfaces are used to model two fluids separated by a fluid-fluid interface. The moving interface is tracked in detail using the level set method, the phase field method, or by a moving mesh, respectively. The level set and phase field methods use a fixed mesh and solve additional equations to track the interface location. The moving mesh method solves the Navier–Stokes equations on a moving mesh with boundary conditions to represent the interface. In this case, equations must be solved for the mesh deformation. Since a surface in the geometry is used to represent the interface between the two fluids in the Moving Mesh interface, the interface itself cannot break up into multiple disconnected surfaces. This means that the Moving Mesh interface cannot be applied to problems such as droplet formation in inkjet devices (in these applications the level set or phase field interfaces are appropriate). These physics interfaces support incompressible flows, where one or both fluids can be non-Newtonian.

The Brinkman Equations, Level Set interface () is used to track the interface between two fluids in porous medium.

The Laminar Three-Phase Flow, Phase Field interface () models laminar flow of three incompressible phases that can be Newtonian or non-Newtonian. The moving fluid-fluid interfaces between the three phases are tracked in detail using the phase-field method.


## *Fluid-Structure Interaction*


---

The Fluid-Structure Interaction interfaces () couple a Single-Phase Flow or Two-Phase Flow, Phase Field interface to a Solid Mechanics interface for studies of deformation induced by fluid forces.

## *Chemical Species Transport*


---


The Transport of Diluted Species Interface () is available for modeling the transport of chemical species and ions. The assumption in these physics interfaces is that one component, a solvent, is present in excess (typically more than 90 mol%). This means that the mixture properties, such as density and viscosity, are independent of concentration. To model concentrated species, the Chemical Reaction Engineering Module is recommended in addition to the Polymer Flow Module.


The Laminar Flow interface () under the Reacting Flow branch combines the functionality of the Single-Phase Flow and Transport of Concentrated Species interfaces. The physics interface is primarily applied to model flow at low to intermediate Reynolds numbers in situations where the mass transport and flow fields have to be coupled, for example in polymerization processes.

## *Heat Transfer and Nonisothermal Flow*

---

The Nonisothermal Flow, Laminar Flow interface () is primarily applied to model flow at low to intermediate Reynolds numbers in situations where the temperature and flow fields have to be coupled. Typical examples include natural convection, where thermal buoyancy forces drive the flow, temperature dependence of constitutive parameters, and curing. This is a multiphysics interface for which the nonlocal couplings between fluid flow and heat transfer are set up automatically. The Nonisothermal Flow, Laminar Flow interface in the Polymer Flow Module has extended functionality, including additional boundary conditions and the Boussinesq approximation.

The Nonisothermal Flow, Viscoelastic Flow interface () is used to model nonisothermal flow of viscoelastic fluids.

The Curing Reaction interface () is intended for modeling the curing of polymers during the heat treatment. The Curing Reaction Heating multiphysics coupling allows to account for reaction heating.

## *Coupling to Other Physics Interfaces*

---

The COMSOL Multiphysics software is not limited to the aforementioned multiphysics couplings — there are many other phenomena that can be easily modeled in polymer flows. In many cases, physics interfaces can be coupled simply by entering expressions from one physics interface (such as the local temperature) into fields in

another physics interface (for example, in the **Model Inputs** section of a **Fluid Properties** node). Additionally, COMSOL Multiphysics includes powerful coupling operators that enable physics interfaces to be linked by projections, extrusions, or scalar quantities such as integrals and global averages.



Nonlocal Couplings and Coupling Operators in the *COMSOL Multiphysics Reference Manual*

---

Other types of fluid flow, such as turbulent flow and dispersed multiphase flow, can be found in the CFD Module. More extensive descriptions of heat transfer, such as in turbulent flow or involving radiation, can be found in the Heat Transfer Module. Furthermore, some applications involving the flow of liquids and gases in porous media are better handled by the Chemical Reaction Engineering Module.

## Single-Phase Flow Interfaces

There are several Single-Phase Flow interfaces available with the Polymer Flow Module. These are grouped by type and found under the **Fluid Flow>Single-Phase Flow** branch (≡) when adding a physics interface. See [Modeling Polymer Flow](#) to help you select which physics interface to use.

In this chapter:

- [The Creeping Flow and Laminar Flow Interfaces](#)
- [The Potential Flow Interface](#)
- [The Rotating Machinery, Laminar Flow Interface](#)
- [The Viscoelastic Flow Interface](#)
- [Theory for the Single-Phase Flow Interfaces](#)
- [Theory for the Rotating Machinery, Laminar Flow Interface](#)
- [Theory for the Viscoelastic Flow Interface](#)
- [Theory for the Potential Flow Interface](#)

See [The Single-Phase Flow, Laminar Flow Interface](#) in the *COMSOL Multiphysics Reference Manual* for other Fluid Flow interface and feature node settings.

# The Creeping Flow and Laminar Flow Interfaces


In this section:

- [The Creeping Flow Interface](#)
- [The Laminar Flow Interface](#)
- [Domain, Boundary, Pair, and Point Nodes for Single-Phase Flow](#)




- [Theory of Laminar Flow](#) in the *COMSOL Multiphysics Reference Manual*

---

In the *COMSOL Multiphysics Reference Manual*, see [Table 2-4](#) for links to common sections such as **Discretization**, **Consistent Stabilization**, **Inconsistent Stabilization**, and **Advanced Settings** sections, some of them accessed by clicking the **Show** button (  ) and choosing the applicable option. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

## *The Creeping Flow Interface*

---

The **Creeping Flow (spf)** interface (  ) is used for simulating fluid flows at very low Reynolds numbers for which the inertial term in the Navier–Stokes equations can be neglected. Creeping flow, also referred to as Stokes flow, occurs in systems with high viscosity or small geometrical length scales (for example, in microfluidics and MEMS devices). The fluid can be compressible or incompressible, as well as Newtonian or non-Newtonian.

The equations solved by the Creeping Flow interface are the Stokes equations for conservation of momentum and the continuity equation for conservation of mass.

The Creeping Flow interface can be used for stationary and time-dependent analyses.

The main feature is the **Fluid Properties** node, which adds the Stokes equations and provides an interface for defining the fluid material and its properties. Except where noted below, see [The Laminar Flow Interface](#) for all other settings.

## PHYSICAL MODEL

By default, the **Neglect inertial term (Stokes flow)** check box is selected. If unchecked, or, if the **Turbulence model type** is changed to **RANS**, the inertial terms are included in the computations.


## DISCRETIZATION

By default, the Creeping Flow interface uses **P2+P1** elements. Contrary to general laminar and turbulent single-phase flow simulations employing purely linear P1+P1 elements, P2+P1 elements are well suited for most creeping flow simulations but are not in general fully conservative. Full conservation properties and high accuracy is obtained by selecting P2+P2 or P3+P3 elements.

Note that P1+P1, P2+P2, and P3+P3 discretizations (the so-called equal order interpolations) require streamline diffusion to be active.

### *The Laminar Flow Interface*

---

The **Laminar Flow (spf)** interface () is used to compute the velocity and pressure fields for the flow of a single-phase fluid in the laminar flow regime. A flow remains laminar as long as the Reynolds number is below a certain critical value. At higher Reynolds numbers, disturbances have a tendency to grow and cause transition to turbulence. This critical Reynolds number depends on the model, but a classical example is pipe flow, where the critical Reynolds number is known to be approximately 2000.

The physics interface supports incompressible flow, weakly compressible flow (the density depends on temperature but not on pressure), and compressible flow at low Mach numbers (typically less than 0.3). It also supports flow of non-Newtonian fluids.

The equations solved by the Laminar Flow interface are the Navier–Stokes equations for conservation of momentum and the continuity equation for conservation of mass.

The Laminar Flow interface can be used for stationary and time-dependent analyses. Time-dependent studies should be used in the high-Reynolds number regime as these flows tend to become inherently unsteady.

When the Laminar Flow interface is added, the following default nodes are also added in the **Model Builder: Fluid Properties**, **Wall** (the default boundary condition is **No slip**), and **Initial Values**. Other nodes that implement, for example, boundary conditions and volume forces, can be added from the **Physics** toolbar or from the context menu displayed when right-clicking **Laminar Flow**.

## 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. Physics interface variables can be referred to 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 `spf`.

### *About the Physics Interface Label Names*

The label for a single-phase flow interface (the node name in the Model Builder) is dynamic and is reset according to the turbulence model and the neglect inertial term (Stokes flow) property set at the physics interface level. The availability also depends on the software license.

If the **Neglect inertial term (Stokes flow)** check box is selected, then the Label changes to **Creeping Flow**, which is the same Label that displays when that interface is added from the Model Wizard or Add Physics window.

## PHYSICAL MODEL

### *Compressibility*

Depending of the fluid properties and the flow regime, three options are available for the **Compressibility** option. In general the computational complexity increases from **Incompressible flow** to **Weakly compressible flow** to **Compressible flow (Ma<0.3)** but the underlying hypotheses are increasingly more restrictive in the opposite direction.

When the **Incompressible flow** option (default) is selected, the incompressible form of the Navier–Stokes and continuity equations is applied. In addition, the fluid density is evaluated at the **Reference pressure level** defined in this section. The **Reference temperature** is set to 293.15 K.

The **Weakly compressible flow** option models compressible flow when the pressure dependency of the density can be neglected. When selected, the compressible form of the Navier–Stokes and continuity equations is applied. In addition, the fluid density is evaluated at the **Reference pressure level** defined in this section.

When the **Compressible flow (Ma<0.3)** option is selected, the compressible form of the Navier–Stokes and continuity equations is applied. **Ma < 0.3** indicates that the inlet and outlet conditions, as well as the stabilization, may not be suitable for transonic and



supersonic flow. For more information, see [The Mach Number Limit](#) in the *COMSOL Multiphysics Reference Manual*.

#### *Swirl Flow*

The velocity component,  $u_\phi$ , in the azimuthal direction can be included for 2D axisymmetric components by selecting the **Swirl flow** check box. While  $u_\phi$  can be nonzero, there can be no gradients in the  $\phi$  direction. Also see [General Single-Phase Flow Theory](#).

#### *Neglect Inertial Term (Stokes Flow)*

For low Reynolds number flows, the inertial terms in the Navier–Stokes equations may be neglected.

#### *Porous Media Domains*

With the addition of various modules, the **Enable porous media domains** check box is available. Selecting this option, a **Fluid and Matrix Properties** node, a **Mass Source** node, and a **Forchheimer Drag** subnode are added to the physics interface. These are described for the **Brinkman Equations** interface in the respective module’s documentation. The **Fluid and Matrix Properties** can be applied on all domains or on a subset of the domains.

#### *Porous Treatment of No Slip Condition*

**Porous treatment of no slip condition** is available when **Enable porous media domains** is activated. It controls how the **No Slip** condition on **Wall** boundaries and **Interior Wall** boundaries adjacent to **Porous Medium** should be treated. The options are **Standard no slip formulation** (default) and **Porous slip**. The latter option provides a unified treatment when the porous matrix is fully resolved as well as when it is under resolved ensuring a smooth transition between regions with different resolutions; see No Slip under [Wall](#).

#### *Include Gravity*

When the **Include gravity** check box is selected, a global **Gravity** feature is shown in the interface model tree, and the buoyancy force is included in the Navier–Stokes equations.

Also, when the **Include gravity** check box is selected, the **Use reduced pressure** option changes the pressure formulation from using the total pressure (default) to using the reduced pressure. This option is suitable for configurations where the density changes

are very small; otherwise, the default formulation can be used. For more information, see [Gravity](#).



Natural convection is induced by variations in magnitude of the buoyancy force that is defined from the density. For incompressible flow, the density is constant; hence the buoyancy force is homogeneous and natural convection cannot be modeled. Nevertheless, using the **Nonisothermal Flow** multiphysics coupling, the buoyancy force is defined using the thermal expansion coefficient and the temperature. Hence, in this case natural convection is accounted for, following the Boussinesq approximation.

### Reference Values

Reference values are global quantities used to evaluate the density of the fluid when the **Incompressible flow** or the **Weakly compressible flow** option is selected.

**Reference pressure level** There are generally two ways to include the pressure in fluid flow computations: either to use the absolute pressure  $p_A = p + p_{\text{ref}}$ , or the gauge pressure  $p$ . When  $p_{\text{ref}}$  is nonzero, the physics interface solves for the gauge pressure whereas material properties are evaluated using the absolute pressure. The reference pressure level is also used to define the reference density.

**Reference temperature** The reference temperature is used to define the reference density.

**Reference position** When **Include gravity** is selected, the reference position can be defined. It corresponds to the location where the total pressure (that includes the hydrostatic pressure) is equal to the **Reference pressure level**.

### DEPENDENT VARIABLES

The following dependent variables (fields) are defined for this physics interface — the **Velocity field  $\mathbf{u}$**  and its components, and the **Pressure  $p$** .

If required, the names of the field, component, and dependent variable can be edited. Editing the name of a scalar dependent variable changes both its field name and the dependent variable name. If a new field name coincides with the name of another field of the same type, the fields share degrees of freedom and dependent variable names. A new field name must not coincide with the name of a field of another type or with a

component name belonging to some other field. Component names must be unique within a model except when two fields share a common field name.

### CONSISTENT STABILIZATION

To enable this section, click the **Show More Options** button (  ) and select **Stabilization** in the **Show More Options** dialog box.

There are two *consistent stabilization methods*: **Streamline diffusion** and **Crosswind diffusion**. Usually, both check boxes for these methods are selected by default and should remain selected for optimal performance. Consistent stabilization methods do not perturb the original transport equation. Streamline diffusion must be selected when using equal-order interpolation for pressure and velocity.

Select the **Use dynamic subgrid time scale** check box to estimate the influence of temporal variations in the velocity field, instead of using the inverse time step, in the consistent stabilization terms. This check box is selected by default

Select the **Limit small time steps effect on stabilization time scale** check box to prevent loss of stabilization at small time steps.

### INCONSISTENT STABILIZATION

To enable this section, click the **Show More Options** button (  ) and select **Stabilization** in the **Show More Options** dialog box.


There is usually just one *inconsistent stabilization method* — **Isotropic diffusion**. This method is equivalent to adding a term to the diffusion coefficient in order to dampen the effect of oscillations by making the system somewhat less dominated by convection. If possible, minimize the use of the inconsistent stabilization method because by using it you no longer solve the original problem. By default, the **Isotropic diffusion** check box is not selected because this type of stabilization adds artificial diffusion and affects the accuracy of the original problem. However, this option can be used to get a good initial guess for underresolved problems.

If required, select the **Isotropic diffusion** check box and enter a **Tuning parameter**  $\delta_{id}$  as a scalar positive value. The default value is 0.25 (a reasonable value to start with is roughly 0.5 divided by the element order). A higher value adds more isotropic diffusion.



- [Numerical Stability — Stabilization Techniques for Fluid Flow](#) in the *COMSOL Multiphysics Reference Manual*

## ADVANCED SETTINGS

To display this section, click the **Show More Options** button (  ) and select **Advanced Physics Options** in the **Show More Options** dialog box. Normally these settings do not need to be changed.

The **Use pseudo time stepping for stationary equation form** is per default set to **Automatic from physics**. This option can add pseudo time derivatives to the equation when the **Stationary equation** form is used in order to speed up convergence. Pseudo time stepping is triggered when the Laminar Flow interface is selected in some multiphysics coupling features. Set **Automatic from physics** to **On** to apply pseudo time stepping also for laminar flows. Set it to **Off** to disable pseudo time stepping completely.

When **Use pseudo time stepping for stationary equation form** is set to **Automatic from physics** or **On**, a **CFL number expression** should also be defined. For the default **Automatic** option, the local CFL number (from the Courant–Friedrichs–Lewy condition) is determined by a PID regulator.

The **Use Block Navier-Stokes preconditioner** check box under **Linear solver** is available when the **Compressibility** option is set to **Incompressible flow**. When this check box is selected, the default solver for time-dependent study steps will use the Block Navier–Stokes preconditioner in iterative solvers for the velocity and pressure. Using this preconditioner may result in shorter solution times for large time dependent problems with high Reynolds numbers.



In the *COMSOL Multiphysics Reference Manual*:

- [Pseudo Time Stepping for Laminar Flow Models](#)
- [Domain, Boundary, Pair, and Point Nodes for Single-Phase Flow](#)
- [Theory of Laminar Flow](#)
- [Block Navier–Stokes](#) in the *COMSOL Multiphysics Reference Manual*.

## DISCRETIZATION

The default discretization for Laminar Flow is **PI+PI** elements — that is, piecewise linear interpolation for velocity and pressure. This is suitable for most flow problems.

Some higher-order interpolations are also available, and they can be cost-effective options to obtain high accuracy for flows with low Reynolds numbers.

The **P2+P2** and **P3+P3** options, the equal-order interpolation options, are the preferred higher-order options because they have higher numerical accuracy than the mixed-order options **P2+P1** and **P3+P2**. The equal-order interpolation options do, however, require streamline diffusion to be active.



*Flow Past a Cylinder*: Application Library path **COMSOL\_Multiphysics/Fluid\_Dynamics/cylinder\_flow**

### *Domain, Boundary, Pair, and Point Nodes for Single-Phase Flow*

The following nodes, listed in alphabetical order, are available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or by right-clicking to access the context menu (all users).

- Boundary Stress
- Free Surface<sup>2</sup>
- Flow Continuity
- Fluid-Fluid Interface<sup>2</sup>
- Fluid Properties
- Gravity
- Initial Values
- Inlet
- Interior Wall
- Line Mass Source<sup>1</sup>
- Open Boundary
- Outlet
- Periodic Flow Condition
- Point Mass Source<sup>1</sup>
- Porous Interface
- Pressure Point Constraint
- Stationary Free Surface
- Symmetry
- Volume Force
- Wall

<sup>1</sup> A feature that may require an additional license

<sup>2</sup> See documentation for [The Laminar Two-Phase Flow, Moving Mesh Interface](#)



For 2D axisymmetric components, COMSOL Multiphysics takes the axial symmetry boundaries (at  $r = 0$ ) into account and adds an **Axial Symmetry** node that is valid on the axial symmetry boundaries only.

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.



For a detailed overview of the functionality available in each product, visit <https://www.comsol.com/products/specifications/>

---

## *Fluid Properties*

---

The **Fluid Properties** node adds the momentum and continuity equations solved by the physics interface, except for volume forces, which are added by the [Volume Force](#) feature. The node also provides an interface for defining the material properties of the fluid.

For the Viscoelastic Flow interface, the **Fluid Properties** node also adds the equations for the components of the elastic stress tensor.

### **MODEL INPUTS**

Fluid properties, such as density and viscosity, can be defined through user inputs, variables, or by selecting a material. For the latter option, additional inputs, for example temperature or pressure, may be required to define these properties.

#### *Temperature*

By default, the **Temperature** model input is set to **Common model input**, and the temperature is controlled from [Default Model Inputs](#) under **Global Definitions** or by a locally defined [Model Input](#). If a Heat Transfer interface is included in the component, it controls the temperature **Common model input**. Alternatively, the temperature field can be selected from another physics interface. All physics interfaces have their own tags (**Name**). For example, if a Heat Transfer in Fluids interface is included in the component, the **Temperature (ht)** option is available for  $T$ .

You can also select **User defined** from the **Temperature** model input in order to manually prescribe  $T$ .


#### *Absolute Pressure*

This input appears when a material requires the absolute pressure as a model input. The absolute pressure is used to evaluate material properties, but it also relates to the value of the calculated pressure field. There are generally two ways to calculate the pressure when describing fluid flow: either to solve for the absolute pressure or for a

pressure (often denoted gauge pressure) that relates to the absolute pressure through a reference pressure.

The choice of pressure variable depends on the system of equations being solved. For example, in a unidirectional incompressible flow problem, the pressure drop over the modeled domain is probably many orders of magnitude smaller than the atmospheric pressure, which, when included, may reduce the stability and convergence properties of the solver. In other cases, such as when the pressure is part of an expression for the gas volume or the diffusion coefficients, it may be more convenient to solve for the absolute pressure.

The default **Absolute pressure**  $p_A$  is  $p + p_{\text{ref}}$ , where  $p$  is the dependent pressure variable from the Navier–Stokes or RANS equations, and  $p_{\text{ref}}$  is from the user input defined at the physics interface level. When  $p_{\text{ref}}$  is nonzero, the physics interface solves for a gauge pressure. If the pressure field instead is an absolute pressure field,  $p_{\text{ref}}$  should be set to 0.

The **Absolute pressure** field can be edited by clicking **Make All Model Inputs Editable** (  ) and entering the desired value in the input field.



---

Model Inputs and Multiphysics Couplings in the *COMSOL Multiphysics Reference Manual*

---

## FLUID PROPERTIES

### *Density*

The density can either be specified by a material, or by a **User defined** expression. The density in a material can depend on temperature and pressure, and these dependencies are automatically replaced by  $p_{\text{ref}}$  for weakly compressible flows and  $p_{\text{ref}}$  and  $T_{\text{ref}}$  for incompressible flows (as specified by the **Compressibility** setting at the physics interface level). If density variations with respect to pressure are to be included in the computations, **Compressibility** must be set to compressible. Any dependencies in the density on quantities other than temperature and pressure must be consistent with the **Compressibility** setting at the interface level.

### *Constitutive Relation*

For laminar flow, the **Specify dynamic viscosity** (default) and **Inelastic non-Newtonian** options are available. Use the default option to either get the dynamic viscosity from a material, or to add a user defined expression. The **Inelastic non-Newtonian** option

includes several inelastic non-Newtonian models for the relationship between the shear stress and shear rate.

If the **Specify dynamic viscosity** option is selected, an inelastic non-Newtonian model can be selected by adding the corresponding **Material Properties** group as a subnode to a material node.

#### *Dynamic Viscosity*

The **Dynamic viscosity**  $\mu$  describes the relationship between the shear rate and the shear stresses in a fluid. Intuitively, water and air have low viscosities, and substances often described as thick (such as oil) have higher viscosities.

#### *Inelastic Non-Newtonian*

Non-Newtonian fluids are encountered in everyday life and in a wide range of the industrial processes. Examples of non-Newtonian fluids include yogurt, paper pulp, and polymer suspensions. Such fluids have a nonlinear relationship between the shear stress and the share rate. The following inelastic non-Newtonian models are available: Power law, Carreau, Carreau–Yasuda, Cross, Cross–Williamson, Sisko, Bingham–Papanastasiou, Herschel–Bulkley–Papanastasiou, Casson–Papanastasiou, DeKee–Turcotte–Papanastasiou, Robertson–Stiff–Papanastasiou, Ellis, and Houska thixotropy.

The following parameters are required for the **Power law**:

- **Fluid consistency coefficient**  $m$
- **Flow behavior index**  $n$
- **Lower shear rate limit**  $\dot{\gamma}_{\min}$
- **Reference shear rate**  $\dot{\gamma}_{\text{ref}}$

For the **Carreau model**, the following parameters are required:

- **Zero shear rate viscosity**  $\mu_0$
- **Infinite shear rate viscosity**  $\mu_{\text{inf}}$
- **Relaxation time**  $\lambda$
- **Power index**  $n$

For the **Carreau–Yasuda model**, the following parameters are required:

- **Zero shear rate viscosity**  $\mu_0$



- **Infinite shear rate viscosity**  $\mu_{\text{inf}}$
- **Relaxation time**  $\lambda$
- **Power index**  $n$
- **Transition parameter**  $\alpha$

For the **Cross model**, the following parameters are required:

- **Zero shear rate viscosity**  $\mu_0$
- **Infinite shear rate viscosity**  $\mu_{\text{inf}}$
- **Power index**  $n$
- **Critical stress**  $\tau_{\text{tr}}$

For the **Cross-Williamson model**, the following parameters are required:

- **Zero shear rate viscosity**  $\mu_0$
- **Relaxation time**  $\lambda$
- **Power index**  $n$

For the **Sisko model**, the following parameters are required:

- **Infinite shear rate viscosity**  $\mu_{\text{inf}}$
- **Fluid consistency coefficient**  $m$
- **Flow behavior index**  $n$
- **Reference shear rate**  $\dot{\gamma}_{\text{ref}}$

For the **Bingham–Papanastasiou model**, the following parameters are required:

- **Plastic viscosity**  $\mu_p$
- **Yield stress**  $\tau_y$
- **Model parameter**  $m_p$

For the **Herschel–Bulkley–Papanastasiou model**, the following parameters are required:

- **Fluid consistency coefficient**  $m$
- **Flow behavior index**  $n$
- **Yield stress**  $\tau_y$
- **Reference shear rate**  $\dot{\gamma}_{\text{ref}}$
- **Model parameter**  $m_p$

For the **Casson–Papanastasiou model**, the following parameters are required:

- **Plastic viscosity**  $\mu_p$
- **Yield stress**  $\tau_y$
- **Model parameter**  $m_p$

For the **DeKee–Turcotte–Papanastasiou model**, the following parameters are required:

- **Shear rate viscosity**  $\mu_{DK}$
- **Relaxation time**  $\lambda_{DK}$
- **Yield stress**  $\tau_y$
- **Model parameter**  $m_p$

For the **Robertson–Stiff–Papanastasiou model**, the following parameters are required:

- **Fluid consistency coefficient**  $m$
- **Flow behavior index**  $n$
- **Yield stress**  $\tau_y$
- **Reference shear rate**  $\dot{\gamma}_{ref}$
- **Model parameter**  $m_p$

For the **Ellis model**, the following parameters are required:

- **Zero shear rate viscosity**  $\mu_0$
- **Shear stress**  $\tau_{1/2}$
- **Power parameter**  $a_e$
- **Initial guess**  $\mu_{app,0}$

For the **Houska thixotropy model**, the following parameters are required:

- **Permanent fluid consistency coefficient**  $m_{y,0}$
- **Thixotropic fluid consistency coefficient**  $m_{y,t}$
- **Flow behavior index**  $n_f$
- **Permanent yield stress**  $\tau_{y,0}$
- **Thixotropic yield stress**  $\tau_{y,t}$
- **Breakdown index**  $n_b$
- **Rebuild coefficient**  $k_f$
- **Breakdown coefficient**  $k_b$
- **Reference shear rate**  $\dot{\gamma}_{ref}$

- **Model parameter  $m_p$**

The default values for the parameters are listed in [Table 3-1](#).

TABLE 3-1: DEFAULT VALUES FOR THE INELASTIC NON-NEWTONIAN MODELS.

NAME	UNIT	VARIABLE	DEFAULT VALUE
Fluid consistency coefficient	Pa·s	$m$	0.001
Flow behavior index / Power index	Dimensionless	$n, n_f$	1
Flow shear rate limit	$s^{-1}$	$\dot{\gamma}_{\min}$	0.01
Reference shear rate	$s^{-1}$	$\dot{\gamma}_{\text{ref}}$	1
Zero shear rate viscosity	Pa·s	$\mu_0$	0.001
Infinite shear rate viscosity	Pa·s	$\mu_{\text{inf}}$	0
Relaxation time	s	$\lambda, \lambda_{\text{DK}}$	0
Transition parameter	Dimensionless	$s$	2
Critical stress	N/m <sup>2</sup>	$\tau_{\text{tr}}$	1
Plastic viscosity	Pa·s	$\mu_p$	0.001
Yield stress	N/m <sup>2</sup>	$\tau_y$	0
Model parameter	s	$m_p$	10
Shear rate viscosity parameter	Pa·s	$\mu_{\text{DK}}$	0.001
Shear stress parameter	N/m <sup>2</sup>	$\tau_{1/2}$	1
Power parameter	Dimensionless	$a_e$	2
Initial guess of $\mu_{\text{app}}$	Pa·s	$\mu_{\text{app},0}$	$\mu_0$
Permanent fluid consistency coefficient	Pa·s	$m_{y,0}$	0.001
Thixotropic fluid consistency coefficient	Pa·s	$m_{y,t}$	0
Permanent yield stress	N/m <sup>2</sup>	$\tau_{y,0}$	0
Thixotropic yield stress	N/m <sup>2</sup>	$\tau_{y,t}$	0
Breakdown index	Dimensionless	$n_b$	1
Rebuild index	$s^{-1}$	$k_f$	1
Breakdown coefficient	$s^{-1}$	$k_b$	1

### Thermal Effects

If  $\mu_0$ ,  $\mu_p$ ,  $m$ , or  $\mu_{\text{DK}}$  is **User defined**, or the **Houska thixotropy** model is used, the thermal effects are activated with five options: **None**, **Arrhenius**, **Williams–Landel–Ferry (WLF)**, **Exponential**, and **User defined**.



- [Non-Newtonian Flow](#) in the *CFD Module User's Guide*

---

### *Volume Force*

---

The **Volume Force** node specifies the volume force  $\mathbf{F}$  on the right-hand side of the momentum equation.

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u} \cdot \nabla) \mathbf{u} = \nabla \cdot \left[ -p \mathbf{I} + \mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - \frac{2}{3} \mu(\nabla \cdot \mathbf{u}) \mathbf{I} \right] + \mathbf{F}$$

If several volume-force nodes are added to the same domain, then the sum of all contributions are added to the momentum equation.

---

### *Initial Values*

---

The initial values serve as initial conditions for a transient simulation or as an initial guess for a nonlinear solver in a stationary simulation. Note that for a transient compressible-flow simulation employing a material for which the density depends on the pressure (such as air), discontinuities in the initial values trigger pressure waves even when the Mach number is small. The pressure waves must be resolved and this puts a restriction on the time step.

#### **INITIAL VALUES**

Initial values or expressions should be specified for the **Velocity field**  $\mathbf{u}$  and the **Pressure**  $p$ .

When **Include gravity** is selected and **Use reduced pressure** not selected in the interface **Physical model** section, the **Compensate for hydrostatic pressure approximation** (named **Compensate for hydrostatic pressure** for compressible flows) check box is available and selected by default. When it is selected, the hydrostatic pressure is automatically added to the pressure entered in  $p$  user input.

## Wall

---

The **Wall** node includes a set of boundary conditions describing fluid-flow conditions at stationary, moving, and leaking walls. For turbulent flow, the description may involve wall functions and asymptotic expressions for certain turbulence variables.

### BOUNDARY CONDITION

Select a **Boundary condition** for the wall.

- No Slip
- Slip
- Leaking Wall
- Navier Slip
- Nonlinear Navier Slip
- Hatzikiriakos Slip
- Asymptotic Slip

### No Slip

**No slip** is the default boundary condition to model solid walls. A no-slip wall is a wall where the fluid velocity relative to the wall velocity is zero. For a stationary wall that means that  $\mathbf{u} = 0$ .

The option for **Porous treatment of no slip condition** is available when **Enable porous media domains** is activated in the **Physical Model** section in the settings for the main physics interface node. It specifies how **Wall** boundaries and **Interior Wall** boundaries adjacent to porous domains are treated. When **Standard no slip formulation** is chosen, a common no slip condition is applied on all solid walls. When the default **Porous slip** is chosen, a blending analytic expression is instead applied on the corresponding wall boundaries adjacent to the porous medium domain. It results in a no slip condition in case the porous length scale is fully resolved by the mesh and a slip condition in the opposite limit when the mesh is much coarser than the porous scale. A smooth transition between these limits is ensured. Note that the interpretation and usage of nonzero slip at the wall is the same as in the **Navier slip** boundary condition. By default, **Velocity formulation** is on and the treatment is based on an approximate reconstruction of the far field pressure gradient using the slip velocity at the wall. If the **Pressure-gradient formulation** is chosen, the local pressure gradient at the wall is employed.

### Slip

The **Slip** option prescribes a no-penetration condition,  $\mathbf{u} \cdot \mathbf{n} = 0$ . It is implicitly assumed that there are no viscous effects at the slip wall and hence, no boundary layer develops. From a modeling point of view, this can be a reasonable approximation if the main effect of the wall is to prevent fluid from leaving the domain.

### Leaking Wall

This boundary condition may be used to simulate a wall where fluid is leaking into or leaving the domain with the velocity  $\mathbf{u} = \mathbf{u}_l$  through a perforated wall. The components of the **Fluid velocity**  $\mathbf{u}_l$  on the leaking wall should be specified.

### Navier Slip

This boundary condition enforces no-penetration at the wall,  $\mathbf{u} \cdot \mathbf{n}_{\text{wall}} = 0$ , and adds a tangential stress

$$\mathbf{K}_{\text{nt}} = -\frac{\mu}{\beta} \mathbf{u}_{\text{slip}}$$

where  $\mathbf{K}_{\text{nt}} = \mathbf{K}_n - (\mathbf{K}_n \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$ ,  $\mathbf{K}_n = \mathbf{K}\mathbf{n}_{\text{wall}}$ , and  $\mathbf{K}$  is the viscous stress tensor.  $\beta$  is a slip length, and  $\mathbf{u}_{\text{slip}} = \mathbf{u} - (\mathbf{u} \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$  is the velocity tangential to the wall.

The **Slip length** setting is per default set to **Factor of minimum element length**. The slip length  $\beta$  is then defined as  $\beta = f_h h_{\text{min}}$ , where  $h_{\text{min}}$  is the smallest element side and  $f_h$  is a user input. Select **User defined** from the **Slip length** selection list in order to manually prescribe  $\beta$  (SI unit: m).

In cases where the wall movement is nonzero, check **Account for the translational wall velocity in the friction force** to use  $\mathbf{u} - \mathbf{u}_{\text{bnd}} - ((\mathbf{u} - \mathbf{u}_{\text{bnd}}) \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$  instead of  $\mathbf{u}_{\text{slip}}$  in the friction force.

For the Viscoelastic Flow interface,  $\mathbf{K}_n$  is the sum of the viscous and the elastic contributions.

### Nonlinear Navier Slip

This boundary condition enforces no-penetration at the wall,  $\mathbf{u} \cdot \mathbf{n}_{\text{wall}} = 0$ , and adds a tangential stress

$$\mathbf{K}_{\text{nt}} = -\left(\frac{\tau_{\text{ref}}}{\|\mathbf{K}_{\text{nt}}\|}\right)^{m-1} \frac{\mu}{\beta} \mathbf{u}_{\text{slip}}$$

where  $\mathbf{K}_{\text{nt}} = \mathbf{K}_n - (\mathbf{K}_n \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$ ,  $\mathbf{K}_n = \mathbf{K}\mathbf{n}_{\text{wall}}$ , and  $\mathbf{K}$  is the viscous stress tensor.  $\tau_{\text{ref}}$  is a reference value used for scaling and dimensionalization and

$\mathbf{u}_{\text{slip}} = \mathbf{u} - (\mathbf{u} \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$  is the velocity tangential to the wall. For further settings, see the [Navier Slip](#) condition.

For Viscoelastic Flow interface,  $\mathbf{K}_n$ , is the sum of the viscous and the elastic contributions.

#### *Hatzikiriakos Slip*

When Hatzikiriakos slip is selected, a slip velocity is applied when the tangential stress is larger than the specified yield stress,  $\tau_y$ ,

$$\mathbf{u}_{\text{slip}} = \begin{cases} k_1 \sinh([k_2(\|\mathbf{K}_{\text{nt}}\| - \tau_y)]) \frac{\mathbf{K}_{\text{nt}}}{\|\mathbf{K}_{\text{nt}}\|}, & \text{if } \|\mathbf{K}_{\text{nt}}\| > \tau_y \\ 0, & \text{if } \|\mathbf{K}_{\text{nt}}\| \leq \tau_y \end{cases}$$

where  $\mathbf{K}_{\text{nt}} = \mathbf{K}_n - (\mathbf{K}_n \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$ ,  $\mathbf{K}_n = \mathbf{K}\mathbf{n}_{\text{wall}}$ , and  $\mathbf{K}$  is the viscous stress tensor.  $k_1$  (SI unit: m/s) and  $k_2$  (SI unit: 1/Pa) are the slip-velocity and compliance coefficients.

In cases where the wall movement is nonzero, check **Account for the translational wall velocity in the friction force** to set the slip velocity relative to the boundary velocity  $\mathbf{u}_{\text{bnd}}$ . Note that the **Velocity of sliding wall**  $\mathbf{u}_w$  is always accounted for in the boundary velocity.

The **Hatzikiriakos Slip** option is not available for the Viscoelastic Flow interface.

#### *Asymptotic Slip*

This boundary condition enforces no-penetration at the wall,  $\mathbf{u} \cdot \mathbf{n}_{\text{wall}} = 0$ , and adds a tangential stress

$$\mathbf{K}_{\text{nt}} = -\frac{1}{k_2}(e^{\|\mathbf{u}_{\text{slip}}\|/k_1} - 1) \frac{\mathbf{u}_{\text{slip}}}{\|\mathbf{u}_{\text{slip}}\|}$$

where  $\mathbf{K}_{\text{nt}} = \mathbf{K}_n - (\mathbf{K}_n \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$ ,  $\mathbf{K}_n = \mathbf{K}\mathbf{n}_{\text{wall}}$ , and  $\mathbf{K}$  is the viscous stress tensor.  $k_1$  (SI unit: m/s) and  $k_2$  (SI unit: 1/Pa) are the slip-velocity and compliance coefficients, while  $\mathbf{u}_{\text{slip}} = \mathbf{u} - (\mathbf{u} \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$  is the velocity tangential to the wall.

In cases where the wall movement is nonzero, check **Account for the translational wall velocity in the friction force** to use  $\mathbf{u} - \mathbf{u}_{\text{bnd}} - ((\mathbf{u} - \mathbf{u}_{\text{bnd}}) \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$  instead of  $\mathbf{u}_{\text{slip}}$  in the friction force.

The **Asymptotic Slip** option is not available for the Viscoelastic Flow interface.

## WALL MOVEMENT

This section contains controls to describe the wall movement relative to the lab (or spatial) frame.

The **Translational velocity** setting controls the translational wall velocity,  $\mathbf{u}_{tr}$ . The list is per default set to **Automatic from frame**. The physics automatically detects if the spatial frame moves. This can for example happen if an ALE interface is present in the model component. If there is no movement  $\mathbf{u}_{tr} = \mathbf{0}$ . If the frame moves,  $\mathbf{u}_{tr}$  becomes equal to the frame movement.  $\mathbf{u}_{tr}$  is accounted for in the actual boundary condition prescribed in the **Boundary condition** section.


Select **Zero (Fixed wall)** from **Translational velocity** selection list to prescribe  $\mathbf{u}_{tr} = \mathbf{0}$ .

Select **Manual** from **Translational velocity** selection list in order to manually prescribe **Velocity of moving wall**,  $\mathbf{u}_{tr}$ . This can for example be used to model an oscillating wall where the magnitude of the oscillations are very small compared to the rest of the model. Specifying translational velocity manually does not automatically cause the associated wall to move. An additional Moving Mesh node needs to be added from Definitions to physically track the wall movement in the spatial reference frame.

The **Sliding wall** option is appropriate if the wall behaves like a conveyor belt with the surface sliding in a tangential direction. A velocity is prescribed at the wall and the boundary itself does not have to actually move relative to the reference frame.

- For 3D components, values or expressions for the **Velocity of sliding wall**  $\mathbf{u}_w$  should be specified. If the velocity vector entered is not in the plane of the wall, COMSOL Multiphysics projects it onto the tangential direction. Its magnitude is adjusted to be the same as the magnitude of the vector entered.
- For 2D components, the tangential direction is unambiguously defined by the direction of the boundary. For this reason, the sliding wall boundary condition has different definitions in different space dimensions. A single entry for the **Velocity of the tangentially moving wall**  $U_w$  should be specified in 2D.
- For 2D axisymmetric components when **Swirl flow** is selected in the physics interface properties, the **Velocity of moving wall**,  $\phi$ -component  $v_w$  can also be specified.


## CONSTRAINT SETTINGS

This section is displayed by clicking the **Show More Options** button (  ) and selecting **Advanced Physics Options** in the **Show More Options** dialog box. The **Constraints** settings can be set to **Automatic**, **Pointwise constraints**, **Nitsche constraints**, or **Weak constraints**. **Mixed constraints** can be selected when imposing a no slip condition exactly.



Depending on the constraint method selected, the following settings are available:

- **Apply reaction terms on** can be set to **Individual dependent variables** (default) or **All physics (symmetric)**. This setting is not available when **Nitsche constraints** is selected.
- Select **Elemental** (default) or **Nodal** under **Constraint method**. This setting is not available for **Nitsche constraints** or **Weak constraints**.

	<p>In the <i>COMSOL Multiphysics Reference Manual</i>:</p> <ul style="list-style-type: none"> <li>• <a href="#">Theory for the Wall Boundary Condition</a></li> <li>• <a href="#">The Moving Mesh Interface</a></li> </ul>
---	--

## *Inlet*

This condition should be used on boundaries for which there is a net flow into the domain. To obtain a numerically well-posed problem, it is advisable to also consider the Outlet conditions when specifying an Inlet condition. For example, if the pressure is specified at the outlet, the velocity may be specified at the inlet, and vice versa. Specifying the velocity vector at both the inlet and the outlet may cause convergence difficulties.

### **BOUNDARY CONDITION**

The available **Boundary condition** options for an inlet are **Velocity**, **Fully developed flow**, **Mass flow**, and **Pressure**. After selecting a **Boundary Condition** from the list, a section with the same or a similar name displays underneath. For example, if **Velocity** is selected, a **Velocity** section, where further settings are defined, is displayed.

### **VELOCITY**

The **Normal inflow velocity** is specified as  $\mathbf{u} = -\mathbf{n}U_0$ , where  $\mathbf{n}$  is the boundary normal pointing out of the domain and  $U_0$  is the normal inflow speed.

The **Velocity field** option sets the velocity vector to  $\mathbf{u} = \mathbf{u}_0$ . The components of the inlet velocity vector  $\mathbf{u}_0$  should be defined for this choice.

## PRESSURE CONDITIONS

This option specifies the normal stress, which in most cases is approximately equal to the pressure.

- For incompressible flow, the **Pressure** list has two options, **Static** and **Total**. For weakly compressible and compressible flow, the static pressure should be specified in the text field.
  - If **Pressure** is **Static**, and the reference pressure  $p_{\text{ref}}$ , defined at the physics interface level, is equal to 0, the value of the pressure  $p_0$ , at the boundary, is the absolute pressure. Otherwise,  $p_0$  is the relative pressure at the boundary.
  - If **Pressure** is **Total**, the **Average** check box is available and cleared by default to prescribe the total pressure pointwise. If it is selected, the averaged total pressure is imposed in the weak forms instead.

When **Include gravity** is selected and **Use reduced pressure** not selected in the interface **Physical model** section, the **Compensate for hydrostatic pressure approximation** (named **Compensate for hydrostatic pressure** for incompressible flows) check box is available and selected by default. When it is selected, the hydrostatic pressure is automatically added to the pressure entered in  $p_0$  user input.

- The **Suppress backflow** option adjusts the inlet pressure locally in order to reduce the amount of fluid exiting the domain through the boundary. If you clear the suppress backflow option, the inlet boundary can become an outlet depending on the pressure field in the rest of the domain.
- **Flow direction** controls in which direction the fluid enters the domain.
  - For **Normal flow**, it prescribes zero tangential velocity component.
  - For **User defined**, an **Inflow velocity direction**  $\mathbf{d}_u$  (dimensionless) should be specified. The magnitude of  $\mathbf{d}_u$  does not matter, only the direction.  $\mathbf{d}_u$  must point into the domain.

## MASS FLOW

The mass flow at an inlet can be specified by the **Mass flow rate** or a **Pointwise mass flux**.

The **Apply condition on each disjoint selection separately** check box is selected per default. When this setting is selected, the mass flow condition is applied separately on each disjoint selection. If this option is not selected, the condition is applied over the whole feature selection. The **Apply condition on each disjoint selection separately** should be disabled only if the flow conditions are known to be identical on each disjoint boundary.

### Mass Flow Rate

The **Mass flow rate** option sets the integrated mass flow over the boundary selection, the **Normal mass flow rate** to a specific value,  $m$ . The mass flow is assumed to be parallel to the boundary normal, and the tangential flow velocity is set to zero.

For 2D components, the **Channel thickness**  $d_{bc}$  is used to define the area across which the mass flow occurs. This setting is not applied to the whole model. Line or surface integrals of the mass flow over the boundary evaluated during postprocessing or used in integration coupling operators do not include this scaling automatically. Such results should be appropriately scaled when comparing them with the specified mass flow.

### Pointwise Mass Flux

The **Pointwise mass flux** sets the mass flow parallel to the boundary normal. The tangential flow velocity is set to zero. The mass flux is a model input, which means that COMSOL Multiphysics can take its value from another physics interface when available. When **User defined** is selected a value or function  $M_f$  should be specified for the **Mass flux**.

## FULLY DEVELOPED FLOW

The **Fully developed flow** option adds contributions to the inflow boundary, which force the flow toward the solution for a fully developed channel flow. The channel can be thought of as a virtual extrusion of the inlet cross section. The inlet boundary must hence be flat in order for the fully developed flow condition to work properly. In 2D axisymmetric models, the inlet normal must be parallel to the symmetry axis.

Select an option to control the flow rate at the inlet:

- **Average velocity**,  $U_{av}$ .
- **Flow rate**,  $V_0$ . Two-dimensional models also require an **Entrance thickness**,  $D_z$ , which is the out-of-plane thickness of the extruded entrance channel.
- **Average pressure**,  $P_{av}$ . Note that  $P_{av}$  is the average pressure on the inflow boundary.

The **Apply condition on each disjoint selection separately** check box is selected per default. When this setting is selected, the fully developed flow condition is applied separately on each disjoint selection. If this option is not selected, the condition is applied over the whole feature selection. The **Apply condition on each disjoint selection separately** should be disabled only if the flow conditions are known to be identical on each disjoint boundary.

The fully developed flow condition requires any volume force to be approximately aligned with the normal of the inlet boundary. The exception is gravity when the


**Include gravity** setting is selected in the physics interface settings. Unless **Use reduced pressure** is also selected, an option to **Compensate for hydrostatic pressure** or **Compensate for hydrostatic pressure approximation** becomes available. It is selected per default and should only be deselected if the inlet normal is aligned with the gravity force and you want to specify an average pressure that includes the hydrostatic pressure.



For the Viscoelastic Flow interface, the fully developed inflow can be used in conjunction with the viscoelastic constitutive model. No additional inputs are required for the elastic stress tensor components. They are instead solved for on the inlet boundary to be consistent with the fully developed flow profile.

### VISCOELASTIC STRESS

For the Viscoelastic Flow interface, the inlet conditions for the elastic stress components are required. Note, that if several branches are specified in the **Fluid Property** node, the initial values entered above are applied to all branches.

### CONSTRAINT SETTINGS

This section is displayed by clicking the **Show** button (  ) and selecting **Advanced Physics Options**.

	<p>In the <i>COMSOL Multiphysics Reference Manual</i>:</p> <ul style="list-style-type: none"> <li>• <a href="#">Prescribing Inlet and Outlet Conditions</a></li> <li>• <a href="#">Normal Stress Boundary Condition</a></li> </ul>
	<p><a href="#">Fully Developed Flow (Inlet)</a></p>

### Outlet

This condition should be used on boundaries for which there is a net outflow from the domain. To obtain a numerically well-posed problem, it is advisable to also consider the Inlet conditions when specifying an Outlet condition. For example, if the velocity is specified at the inlet, the pressure may be specified at the outlet, and vice versa. Specifying the velocity vector at both the inlet and the outlet may cause convergence difficulties. Selecting appropriate outlet conditions for the Navier–Stokes equations is a nontrivial task. Generally, if there is something interesting happening at an outflow

boundary, the computational domain should be extended to include this phenomenon.

## BOUNDARY CONDITION

The available **Boundary condition** options for an outlet are **Pressure**, **Fully developed flow**, and **Velocity**.

### PRESSURE CONDITIONS

This option specifies the normal stress, which in most cases is approximately equal to the pressure. The tangential stress component is set to zero.

- For incompressible flow, the **Pressure** list has two options, **Static** and **Total**. For weakly compressible and compressible flow, the static pressure should be specified in the text field.
  - If **Pressure** is **Static**, and the reference pressure  $p_{\text{ref}}$ , defined at the physics interface level, is equal to 0, the value of the pressure  $p_0$ , at the boundary, is the absolute pressure. Otherwise,  $p_0$  is the relative pressure at the boundary.
  - If **Pressure** is **Total**, the **Average** check box is available and selected by default to prescribe the averaged total pressure in the weak forms. If it is cleared, the total pressure is imposed pointwise.

When **Include gravity** is selected and **Use reduced pressure** not selected in the interface **Physical model** section, the **Compensate for hydrostatic pressure approximation** (named **Compensate for hydrostatic pressure** for incompressible flows) check box is available and selected by default. When it is selected, the hydrostatic pressure is automatically added to the pressure entered in  $p_0$  user input.

- The **Normal flow** option changes the no tangential stress condition to a no tangential velocity condition. This forces the flow to exit (or enter) the domain perpendicularly to the outlet boundary.
- The **Suppress backflow** check box is selected by default. This option adjusts the outlet pressure in order to reduce the amount of fluid entering the domain through the boundary.

### VELOCITY

See the **Inlet** node [Velocity](#) section for the settings.

### FULLY DEVELOPED FLOW

This boundary condition is applicable when the flow exits the domain into a long pipe or channel, at the end of which a flow profile is fully developed.

The channel can be thought of as a virtual extrusion of the outlet cross section. The outlet boundary must hence be flat in order for the fully developed flow condition to work properly. In 2D axisymmetric models, the outlet normal must be parallel to the symmetry axis.

Select an option to control the flow rate at the outlet:


- **Average velocity**,  $U_{av}$ .
- **Flow rate**,  $V_0$ . Two-dimensional models also require an **Entrance thickness**,  $D_z$ , which is the out-of-plane thickness of the extruded entrance channel.
- **Average pressure**,  $P_{av}$ . Observe that  $P_{av}$  is the average pressure on the outflow.

The **Apply condition on each disjoint selection separately** check box is selected per default. When this setting is selected, the fully developed flow condition is applied separately on each disjoint selection. If this option is not selected, the condition is applied over the whole feature selection. The **Apply condition on each disjoint selection separately** should be disabled only if the flow conditions are known to be identical on each disjoint boundary.

The fully developed flow condition requires any volume force to be approximately aligned with the normal of the outlet boundary. The exception is gravity when the **Include gravity** setting is selected in the physics interface settings. Unless **Use reduced pressure** is also selected, an option to **Compensate for hydrostatic pressure** or **Compensate for hydrostatic pressure approximation** becomes available. It is selected per default and should only be deselected if the outlet normal is aligned with the gravity force and you want to specify an average pressure that includes the hydrostatic pressure.

For the Viscoelastic Flow interface, the fully developed outflow can be used in conjunction with the viscoelastic constitutive model. No additional inputs are required for the elastic stress tensor components. They are instead solved for on the outlet boundary to be consistent with the fully developed flow profile.

## CONSTRAINT SETTINGS

This section is displayed by clicking the **Show** button (  ) and selecting **Advanced Physics Options**.



Prescribing Inlet and Outlet Conditions in the *COMSOL Multiphysics Reference Manual*



### *Symmetry*

This node provides a boundary condition for symmetry boundaries. It should only be used when the geometry and expected solution have mirror symmetry. By using symmetries in a model its size can be reduced by one-half or more, making this an efficient tool for solving large problems.

The **Symmetry** boundary condition prescribes no penetration and vanishing shear stresses. The boundary condition is a combination of a Dirichlet condition and a Neumann condition:

$$\mathbf{u} \cdot \mathbf{n} = 0, \quad \left( -p\mathbf{I} + \left( \mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I} \right) \right) \mathbf{n} = \mathbf{0}$$

$$\mathbf{u} \cdot \mathbf{n} = 0, \quad (-p\mathbf{I} + \mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T))\mathbf{n} = \mathbf{0}$$

for the compressible and incompressible formulations. The Dirichlet condition takes precedence over the Neumann condition, and the above equations are equivalent to the following equation for both the compressible and incompressible formulations:


$$\mathbf{u} \cdot \mathbf{n} = 0, \quad \mathbf{K} - (\mathbf{K} \cdot \mathbf{n})\mathbf{n} = \mathbf{0}$$

$$\mathbf{K} = \mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T)\mathbf{n}$$

#### **BOUNDARY SELECTION**

For 2D axial symmetry, a boundary condition does not need to be defined for the symmetry axis at  $r = 0$ . The software automatically provides a condition that prescribes  $u_r = 0$  and vanishing stresses in the  $z$  direction and adds an **Axial Symmetry** node that implements these conditions on the axial symmetry boundaries only.

#### **CONSTRAINT SETTINGS**

This section is displayed by clicking the **Show** button (  ) and selecting **Advanced Physics Options**.

## Open Boundary

---

The **Open Boundary** condition describes boundaries in contact with a large volume of fluid. Fluid can both enter and leave the domain on boundaries with this type of condition.

### BOUNDARY CONDITIONS

The **Boundary condition** options for open boundaries are **Normal stress** and **No viscous stress**.

#### Normal Stress

The **Normal stress**  $f_0$  condition implicitly imposes  $p \approx f_0$ .

When **Include gravity** is selected and **Use reduced pressure** not selected in the interface **Physical model** section, the **Compensate for hydrostatic pressure approximation** (named **Compensate for hydrostatic pressure** for incompressible flows) check box is available and selected by default. When it is selected, the hydrostatic pressure is automatically added to the pressure entered in  $f_0$  user input.

#### No Viscous Stress

The **No Viscous Stress** condition specifies vanishing viscous stress on the boundary. This condition does not provide sufficient information to fully specify the flow at the open boundary and must at least be combined with pressure constraints at adjacent points.

The **No viscous stress** condition prescribes:

$$\left( \mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I} \right) \mathbf{n} = \mathbf{0}$$
$$\mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T) \mathbf{n} = \mathbf{0}$$

for the compressible and the incompressible formulations. This condition can be useful in some situations because it does not impose any constraint on the pressure. A typical example is a model with volume forces that give rise to pressure gradients that are hard to prescribe in advance. To make the model numerically stable, this boundary condition should be combined with a point constraint on the pressure.

### VISCOELASTIC STRESS

For Viscoelastic Flow interface, **Viscoelastic Stress** section is available. See [Viscoelastic Stress](#).



## Boundary Stress

The **Boundary Stress** node adds a boundary condition that represents a general class of conditions also known as traction boundary conditions.

### BOUNDARY CONDITION

The **Boundary condition** options for the boundary stress are **General stress**, **Normal stress**, and **Normal stress, normal flow**.

#### General Stress

When **General stress** is selected, the components for the **Stress**  $\mathbf{F}$  should be specified. The total stress on the boundary is set equal to the given stress  $\mathbf{F}$ :

$$\left(-p\mathbf{I} + \left(\mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I}\right)\right)\mathbf{n} = \mathbf{F}$$

$$(-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T))\mathbf{n} = \mathbf{F}$$

for the compressible and the incompressible formulations.

This boundary condition implicitly sets a constraint on the pressure that for 2D flows is

$$p = 2\mu \frac{\partial u_n}{\partial n} - \mathbf{n} \cdot \mathbf{F} \quad (3-1)$$

If  $\partial u_n / \partial n$  is small, [Equation 3-1](#) states that  $p \approx -\mathbf{n} \cdot \mathbf{F}$ .

#### Normal Stress

[Normal Stress](#) is described for the Open Boundary node.

#### Normal Stress, Normal Flow

For **Normal stress, normal flow**, the magnitude of the **Normal stress**  $f_0$  should be specified. The tangential velocity is set to zero on the boundary:

$$\mathbf{n}^T \left(-p\mathbf{I} + \left(\mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I}\right)\right)\mathbf{n} = -f_0, \quad \mathbf{t} \cdot \mathbf{u} = 0$$

$$\mathbf{n}^T (-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T))\mathbf{n} = -f_0, \quad \mathbf{t} \cdot \mathbf{u} = 0$$

for the compressible and the incompressible formulations.

This boundary condition implicitly sets a constraint on the pressure that for 2D flows is

$$p = 2\mu \frac{\partial u_n}{\partial n} + f_0 \quad (3-2)$$


If  $\partial u_n / \partial n$  is small, [Equation 3-2](#) states that  $p \approx f_0$ .

When **Include gravity** is selected and **Use reduced pressure** not selected in the interface **Physical model** section, the **Compensate for hydrostatic pressure approximation** (named **Compensate for hydrostatic pressure** for incompressible flows) check box is available and selected by default. When it is selected, the hydrostatic pressure is automatically added to the pressure entered in  $f_0$  or  $p_{\text{hydro}}$  is added to  $\mathbf{F}$  depending of the selected option.

### VISCOELASTIC STRESS

For Viscoelastic Flow interface, **Viscoelastic Stress** section is available. See [Viscoelastic Stress](#).

### CONSTRAINT SETTINGS

This section is displayed by clicking the **Show** button (  ) and selecting **Advanced Physics Options**.

If [Normal Stress](#), [Normal Flow](#) is selected as the **Boundary condition**, then to **Apply reaction terms on** all dependent variables, the **All physics (symmetric)** option should be selected. Alternatively, the **Individual dependent variables** could be selected to restrict the reaction terms as needed.

### *Periodic Flow Condition*

The **Periodic Flow Condition** splits its selection into a source group and a destination group (the **Destination Selection** section is optional). Fluid that leaves the domain through one of the destination boundaries enters the domain through the corresponding source boundary. This corresponds to a situation where the geometry is a periodic part of a larger geometry. If the boundaries are not parallel to each other, the velocity vector is automatically transformed.

If the boundaries are curved, the orientation of the source must be specified manually (see [Orientation of Source](#)).

No input is required when **Compressible flow (Ma<0.3)** is selected for **Compressibility** under the **Physical Model** section for the physics interface. Typically when a periodic boundary condition is used with a compressible flow, the pressure is the same at both boundaries and the flow is driven by a volume force.

## BOUNDARY SELECTION



The software usually automatically identifies the boundaries as either source boundaries or destination boundaries, as indicated in the selection list. This works fine for cases like opposing parallel boundaries. In other cases, right-click **Periodic Flow Condition** and select **Manual Destination Selection** to control the destination. By default it contains the selection that COMSOL Multiphysics identifies.

## DESTINATION SELECTION

This section is available for specifying the destination boundaries, if needed, when the **Manual Destination Selection** option is selected in the context menu for the **Periodic Flow Condition** node. You can only select destination boundaries from the union of all source and destination boundaries.


## FLOW CONDITION

This section is available when **Incompressible flow** is selected for **Compressibility** under the **Physical Model** section for the physics interface.

The **Flow Condition** at the boundary is specified through a value or expression for either the **Pressure difference**,  $p_{\text{src}} - p_{\text{dst}}$ , or the **Mass flow**,  $\dot{m}$ . The mass flow option ensures a pressure difference is achieved across the source and destination boundaries. This pressure difference can, for example, drive the fully developed flow in a channel.

To set up a periodic boundary condition, both boundaries must be selected in the **Periodic Flow Condition** node. COMSOL Multiphysics automatically assigns one boundary as the source and the other as the destination. To manually set the destination selection, a **Destination Selection** subnode is available from the context menu (by right-clicking the parent node) or from the **Physics** toolbar, **Attributes** menu. All destination sides must be connected.

## ORIENTATION OF SOURCE


To display this section, click the **Show More Options** button (  ) and select **Advanced Physics Options** in the **Show More Options** dialog box. For information about the **Orientation of Source** section, see [Orientation of Source and Destination](#) in the *COMSOL Multiphysics Reference Manual*.

## ORIENTATION OF DESTINATION

This section appears if the setting for **Transform to intermediate map** in the **Orientation of Source** section is changed from the default value, **Automatic**, and **Advanced Physics**

**Options** is selected in the **Show More Options** dialog box. For information about the **Orientation of Destination** section, see [Orientation of Source and Destination](#) in the *COMSOL Multiphysics Reference Manual*.

## MAPPING BETWEEN SOURCE AND DESTINATION

To display this section, click the **Show More Options** button (  ) and select **Advanced Physics Options** in the **Show More Options** dialog box. For information about the **Mapping Between Source and Destination** section, see [Mapping Between Source and Destination](#) in the *COMSOL Multiphysics Reference Manual*.

## *Interior Wall*

---

The **Interior Wall** boundary condition can only be applied on interior boundaries.

It is similar to the [Wall](#) boundary condition available on exterior boundaries except that it applies on both sides of an interior boundary. It allows discontinuities (velocity, pressure) across the boundary. The **Interior Wall** boundary condition can be used to avoid meshing thin structures by applying no-slip conditions on interior curves and surfaces instead. Slip conditions and conditions for a moving wall can also be prescribed. It is compatible with laminar and turbulent flow.

## BOUNDARY CONDITION

The following **Boundary condition** options are available.

### *No Slip*

The **No slip** condition models solid walls. No slip walls are walls where the fluid velocity relative to the wall is zero. For an interior stationary wall this means that  $\mathbf{u} = 0$  on both sides of the wall.

### *Slip*

The **Slip** condition prescribes a no-penetration condition,  $\mathbf{u} \cdot \mathbf{n} = 0$ . It implicitly assumes that there are no viscous effects on either side of the slip wall and hence, no boundary layer develops. From a modeling point of view, this can be a reasonable approximation if the important effect is to prevent the exchange of fluid between the regions separated by the interior wall.

### *Navier Slip*

The **Navier slip** condition enforces no-penetration at the wall,  $\mathbf{u} \cdot \mathbf{n}_{\text{wall}} = 0$ , and adds a tangential stress on each side of the wall

$$\mathbf{K}_{\text{nt}} = -\frac{\mu}{\beta} \mathbf{u}_{\text{slip}}$$

where  $\beta$  is a slip length. For more information, see the **Navier Slip** option in the [Wall](#) feature.

#### *Nonlinear Navier Slip*

This boundary condition enforces no-penetration at the wall,  $\mathbf{u} \cdot \mathbf{n}_{\text{wall}} = 0$ , and adds a tangential stress on each side of the wall

$$\mathbf{K}_{\text{nt}} = -\left(\frac{\tau_{\text{ref}}}{\|\mathbf{K}_{\text{nt}}\|}\right)^{m-1} \frac{\mu}{\beta} \mathbf{u}_{\text{slip}}$$

For more information, see the **Nonlinear Navier Slip** option in the [Wall](#) feature.

#### *Hatzikiriakos Slip*

When Hatzikiriakos slip is selected, a slip velocity is applied on each side of the interior wall when the tangential stress on the corresponding side is larger than the specified yield stress,  $\tau_y$ ,

$$\mathbf{u}_{\text{slip}} = \begin{cases} k_1 \sinh([k_2(\|\mathbf{K}_{\text{nt}}\| - \tau_y)]) \frac{\mathbf{K}_{\text{nt}}}{\|\mathbf{K}_{\text{nt}}\|}, & \text{if } \|\mathbf{K}_{\text{nt}}\| > \tau_y \\ 0, & \text{if } \|\mathbf{K}_{\text{nt}}\| \leq \tau_y \end{cases}$$

For more information, see the **Hatzikiriakos Slip** option in the [Wall](#) feature.

The **Hatzikiriakos Slip** option is not available for the Viscoelastic Flow interface.

#### *Asymptotic Slip*

This boundary condition enforces no-penetration at the wall,  $\mathbf{u} \cdot \mathbf{n}_{\text{wall}} = 0$ , and adds a tangential stress on each side of the wall

$$\mathbf{K}_{\text{nt}} = -\frac{1}{k_2} (e^{\|\mathbf{u}_{\text{slip}}\|/k_1} - 1) \frac{\mathbf{u}_{\text{slip}}}{\|\mathbf{u}_{\text{slip}}\|}$$

For more information, see the **Asymptotic Slip** option in the [Wall](#) feature.

The **Asymptotic Slip** option is not available for the Viscoelastic Flow interface.

## WALL MOVEMENT


This section contains controls to describe the wall movement relative to the lab (or spatial) frame.

The **Translational velocity** setting controls the translational wall velocity,  $\mathbf{u}_{tr}$ . The list is per default set to **Automatic from frame**. The physics automatically detects if the spatial frame moves. This can for example happen if an ALE interface is present in the model component. If there is no movement  $\mathbf{u}_{tr} = \mathbf{0}$ . If the frame moves,  $\mathbf{u}_{tr}$  becomes equal to the frame movement.  $\mathbf{u}_{tr}$  is accounted for in the actual boundary condition prescribed in the **Boundary condition** section.

Select **Zero (Fixed wall)** from **Translational velocity** selection list to prescribe  $\mathbf{u}_{tr} = \mathbf{0}$ .

Select **Manual** from **Translational velocity** selection list in order to manually prescribe **Velocity of moving wall**,  $\mathbf{u}_{tr}$ . This can for example be used to model an oscillating wall where the magnitude of the oscillations are very small compared to the rest of the model. Specifying translational velocity manually does not automatically cause the associated wall to move. An additional Moving Mesh interface needs to be added to physically track the wall movement in the spatial reference frame.

## CONSTRAINT SETTINGS

This section is displayed by clicking the **Show** button (  ) and selecting **Advanced Physics Options**. The **Constraints** settings can be set to **Automatic**, **Pointwise constraints**, or **Nitsche constraints**. **Mixed constraints** can be selected when imposing a no slip condition exactly.

### *Porous Interface*

---

The **Porous Interface** condition is designed to properly account for a sudden jump in porosity,  $\epsilon_p$ , across a boundary between different domains. It describes possible discontinuity of stress across the interface, and also reveals modified behavior of tangential velocity due to boundary layer on the upstream side of the interface. Choose among the following options under the **Continuity conditions**:

- **Velocity and stress** — stress continuity is invoked;
- **Velocity and porosity-corrected stress** (default)— the sum of stress and a contribution due to the inertial term being constant across the interface;
- **Velocity and porosity-corrected stress with loss** — the sum of the stress and a contribution due to the inertial term is lost in the downstream direction.

All three options give continuity of the velocity across the interface in constant-density flows, while in case of a jump in density:

- 1 give continuity of the mass flux (product of density and normal velocity) and tangential velocity;
- 2 account for the density-jump correction in the stress condition.

When **Velocity and porosity-corrected stress with loss** is chosen, a **Pressure loss model** needs to be specified. The options are:

- **User-defined loss coefficients** (default). Enter values for **Expansion loss coefficient** and **Contraction loss coefficient**;
- **Borda–Carnot**. Specify **Vena contracta model**.

**Vena contracta model** allows two options:

- **User defined** (default) — enter value of **Vena contracta coefficient**;
- **Weisbach** — enter values of **Lower limit** and **Exponent**.

Notice that the notions ‘expansion’ and ‘contraction’ used in this section correspond not to geometrical changes, but to the variations of the microscopic structure across the interface in the flow direction — increasing or decreasing porosity,  $\varepsilon_p$ , respectively.

### *Flow Continuity*

---

The **Flow Continuity** condition is suitable for pairs where the boundaries match; it prescribes that the flow field is continuous across the pair.

The parts of the boundary where a source boundary lacks a corresponding destination boundary and vice versa will get conditions from other boundary conditions under the physics interface acting as fallback boundary conditions on the nonoverlapping parts. By default, the [Wall](#) node is used as fallback unless another feature has been selected for the boundary.

When the condition is set for pairs adjacent to domains with different porosity  $\varepsilon_p$ , it imposes conditions identical to the **Porous Interface** feature, which is designed to properly account for a sudden jump in porosity. See the section [Porous Interface](#) above

for further details (note that the **Flow Continuity** does not account for the density-jump corrections),



### *Pressure Point Constraint*

---


The **Pressure Point Constraint** condition can be used to specify the pressure level. If it is not possible to specify the pressure level using a boundary condition, the pressure level must be set in some other way, for example, by specifying a fixed pressure at a point.

#### **PRESSURE CONSTRAINT**

The relative pressure value is set by specifying the **Pressure**  $p_0$ . If the reference pressure  $p_{\text{ref}}$  defined at the physics interface level is equal to zero,  $p_0$  represents the absolute pressure.

When **Include gravity** is selected and **Use reduced pressure** not selected in the interface **Physical model** section, the **Compensate for hydrostatic pressure approximation** (named **Compensate for hydrostatic pressure** for incompressible flows) check box is available and selected by default. When it is selected, the hydrostatic pressure is automatically added to the pressure entered in  $p_0$  user input.

#### **CONSTRAINT SETTINGS**

This section is displayed by clicking the **Show** button (  ) and selecting **Advanced Physics Options**.

The **Apply reaction terms on** setting is set per default to **Individual dependent variables**. **All physics (symmetric)** cannot be used together with a segregated solver when fluid flow is coupled with Moving Mesh, Level Set, or Phase Field in Fluids.

### *Point Mass Source*

---

This feature requires at least one of the following licenses: Battery Design Module, CFD Module, Chemical Reaction Engineering Module, Corrosion Module, Electrochemistry Module, Electrodeposition Module, Fuel Cell & Electrolyzer Module, Microfluidics Module, Pipe Flow Module, or Subsurface Flow Module.



The **Point Mass Source** feature models mass flow originating from an infinitely small domain centered around a point.

For the Reacting Flow in Porous Media, Diluted Species interface, which is available with the CFD Module, Chemical Reaction Engineering Module, or Battery Design Module, there are two nodes: one for the fluid flow (**Fluid Point Source**) and one for the species (**Species Point Source**).

**SOURCE STRENGTH**

The source **Mass flux**,  $\dot{q}_p$ , should be specified. A positive value results in mass being ejected from the point into the computational domain. A negative value results in mass being removed from the computational domain.

Point sources located on a boundary or on an edge affect the adjacent computational domains. This has the effect, for example, that a point source located on a symmetry plane has twice the given strength.



[Mass Sources for Fluid Flow](#) in the *COMSOL Multiphysics Reference Manual*

*Line Mass Source*

This feature requires at least one of the following licenses: Battery Design Module, CFD Module, Chemical Reaction Engineering Module, Corrosion Module, Electrochemistry Module, Electrodeposition Module, Fuel Cell & Electrolyzer Module, Microfluidics Module, Pipe Flow Module, or Subsurface Flow Module.

The **Line Mass Source** feature models mass flow originating from a tube region with infinitely small radius.

For the Reacting Flow in Porous Media, Diluted Species interface, which is available with the CFD Module, Chemical Reaction Engineering Module, or Battery Design Module, there are two nodes, one for the fluid flow (**Fluid Line Source**) and one for the species (**Species Line Source**).

## SELECTION

The **Line Mass Source** feature is available for all dimensions, but the applicable selection differs between the dimensions.

MODEL DIMENSION	APPLICABLE GEOMETRICAL ENTITY
2D	Points
2D Axisymmetry	Points not on the axis of symmetry
3D	Edges

## SOURCE STRENGTH

The source **Mass flux**,  $\dot{q}_1$ , should be specified. A positive value results in mass being ejected from the line into the computational domain and a negative value means that mass is removed from the computational domain.

Line sources located on a boundary affect the adjacent computational domains. This, for example, has the effect that a line source located on a symmetry plane has twice the given strength.



[Mass Sources for Fluid Flow](#) in the *COMSOL Multiphysics Reference Manual*

## Gravity

The **Gravity** global feature is automatically added when **Include gravity** is selected at the interface level in the **Physical Model** settings. It defines the gravity forces from the **Acceleration of gravity** value.

## ACCELERATION OF GRAVITY

The **Acceleration of gravity** (SI unit m/s, default value  $-g_{\text{const}}\mathbf{e}_z$  in 2D axial symmetry and 3D and  $-g_{\text{const}}\mathbf{e}_y$  in 2D) is used to define the gravity forces. It should be a global quantity.


# The Potential Flow Interface

In this section

- [The Incompressible Potential Flow Interface](#)

## *The Incompressible Potential Flow Interface*

---

The **Incompressible Potential Flow (ipf)** interface () is used for simulating irrotational, inviscid, and incompressible flow. The main application of the interface is to provide initial solutions for other Single-Phase Flow interfaces.

The equation solved by the Incompressible Potential Flow interface is the Laplace equation for a velocity potential. This can be used to derive both a velocity and static pressure.

The Incompressible Potential Flow interface is available for stationary analyses. The main feature is Fluid Properties, which adds the Laplace equation. The Fluid Properties feature provides an interface for defining the fluid material and its properties used to derive the velocity and pressure. When this physics interface is added, the following default nodes are also added in the **Model Builder**:

- **Fluid Properties**
- **Initial Values**
- **Wall**

Then, from the **Physics** toolbar, add other nodes that implement boundary conditions. You can also right-click **Incompressible Potential Flow** 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 ipf.

## PRESSURE

This node specifies **Total pressure** (SI unit: Pa) used to compute the pointwise static pressure from the potential flow solution.



## DEPENDENT VARIABLES

This physics interface solves for the **Velocity Potential**  $\phi$ -dependent variable (field).

If required, the names of the field, component, and dependent variable can be edited. Editing the name of a scalar dependent variable changes both its field name and the dependent variable name. If a new field name coincides with the name of another field of the same type, the fields share degrees of freedom and dependent variable names. A new field name must not coincide with the name of a field of another type or with a component name belonging to some other field. Component names must be unique within a model except when two fields share a common field name.

## DISCRETIZATION

Select shape function type for the velocity potential. The default setting is **Quadratic**.

	<ul style="list-style-type: none"><li>• <a href="#">Domain, Boundary, Point, and Pair Nodes for the Viscoelastic Interface</a></li><li>• <a href="#">Theory for the Incompressible Potential Flow Interface</a></li></ul>
	<p><i>Flow Around an Inclined NACA 0012 Airfoil:</i> Application Library path <b>CFD_Module/Verification_Examples/naca0012_airfoil</b></p>

*Domain, Boundary, Point, and Pair Nodes for the Incompressible Potential Flow Interface*

The [The Incompressible Potential Flow Interface](#) has the following domain, boundary and pair nodes:

- [Initial Values](#)
- [Interior Wall](#)
- [Flow Continuity](#)
- [Fluid Properties](#)
- [Open Boundary](#)
- [Periodic Flow Condition](#) (see [Periodic Boundary Conditions](#) in the *COMSOL Multiphysics Reference Manual*)
- [Potential Point Constraint](#)
- [Symmetry](#)
- [Velocity](#)
- [Wall](#)

<sup>1</sup> Described for the Laminar Flow interface.



Boundary conditions for axial symmetry boundaries are not required. For the symmetry axis at  $r = 0$ , the software automatically provides a suitable boundary condition and adds an **Axial Symmetry** node that is valid on the axial symmetry boundaries only.



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.

*Fluid Properties*

The **Fluid Properties** node adds the Laplace equation solved by the physics interface. The node also provides an interface for defining the density of the fluid, and builds the velocity and static pressure.

*Material Properties*

The density can either be specified by a material, or by a **User defined** expression.

## *Initial Values*

---

Use the **Initial Values** node to define the initial values of the velocity potential variable.

### **INITIAL VALUES**

An initial value or expression should be specified for the **Velocity Potential**  $\phi$ .

## *Wall*

---

The **Wall** node sets a slip wall condition for the fluid. This prescribes a no-penetration condition,  $\mathbf{u} \cdot \mathbf{n} = 0$ .

### **WALL MOVEMENT**

This section contains controls to describe the wall movement relative to the lab (or spatial) frame.

The **Normal wall velocity** setting controls the normal wall velocity  $U_{\text{bnd}}$ . The list is per default set to **Automatic from frame**. The physics automatically detects if the spatial frame moves. This can for example happen if an ALE interface is present in the model component. If there is no movement  $U_{\text{bnd}} = 0$ . If the frame moves,  $U_{\text{bnd}}$  becomes equal to the frame movement projected on the wall normal.  $U_{\text{bnd}}$  is accounted for in the actual nonpenetration boundary condition prescribed, so that  $\mathbf{u} \cdot \mathbf{n} = U_{\text{bnd}}$ .

Select **Zero (Fixed wall)** from **Normal wall velocity** selection list to always prescribe  $U_{\text{bnd}} = 0$ .

## *Velocity*

---

The **Velocity** node sets a normal inflow velocity condition for the fluid. This prescribes a flux condition for the velocity potential,  $\nabla \phi \cdot \mathbf{n} = -U_{\text{in}}$ .

## *Open Boundary*


---

The **Open Boundary** node sets a normal flow condition for the fluid. The level of  $\phi$  is set to a constant value along the boundary, so that the tangential derivative of  $\phi$  is zero.

### **OPEN BOUNDARY**

The potential level is set by specifying the **Velocity potential**  $\phi_{\text{bnd}}$ .

## CONSTRAINT SETTINGS

This section is displayed by clicking the **Show** button (  ) and selecting **Advanced Physics Options**.

### *Symmetry*

---

The **Symmetry** node prescribes a symmetry condition for the velocity potential,  $\nabla\phi \cdot \mathbf{n} = 0$ .

### *Potential Point Constraint*


---

The **Potential Point Constraint** node can be used to specify the velocity potential level. If it is not possible to specify the velocity potential level using a boundary condition, the level must be set in some other way, for example, by specifying a fixed potential at a point.

## POTENTIAL POINT CONSTRAINT

The potential level is set by specifying the **Velocity potential**  $\phi_p$ .

## CONSTRAINT SETTINGS

This section is displayed by clicking the **Show** button (  ) and selecting **Advanced Physics Options**.

### *Periodic Flow Condition*

---

The **Periodic Flow Condition** splits its selection into a source group and a destination group (the **Destination Selection** section is optional). Fluid that leaves the domain through one of the destination boundaries enters the domain through the corresponding source boundary. This corresponds to a situation where the geometry is a periodic part of a larger geometry. If the boundaries are not parallel to each other, the velocity vector is automatically transformed.

If the boundaries are curved, the orientation of the source must be specified manually (see [Orientation of Source](#)).

## OFFSET

The **Offset** at the boundary specifies a value or expression for either the **Velocity potential difference**,  $\phi_{\text{src}} = -\phi_{\text{dst}}$ . The offset ensures a gradient of the Velocity potential between the source and destination, so that the velocity is not zero.

## BOUNDARY SELECTION

To set up a periodic boundary condition, both boundaries must be selected in the **Periodic Flow Condition** node. COMSOL Multiphysics automatically assigns one boundary as the source and the other as the destination. To manually set the destination selection, a **Destination Selection** subnode is available from the context menu (by right-clicking the parent node) or from the **Physics** toolbar, **Attributes** menu. All destination sides must be connected.




The software usually automatically identifies the boundaries as either source boundaries or destination boundaries, as indicated in the selection list. This works fine for cases like opposing parallel boundaries. In other cases, right-click **Periodic Flow Condition** and select **Manual Destination Selection** to control the destination. By default it contains the selection that COMSOL Multiphysics identifies.

## DESTINATION SELECTION

This section is available for specifying the destination boundaries, if needed, when the **Manual Destination Selection** option is selected in the context menu for the **Periodic Flow Condition** node. You can only select destination boundaries from the union of all source and destination boundaries.


## ORIENTATION OF SOURCE

To display this section, click the **Show More Options** button (  ) and select **Advanced Physics Options** in the **Show More Options** dialog box. For information about the **Orientation of Source** section, see [Orientation of Source and Destination](#) in the *COMSOL Multiphysics Reference Manual*.

## ORIENTATION OF DESTINATION

This section appears if the setting for **Transform to intermediate map** in the **Orientation of Source** section is changed from the default value, **Automatic**, and **Advanced Physics Options** is selected in the **Show More Options** dialog box. For information about the **Orientation of Destination** section, see [Orientation of Source and Destination](#) in the *COMSOL Multiphysics Reference Manual*.

## MAPPING BETWEEN SOURCE AND DESTINATION

To display this section, click the **Show More Options** button (  ) and select **Advanced Physics Options** in the **Show More Options** dialog box. For information about the



**Mapping Between Source and Destination** section, see [Mapping Between Source and Destination](#) in the *COMSOL Multiphysics Reference Manual*.

### *Interior Wall*

---

The **Interior Wall** boundary condition can only be applied on interior boundaries.

It is similar to the [Wall](#) node available on exterior boundaries except that it applies on both sides of an interior boundary. It allows discontinuities of the velocity potential across the boundary. The **Interior Wall** boundary condition can be used to avoid meshing thin structures by applying a no-penetration condition,  $\mathbf{u} \cdot \mathbf{n} = U_{\text{bnd}}$ , on interior curves and surfaces instead. Slip conditions and conditions for a moving wall can also be prescribed.

#### **WALL MOVEMENT**

This section contains controls to describe the wall movement relative to the lab (or spatial) frame.

The **Normal wall velocity** setting controls the normal wall velocity  $U_{\text{bnd}}$ . The list is per default set to **Automatic from frame**. The physics automatically detects if the spatial frame moves. This can for example happen if an ALE interface is present in the model component. If there is no movement  $U_{\text{bnd}} = 0$ . If the frame moves,  $U_{\text{bnd}}$  becomes equal to the frame movement projected on the wall normal.  $U_{\text{bnd}}$  is accounted for in the actual no-penetration boundary condition prescribed, so that  $\mathbf{u} \cdot \mathbf{n} = U_{\text{bnd}}$ .

Select **Zero (Fixed wall)** from **Normal wall velocity** selection list to always prescribe  $U_{\text{bnd}} = 0$ .

### *Flow Continuity*

---

The **Continuity** condition is suitable for pairs where the boundaries match; it prescribes that the velocity potential variable is continuous across the pair.

The parts of the boundary where a source boundary lacks a corresponding destination boundary and vice versa will get conditions from other boundary conditions under the physics interface acting as fallback boundary conditions on the nonoverlapping parts.

By default, the [Wall](#) node is used as fallback unless another feature has been selected for the boundary.



About Identity and Contact Pairs

---

#### LOCATION IN USER INTERFACE


Right-click the physics interface and select **Pairs>Continuity**.

*Ribbon*

Physics tab with the physics interface selected:

**Pairs>Continuity**

# The Rotating Machinery, Laminar Flow Interface

The **Rotating Machinery, Laminar Flow (spf)** interface, found under the **Single-Phase Flow>Rotating Machinery** branch () when adding a physics interface, is used for modeling flow where one or more of the boundaries rotate in a periodic fashion. This is used for mixers and propellers.

The physics interface supports compressible, weakly compressible and incompressible flow, and the flow of non-Newtonian fluids. In addition, the physics interface supports creeping flow, although the shallow channel approximation is redundant.

In this section:

- [Moving Mesh](#)
- [The Rotating Machinery, Laminar Flow Interface](#)
- [Domain, Boundary, Point, and Pair Nodes for the Rotating Machinery Interfaces](#)

## *Moving Mesh*

---

The Rotating Machinery, Fluid Flow interfaces rely on ALE (arbitrary Lagrangian–Eulerian) moving mesh functionality for their rotating domain and free surface implementation.

### **ROTATING DOMAINS**

When a **Rotating Machinery, Fluid Flow** interface is added using the **Model Wizard**, a **Rotating Domain** node is added automatically in the **Model Builder** under the **Definitions>Moving Mesh** node. It contains settings for specifying the rotation of one or several domains. By default, **All domains** are selected. Under the **Rotating Domain**, you can exclude any nonrotating fluid domains. For both 3D and 2D components, the **Rotation type** should be set to **Specified rotational velocity**.

Select a **Rotational velocity expression** — **General revolutions per time** (the default), **Constant revolutions per time**, **General angular velocity**, or **Constant angular velocity**.

- For **Constant revolutions per time**, or **General revolutions per time**, enter a value or expression respectively in the **Revolutions per time** input field and optionally an **Initial angle**. The angular velocity in this case is defined as the input multiplied by  $2\pi$ .
- For **Constant angular velocity**, or **General angular velocity** enter a value or an expression respectively in the **Angular velocity** input field and optionally an **Initial angle**.

### 3D Components



For 3D components, select the **Rotation axis base point**,  $\mathbf{r}_{\text{ax}}$ . The default is the origin. Select the **Rotation axis**,  $\mathbf{u}_{\text{rot}}$ , the  $z$ -axis is the default. If the  $x$ -axis is selected, it corresponds to a rotational axis (1, 0, 0) with the origin as the base point, and correspondingly for the  $y$ -axis and  $z$ -axis.

### 2D Components

For 2D components, enter coordinates for the **Rotation axis base point**,  $\mathbf{r}_{\text{ax}}$ . The default is the origin (0, 0).

## *The Rotating Machinery, Laminar Flow Interface*



---

The **Rotating Machinery, Laminar Flow (spf)** interface () , found under the **Single-Phase Flow > Rotating Machinery** branch () , is used to simulate flow at low to moderate Reynolds numbers in geometries with one or more rotating parts. The physics interface supports incompressible, weakly compressible and compressible flows at low Mach numbers (typically less than 0.3). It also supports modeling of non-Newtonian fluids. The physics interface is available for 3D and 2D components.

There are two study types available for this physics interface. Using the **Time Dependent** study type, rotation is achieved through moving mesh functionality, also known as sliding mesh. Using the **Frozen Rotor** study type, the rotating parts are kept frozen in position, and rotation is accounted for by the inclusion of centrifugal and Coriolis forces. In both types, the momentum balance is governed by the Navier–Stokes equations, and the mass conservation is governed by the continuity equation. See [Theory for the Rotating Machinery, Laminar Flow Interface](#).

When this physics interface is added, the following default physics nodes are also added in the **Model Builder** under **Laminar Flow** — **Fluid Properties**, **Wall**, and **Initial Values**. A **Moving Mesh** interface with a **Rotating Domain** node is added automatically in the **Model Builder** under the **Definitions** node. Then, from the **Physics** toolbar, add other nodes

that implement, for example, boundary conditions and volume forces. You can also right-click **Laminar Flow** to select physics features from the context menu. See [Domain, Boundary, Pair, and Point Nodes for Single-Phase Flow](#).

	Pseudo Time Stepping for Laminar Flow Models
	<i>Behavior of a Power-Law Fluid in a Mixer</i> : Application Library path <b>Polymer_Flow_Module/Verification_Examples/powerLaw_mixer</b>

**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 spf.

In addition to the settings described below, see [The Creeping Flow and Laminar Flow Interfaces](#) for all the other settings available. See [Domain, Boundary, Point, and Pair Nodes for the Rotating Machinery Interfaces](#) for links to all the physics nodes.

**ADVANCED SETTINGS**

To display this section, click the **Show More Options** button (  ) and select **Advanced Physics Options** in the **Show More Options** dialog box. Normally these settings do not need to be changed.

*Pseudo Time Stepping*

Select the **Use pseudo time stepping for stationary equation form** check box to add pseudo time derivatives to the equation when the **Frozen Rotor** equation form is used. (Frozen rotor is a pseudo stationary formulation.) When selected, also choose a **CFL number expression** — **Automatic** (the default) or **Manual**. **Automatic** sets the local CFL number (from the Courant–Friedrichs–Lewy condition) to the built-in variable CFLCMP which in turn triggers a PID regulator for the CFL number. For **Manual** enter a **Local CFL number**  $CFL_{loc}$  (dimensionless).

## *Domain, Boundary, Point, and Pair Nodes for the Rotating Machinery Interfaces*

---

All versions of the Rotating Machinery, Fluid Flow interfaces use the following domain, boundary, point, and pair physics nodes.

- [Contact Angle](#)
- [Stationary Free Surface](#)

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

The following nodes, available in the Fluid Flow interfaces, are described for the Laminar Flow interface:

- |                                    |   |
|------------------------------------|---|
| • <a href="#">Boundary Stress</a>  | • <a href="#">Outlet</a>                    |
| • <a href="#">Initial Values</a>   | • <a href="#">Periodic Flow Condition</a>   |
| • <a href="#">Interior Wall</a>    | • <a href="#">Point Mass Source</a>         |
| • <a href="#">Flow Continuity</a>  | • <a href="#">Pressure Point Constraint</a> |
| • <a href="#">Fluid Properties</a> | • <a href="#">Symmetry</a>                  |
| • <a href="#">Inlet</a>            | • <a href="#">Volume Force</a>              |
| • <a href="#">Line Mass Source</a> | • <a href="#">Wall</a>                      |
| • <a href="#">Open Boundary</a>    |   |



[Domain, Boundary, Pair, and Point Nodes for Single-Phase Flow](#)

---

## *Stationary Free Surface*

---

This postprocessing feature is used to estimate the deformation of a free surface from the pressure distribution on the boundary when using the **Frozen Rotor** or **Frozen Rotor with Initialization** study types. When computing the flow field, a slip condition together

with a constant pressure level  $p_{\text{ext}}$  is applied on the boundary. The free surface deformation  $\eta_{\text{FS}}$  is evaluated from the linearized free surface condition in a separate study step, [Stationary Free Surface](#), using the pressure from the Frozen Rotor study step:

$$p(\mathbf{x}_0) - p_{\text{ext}} + \hat{\mathbf{n}} \cdot \nabla p \Big|_{\mathbf{x} = \mathbf{x}_0} \eta_{\text{FS}} = -\sigma \nabla_S^2 \eta_{\text{FS}} \quad (3-3)$$

Here,  $\sigma$  is the surface tension coefficient,  $\mathbf{x} = \mathbf{x}_0$  represents the position of the undisturbed surface (the feature selection), and  $\hat{\mathbf{n}}$  is its unit normal. The correct study sequence can be triggered by toggling the study window once a selection for the Stationary Free Surface feature has been made.

The Stationary Free Surface feature is supported by laminar and turbulent flow, and is applicable for small surface deformations ( $\eta_{\text{FS}}$  and its gradient should be small compared to the dimensions of the computational domain). If the surface deformation is large, a time-dependent study with a **Deforming Domain** and a [Free Surface](#) feature should be used instead.

Iterative solvers are necessary to reduce the cost of models with large number of degrees of freedom. The fluid flow interfaces use a **Smoothed aggregation AMG** solver per default. When the Stationary Free Surface feature is active and has a nonempty selection in a frozen rotor or stationary study step, a **Geometric multigrid** solver is used instead to ensure robustness of the model.



[The Choice of Solver and Solver Settings](#) in the *CFD Module User's Guide* and [Studies and Solvers](#) in the *COMSOL Multiphysics Reference Manual*

## STATIONARY FREE SURFACE

Enter an **External pressure**  $p_{\text{ext}}$  corresponding to the average pressure level on the free surface.

## SURFACE TENSION

Select the **Include surface tension force in momentum equation** check box to include the surface tension force in the momentum equation.

The **Surface tension coefficient**  $\sigma$  (SI unit: N/m) can be specified from predefined libraries, by selecting **Library coefficient, liquid/gas interface** (the default) or **Library coefficient, liquid/liquid interface**, or be set to **User defined**.

- For **Library coefficient, liquid/gas interface** select an option from the list — **Water/Air** (the default), **Acetone/Air**, **Acetic acid/Air**, **Ethanol/Air**, **Ethylene glycol/Ethylene glycol vapor**, **Diethyl ether/Air**, **Glycerol/Air**, **Heptane/Nitrogen**, **Mercury/Mercury vapor**, or **Toluene/Air**.
- For **Library coefficient, liquid/liquid interface** select an option from the list — **Benzene/Water, 20°C**, **Corn oil/Water, 20°C**, **Ether/Water, 20°C**, **Hexane/Water, 20°C**, **Mercury/Water, 20°C**, or **Olive oil/Water, 20°C**.

Only one Stationary Free Surface feature can be applied in a Rotating Machinery, Fluid Flow interface, and it requires the study to be stationary (Frozen Rotor).


### *Contact Angle*

---

This feature specifies the contact angle between the free surface and a solid wall. The Contact Angle feature is a subfeature to, and added under a [Stationary Free Surface](#) feature. See the [Contact Angle](#) feature in the Multiphase Flow Interfaces chapter.



# The Viscoelastic Flow Interface

The **Viscoelastic Flow (vef)** interface () is used for simulating single-phase flows of viscoelastic fluids. The physics interface is only suitable for incompressible flows.

The equations solved by the Viscoelastic Flow interface are the continuity equation for conservation of mass, the Navier–Stokes equations, augmented by an elastic stress tensor term, and a constitutive equation that defines the extra elastic stress contribution.

The Viscoelastic Flow interface is available for stationary and time-dependent analyses. The main feature is Fluid Properties, which adds the Navier–Stokes equations, continuity equations, and the constitutive relations between the extra elastic stress term and the velocity gradient. The Fluid Properties feature provides an interface for defining the fluid material and its properties. Viscoelastic, Newtonian, and Inelastic non-Newtonian constitutive relations are available. The boundary conditions are essentially the same as for the Laminar Flow interface. When this physics interface is added, the following default nodes are also added in the **Model Builder**:

- **Fluid Properties**
- **Initial Values**
- **Wall**

Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions and volume forces. You can also right-click **Viscoelastic Flow** 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 `vef`.

## PHYSICAL MODEL

This node specifies the properties of the Viscoelastic Flow interface, which describe the overall type of fluid flow model.

### *Include Gravity*

When the **Include gravity** check box is selected, a global **Gravity** feature is shown in the interface model tree.

Also, when the **Include gravity** check box is selected, the **Use reduced pressure** option changes the pressure formulation from using the total pressure (default) to using the reduced pressure. For more information, see [Gravity](#).

## VISCOELASTIC CONSTITUTIVE EQUATION FORMULATION

This node specifies rheological constitutive equation formulation. Stress (the default), conformation formulations are available.



## DEPENDENT VARIABLES

The following dependent variables (fields) are defined for this physics interface:

- **Velocity field  $\mathbf{u}$**  and its components
- **Pressure  $p$**

If the **Viscoelastic** constitutive relation is selected, the **Fluid Properties** node also adds as variables the components of the elastic stress tensor.

For all other settings, see [The Laminar Flow Interface](#).

	<ul style="list-style-type: none"><li>• <a href="#">Domain, Boundary, Point, and Pair Nodes for the Viscoelastic Interface</a></li><li>• <a href="#">Theory for the Viscoelastic Flow Interface</a></li></ul>
	<i>Flow of Viscoelastic Fluid Past a Cylinder:</i> Application Library path <b>Polymer_Flow_Module/Verification_Examples/cylinder_flow_viscoelastic</b>

### *Domain, Boundary, Point, and Pair Nodes for the Viscoelastic Interface*

For the Viscoelastic Flow interface the [Fluid Properties](#) node is described in this section.

The following nodes (listed in alphabetical order) are described in [Domain](#), [Boundary](#), [Pair](#), and [Point Nodes for Single-Phase Flow](#):

- [Boundary Stress](#)
- [Flow Continuity](#)
- [Free Surface](#)
- [Fluid-Fluid Interface](#)
- [Gravity](#)
- [Initial Values](#)
- [Inlet](#)
- [Interior Wall](#)
- [Open Boundary](#)
- [Outlet](#)
- [Periodic Flow Condition](#)
- [Point Mass Source](#)
- [Pressure Point Constraint](#)
- [Symmetry](#)
- [Volume Force](#)
- [Wall](#)

## *Fluid Properties*

---

The **Fluid Properties** node adds the momentum and continuity equations solved by the physics interface. For the viscoelastic fluids, the constitutive relations for the extra elastic stress tensor are also added. The node also provides an interface for defining the material properties of the fluid.

Except where included below, see [Fluid Properties](#) for all the other settings.

### *Constitutive Relation*

**Viscoelastic** (default), **Specify dynamic viscosity** and **Inelastic non-Newtonian** options are available. If **Viscoelastic** constitutive relation is selected, the **Fluid Properties** node also adds the viscoelastic variables and equations.

For **Viscoelastic** constitutive relation, specify **Solvent viscosity**  $\mu_s$  and select a **Material model** — **Oldroyd-B**, **Giesekus**, **LPTT**, **EPTT**, **Rolie-Poly**, **FENE-CR**, or **FENE-P**. Viscoelastic material can be described as consisting of one or more branches. For each viscoelastic branch, enter parameters in the table. See the setting for each viscoelastic model that follows.

### *Oldroyd-B*

For **Oldroyd-B** enter the model parameters in the table. In each **Branch** row enter the polymer viscosity  $\mu_{em}$  in the **Viscosity** column, and the relaxation time  $\lambda_{em}$  in the **Relaxation time** column.

### *Giesekus*

For **Giesekus**, in each **Branch** row enter the polymer viscosity  $\mu_{em}$  in the **Viscosity** column, the relaxation time  $\lambda_{em}$  in the **Relaxation time** column, and the mobility factor  $\alpha_{em}$  in the **Mobility** column.

### *FENE-P and FENE-CR*

For **FENE-P**, in each **Branch** row enter the polymer viscosity  $\mu_{em}$  in the **Viscosity** column, the relaxation time  $\lambda_{em}$  in the **Relaxation time** column, and the extensibility parameter  $L_{em}$  in the **Extensibility** column.

### *LPTT*





For **LPTT**, in each **Branch** row enter the polymer viscosity  $\mu_{em}$  in the **Viscosity** column, the relaxation time  $\lambda_{em}$  in the **Relaxation time** column, and the extensibility  $\varepsilon_{em}$  in the **Extensibility** column.

### *EPTT*


For **LPTT**, in each **Branch** row enter the polymer viscosity  $\mu_{em}$  in the **Viscosity** column, the relaxation time  $\lambda_{em}$  in the **Relaxation time** column, and the extensibility  $\varepsilon_{em}$  in the **Extensibility** column.

### *Rolie-Poly*

For **LPTT**, in each **Branch** row enter the polymer viscosity  $\mu_{em}$  in the **Viscosity** column, the relaxation time  $\lambda_{em}$  in the **Relaxation time** column, Rouse relaxation time  $\lambda_{Rm}$  in the **Rouse relaxation time** column, convective constraint release coefficient  $\beta_m$  and exponent  $\delta_m$  in the **CCR coefficient** and **CCR exponent** columns, respectively.

- Use the **Add** button (  ) to add a row to the table and the **Delete** button (  ) to delete a row in the table.
- Use the **Load from file** button (  ) and the **Save to file** button (  ) to load and store data for the branches in a text file with three space-separated columns.

## **DISCRETIZATION**

To display this section, click the **Show More Options** button (  ) and select **Advanced Physics Options** in the **Show More Options** dialog box. Select shape function type for the components of the auxiliary viscoelastic tensor. The default setting is **Linear**.

## **THERMAL EFFECTS**


Viscoelastic properties have a strong dependence on the temperature. To model the temperature dependence, the solvent and polymer viscosities and relaxation time are modified to  $\alpha_T \mu_s$ ,  $\alpha_T \mu_{em}$ , and  $\alpha_T \lambda_{em}$ .

Select a **Thermal function** — **None**, **Arrhenius**, **Williams–Landel–Ferry (WLF)**, **Exponential**, or **User defined**. When the default, **None**, is kept, the thermal function  $\alpha_T(T)$  is set to unity and the viscosities and relaxation time are not modified.



- [Thermal Effects](#)
- 

### INITIAL VALUES

To display this section, click the **Show More Options** button (  ) and select **Advanced Physics Options** in the **Show More Options** dialog box. Enter initial values or expressions for the components of the auxiliary viscoelastic tensor. The initial values can serve as an initial condition for a transient simulation or as an initial guess for a nonlinear solver. Note, that if several branches are specified, the initial values entered above are applied to all branches.

---

**Note:** The **Interior Walls** boundary condition and Pair conditions are not applicable on the boundaries that are adjacent to the Fluid properties nodes with different number of branches.

---

# Theory for the Single-Phase Flow Interfaces

The theory for the Single-Phase Flow, Laminar Flow interface is described in this section:

- [General Single-Phase Flow Theory](#)
- [Compressible Flow](#)
- [Weakly Compressible Flow](#)
- [The Mach Number Limit](#)
- [Incompressible Flow](#)
- [The Reynolds Number](#)
- [Non-Newtonian Flow](#)
- [Gravity](#)
- [The Boussinesq Approximation](#)
- [Theory for the Wall Boundary Condition](#)
- [Prescribing Inlet and Outlet Conditions](#)
- [Mass Flow](#)
- [Fully Developed Flow \(Inlet\)](#)
- [Fully Developed Flow \(Outlet\)](#)
- [Normal Stress Boundary Condition](#)
- [Porous Interface](#)
- [Mass Sources for Fluid Flow](#)
- [Numerical Stability — Stabilization Techniques for Fluid Flow](#)
- [Solvers for Laminar Flow](#)
- [Pseudo Time Stepping for Laminar Flow Models](#)
- [Nitsche Constraints](#)

- [Particle Tracing in Fluid Flow](#)
- [References for the Single-Phase Flow, Laminar Flow Interfaces](#)



The theory about most boundary conditions is found in [Ref. 2](#).

### *General Single-Phase Flow Theory*

The Single-Phase Fluid Flow interfaces are based on the Navier–Stokes equations, which in their most general form read

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (3-4)$$

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho (\mathbf{u} \cdot \nabla) \mathbf{u} = \nabla \cdot [-p \mathbf{I} + \mathbf{K}] + \mathbf{F} \quad (3-5)$$

$$\rho C_p \left( \frac{\partial T}{\partial t} + (\mathbf{u} \cdot \nabla) T \right) = -(\nabla \cdot \mathbf{q}) + \mathbf{K} : \mathbf{S} - \left. \frac{T}{\rho} \frac{\partial \rho}{\partial T} \right|_p \left( \frac{\partial p}{\partial t} + (\mathbf{u} \cdot \nabla) p \right) + Q \quad (3-6)$$

where

- $\rho$  is the density (SI unit: kg/m<sup>3</sup>)
- $\mathbf{u}$  is the velocity vector (SI unit: m/s)
- $p$  is pressure (SI unit: Pa)
- $\mathbf{I}$  is the identity matrix (unitless)
- $\mathbf{K}$  is the viscous stress tensor (SI unit: Pa)
- $\mathbf{F}$  is the volume force vector (SI unit: N/m<sup>3</sup>)
- $C_p$  is the specific heat capacity at constant pressure (SI unit: J/(kg·K))
- $T$  is the absolute temperature (SI unit: K)
- $\mathbf{q}$  is the heat flux vector (SI unit: W/m<sup>2</sup>)
- $Q$  contains the heat sources (SI unit: W/m<sup>3</sup>)
- $\mathbf{S}$  is the strain-rate tensor:

$$\mathbf{S} = \frac{1}{2}(\nabla \mathbf{u} + (\nabla \mathbf{u})^T)$$

The operation “:” denotes a contraction between tensors defined by

$$\mathbf{a}:\mathbf{b} = \sum_n \sum_m a_{nm} b_{nm} \quad (3-7)$$

This is sometimes referred to as the double dot product.

Equation 3-4 is the continuity equation and represents conservation of mass.

Equation 3-5 is a vector equation which represents conservation of momentum.

Equation 3-6 describes the conservation of energy, formulated in terms of temperature. This is an intuitive formulation that facilitates boundary condition specifications.

To close the equation system, Equation 3-4 through Equation 3-6, constitutive relations are needed.

For a Newtonian fluid, which has a linear relationship between stress and strain, Stokes (Ref. 1) deduced the following expression:

$$\mathbf{K} = 2\mu\mathbf{S} - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I} \quad (3-8)$$

The dynamic viscosity,  $\mu$  (SI unit: Pa·s), for a Newtonian fluid is allowed to depend on the thermodynamic state but not on the velocity field. All gases and many liquids can be considered Newtonian.

For an inelastic non-Newtonian fluid, the relationship between stress and strain rate is nonlinear, and an apparent viscosity is introduced instead of the dynamic viscosity. Examples of non-Newtonian fluids are honey, mud, blood, liquid metals, and most polymer solutions.

With the Polymer Flow Module, you can model flows of non-Newtonian fluids using the predefined constitutive models. The following models which describe the stress-strain relationship for non-Newtonian fluids are available: Power law, Carreau, Carreau-Yasuda, Cross, Cross-Williamson, Bingham-Papanastasiou, Herschel-Bulkley-Papanastasiou, Casson-Papanastasiou, DeKee-Turcotte-Papanastasiou, Robertson-Stiff-Papanastasiou, Ellis, and Houska thixotropy.



In theory, the same equations describe both laminar and turbulent flows. In practice, however, the mesh resolution required to simulate turbulence with the Laminar Flow interface makes such an approach impractical.



There are several books where derivations of the Navier–Stokes equations and detailed explanations of concepts such as Newtonian fluids can be found. See, for example, the classical text by Batchelor (Ref. 3) and the more recent work by Panton (Ref. 4).

Many applications describe isothermal flows for which Equation 3-6 is decoupled from Equation 3-4 and Equation 3-5.

## 2D AXISYMMETRIC FORMULATIONS

A 2D axisymmetric formulation of Equation 3-4 and Equation 3-5 requires  $\partial/\partial\phi$  to be zero. That is, there must be no gradients in the azimuthal direction. A common additional assumption is, however, that  $u_\phi = 0$ . In such cases, the  $\phi$ -equation can be removed from Equation 3-5. The resulting system of equations is both easier to converge and computationally less expensive compared to retaining the  $\phi$ -equation. The default 2D axisymmetric formulation of Equation 3-4 and Equation 3-5 therefore assumes that

$$\begin{aligned}\partial/\partial\phi &= 0 \\ u_\phi &= 0\end{aligned}$$

## Compressible Flow

The equations of motion for a single-phase fluid are the continuity equation:

$$\frac{\partial\rho}{\partial t} + \nabla \cdot (\rho\mathbf{u}) = 0 \quad (3-9)$$

and the momentum equation:

$$\rho \frac{\partial\mathbf{u}}{\partial t} + \rho\mathbf{u} \cdot \nabla\mathbf{u} = -\nabla p + \nabla \cdot \left( \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I} \right) + \mathbf{F} \quad (3-10)$$

These equations are applicable for incompressible as well as for compressible flow with density and viscosity variations.

### *Weakly Compressible Flow*

---

The same equations as for [Compressible Flow](#) are applied for weakly compressible flow. The only difference is that the density is evaluated at the reference pressure. The density may be a function of other quantities, in particular it may be temperature dependent.

The weakly compressible flow equations are valid for incompressible as well as compressible flow with density variations independent of the pressure.

Provided that the densities dependency pressure is specified through model inputs, the density is automatically evaluated at the reference pressure level.

### *The Mach Number Limit*

---

An important dimensionless number in fluid dynamics is the Mach number,  $\text{Ma}$ , defined by

$$\text{Ma} = \frac{|\mathbf{u}|}{a}$$

where  $a$  is the speed of sound. A flow is formally incompressible when  $\text{Ma} = 0$ . This is theoretically achieved by letting the speed of sound tend to infinity. The Navier–Stokes equations then have the mathematical property that pressure disturbances are instantaneously propagated throughout the entire domain. This results in a parabolic equation system.

The momentum equation, [Equation 3-10](#), is parabolic for unsteady flow and elliptic for steady flow, whereas the continuity equation, [Equation 3-9](#), is hyperbolic for both steady and unsteady flow. The combined system of equations is thus hybrid parabolic-hyperbolic for unsteady flow and hybrid elliptic-hyperbolic for steady flow. An exception occurs when the viscous term in [Equation 3-10](#) becomes vanishingly small, such as at an outflow boundary, in which case the momentum equation becomes locally hyperbolic. The number of boundary conditions to apply on the boundary then depends on the number of characteristics propagating into the computational domain. For the purely hyperbolic system, the number of characteristics propagating from the boundary into the domain changes as the Mach number passes through unity. Hence, the number of boundary conditions required to obtain a numerically well-posed system must also change. The compressible formulation of the laminar and turbulent interfaces uses the same boundary conditions as the incompressible formulation, which implies that the compressible interfaces are not suitable for flows with a Mach number

larger than or equal to one. Yet, the practical Mach number limit is lower than one. The main reason is that the numerical scheme (stabilization and boundary conditions) of the Laminar Flow interface does not recognize the direction and speed of pressure waves. The fully compressible Navier–Stokes equations do, for example, start to display very sharp gradients already at moderate Mach numbers. But the stabilization for the single-phase flow interface does not necessarily capture these gradients. It is impossible to give an exact limit where the low Mach number regime ends and the moderate Mach number regime begins, but a rule of thumb is that the Mach number effects start to appear at  $Ma = 0.3$ . For this reason, the compressible formulation is referred to as *Compressible flow* ( $Ma < 0.3$ ) in COMSOL Multiphysics.

### *Incompressible Flow*

---

When the temperature variations in the flow are small, a single-phase fluid can often be assumed incompressible; that is,  $\rho$  is constant or nearly constant. This is the case for all liquids under normal conditions and also for gases at low velocities. For constant  $\rho$ , Equation 3-9 reduces to

$$\rho \nabla \cdot \mathbf{u} = 0 \quad (3-11)$$

and Equation 3-10 becomes

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho (\mathbf{u} \cdot \nabla) \mathbf{u} = \nabla \cdot [-p \mathbf{I} + \mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)] + \mathbf{F} \quad (3-12)$$

Provided that the densities dependency on temperature and pressure is specified through model inputs, the density is evaluated at the reference pressure level and at the reference temperature. However, if the density is a function of other quantities such as a concentration field, or if the density is specified by a user defined expression, the user has to make sure that the density is defined as constant when the incompressible flow formulation is used.

### *The Reynolds Number*

---

A fundamental characteristic in analyses of fluid flow is the Reynolds number:

$$Re = \frac{\rho U L}{\mu}$$

where  $U$  denotes a velocity scale, and  $L$  denotes a representative length. The Reynolds number represents the ratio between inertial and viscous forces. At low Reynolds

numbers, viscous forces dominate and tend to damp out all disturbances, which leads to laminar flow. At high Reynolds numbers, the damping in the system is very low, giving small disturbances the possibility to grow by nonlinear interactions. If the Reynolds number is high enough, the flow field eventually ends up in a chaotic state called turbulence.

Note that the Reynolds number can have different meanings depending on the length scale and velocity scale. To be able to compare two Reynolds numbers, they must be based on equivalent length and velocity scales.

The Fluid Flow interfaces automatically calculate the local cell Reynolds number  $\text{Re}^c = \rho |\mathbf{u}| h / (2\mu)$  using the element length  $h$  for  $L$  and the magnitude of the velocity vector  $\mathbf{u}$  for the velocity scale  $U$ . This Reynolds number is not related to the character of the flow field, but to the stability of the numerical discretization. The risk for numerical oscillations in the solution increases as  $\text{Re}^c$  grows. The cell Reynolds number is a predefined quantity available for visualization and evaluation (typically it is available as: `spf.cellRe`).

---

### *Non-Newtonian Flow*

For an inelastic non-Newtonian fluid, the relationship between stress and strain rate is nonlinear, and we may express the constitutive relation in terms of an apparent viscosity. For the incompressible flow, it is:

$$\mathbf{K} = 2\mu_{\text{app}}(\dot{\gamma}) \mathbf{S}$$

where  $\dot{\gamma}$  is the shear rate,

$$\dot{\gamma} = \sqrt{2\mathbf{S}:\mathbf{S}}$$

and the contraction operator “:” is defined by

$$\mathbf{a}:\mathbf{b} = \sum_n \sum_m a_{nm} b_{nm}$$

The Laminar Flow interfaces provide various predefined inelastic non-Newtonian constitutive models including Power law, Carreau, Carreau–Yasuda, Cross, Cross–Williamson, Sisko, Bingham–Papanastasiou, Herschel–Bulkley–Papanastasiou, Casson–Papanastasiou, DeKee–Turcotte–Papanastasiou, Robertson–Stiff–Papanastasiou, Ellis, and Houska thixotropy.

### POWER LAW

The Power law model is an example of a generalized Newtonian model. It prescribes

$$\mu_{\text{app}} = m \left( \frac{\dot{\gamma}}{\dot{\gamma}_{\text{ref}}} \right)^{n-1} \quad (3-13)$$

where  $m, n$  are scalars that can be set to arbitrary values and  $\dot{\gamma}_{\text{ref}}$  denotes a reference shear rate for which the default value is  $1 \text{ s}^{-1}$ . For  $n > 1$ , the power law describes a shear thickening (dilatant) fluid. For  $n < 1$ , it describes a shear thinning (pseudoplastic) fluid. A value of  $n$  equal to one gives the expression for a *Newtonian fluid*.

Equation 3-13 predicts an infinite viscosity at zero shear rate for  $n < 1$ . This is however never the case physically. Instead, most fluids have a constant viscosity for shear rates smaller than  $10^{-2} \text{ s}^{-1}$  (Ref. 17). Since infinite viscosity also makes models using Equation 3-13 difficult to solve, COMSOL Multiphysics implements the Power law model as

$$\mu_{\text{app}} = m \left( \frac{\max(\dot{\gamma}, \dot{\gamma}_{\text{min}})}{\dot{\gamma}_{\text{ref}}} \right)^{n-1} \quad (3-14)$$

where  $\dot{\gamma}_{\text{min}}$  is a lower limit for the evaluation of the shear rate magnitude. The default value for  $\dot{\gamma}_{\text{min}}$  is  $10^{-2} \text{ s}^{-1}$ , but can be given an arbitrary value or expression using the corresponding text field.

### CARREAU MODEL

The Carreau model defines the viscosity in terms of the following four-parameter expression

$$\mu_{\text{app}} = \mu_{\infty} + (\mu_0 - \mu_{\infty}) [1 + (\lambda \dot{\gamma})^2]^{\frac{n-1}{2}} \quad (3-15)$$

where  $\lambda$  is a parameter with the unit of time,  $\mu_0$  is the zero shear rate viscosity,  $\mu_{\infty}$  is the infinite shear-rate viscosity, and  $n$  is a dimensionless parameter. This expression is able to describe the viscosity for most stationary polymer flows.

### CARREAU-YASUDA

The Carreau-Yasuda model is a generalized version of the Carreau model with the transition parameter  $a$  which allows for modifying the stiffness of the transition from constant  $\mu_0$  to thinning. So, it has five parameters in the expression:

$$\mu_{\text{app}} = \mu_{\infty} + (\mu_0 - \mu_{\infty})[1 + (\lambda\dot{\gamma})^a]^{\frac{n-1}{a}} \quad (3-16)$$

### CROSS

The Cross model is a special case of the Carreau–Yasuda model with  $a = 1 - n$  and  $\lambda = \mu_0/\tau_{\text{tr}}$  where  $\tau_{\text{tr}}$  is the critical stress. So, it has the following form:

$$\mu_{\text{app}} = \mu_{\infty} + (\mu_0 - \mu_{\infty}) \left[ 1 + \left( \frac{\mu_0 \dot{\gamma}}{\tau_{\text{tr}}} \right)^{1-n} \right]^{-1} \quad (3-17)$$

### CROSS–WILLIAMSON

The Cross–Williamson model is a special case of the Carreau–Yasuda model with  $\mu_{\infty} = 0$  and  $a = 1$ . So, its equation has three parameters:

$$\mu_{\text{app}} = \mu_0 [1 + \lambda \dot{\gamma}]^{n-1} \quad (3-18)$$

### SISKO

The Sisko model includes the infinite shear plateau and the power law region:

$$\mu_{\text{app}} = \mu_{\infty} + m \left( \frac{\dot{\gamma}}{\dot{\gamma}_{\text{ref}}} \right)^{n-1} \quad (3-19)$$

### BINGHAM–PAPANASTASIOU

Viscoplastic fluid behavior is characterized by existence of the yield stress  $\tau_y$  — a limit which must be exceeded before significant deformation can occur. To model the stress-deformation behavior of viscoplastic materials, different constitutive equations have been propose. The Bingham plastic model is written as

$$\mu_{\text{app}} = \mu_p + \frac{\tau_y}{\dot{\gamma}}, \quad |\mathbf{K}| > \tau_y$$

where  $\mu_p$  is the plastic viscosity.

To allow computation in both yielded and unyielded region, the Papanastasiou continuous regularization for the viscosity function is used:

$$\mu_{\text{app}} = \mu_p + \frac{\tau_y}{\dot{\gamma}} [1 - \exp(-m_p \dot{\gamma})] \quad (3-20)$$

where  $m_p$  is a scale that controls the exponential growth of stress.

### HERSCHEL–BULKLEY–PAPANASTASIOU

The Herschel–Bulkley–Papanastasiou model combines the effects of the Power law the Papanastasiou continuous regularization:

$$\mu_{\text{app}} = m \left( \frac{\dot{\gamma}}{\dot{\gamma}_{\text{ref}}} \right)^{n-1} + \frac{\tau_y}{\dot{\gamma}} [1 - \exp(-m_p \dot{\gamma})] \quad (3-21)$$

where  $m, n$  are scalars that can be set to arbitrary values, and  $\dot{\gamma}_{\text{ref}}$  is the reference shear rate. In case  $n$  equals one, the Bingham–Papanastasiou model is recovered. This equation is valid in both yielded and unyielded regions. The exponent  $m_p$  controls the smoothness of the viscosity function.

### CASSON–PAPANASTASIOU

The Casson–Papanastasiou model combines Casson equation with the Papanastasiou regularization:

$$\mu_{\text{app}} = \left( \sqrt{\mu_p} + \sqrt{\frac{\tau_y}{\dot{\gamma}}} [1 - \exp(-\sqrt{m_p \dot{\gamma}})] \right)^2 \quad (3-22)$$

The equation is valid in both yielded and unyielded regions.

### DEKEE–TURCOTTE–PAPANASTASIOU

The DeKee–Turcotte–Papanastasiou model combines the DeKee–Turcotte equation with the Papanastasiou regularization

$$\mu_{\text{app}} = \mu_{\text{DK}} \exp(-\lambda_{\text{DK}} \dot{\gamma}) + \frac{\tau_y}{\dot{\gamma}} [1 - \exp(-m_p \dot{\gamma})] \quad (3-23)$$

where  $\mu_{\text{DK}}$  and  $\lambda_{\text{DK}}$  denote the shear rate viscosity and relaxation time respectively. In case  $\lambda_{\text{DK}}$  equals zero, it recovers the Bingham–Papanastasiou model.

### ROBERTSON–STIFF–PAPANASTASIOU

The Robertson–Stiff–Papanastasiou model combines the Robertson–Stiff equation with the Papanastasiou regularization

$$\mu_{\text{app}} = \left( \left( m \left( \frac{\dot{\gamma}}{\dot{\gamma}_{\text{ref}}} \right)^{n-1} \right)^{\frac{1}{n}} + \left( \frac{\tau_y}{\dot{\gamma}} [1 - \exp(-m_p \dot{\gamma})] \right)^{\frac{1}{n}} \right)^n \quad (3-24)$$

In case  $n$  equals one, it also recovers the Bingham–Papanastasiou model.

## ELLIS

In the Ellis model,  $\mu_{\text{app}}$  is computed recursively using the following equation

$$\mu_{\text{app}} = \mu_0 \left[ 1 + \left( \frac{\dot{\gamma} \mu_{\text{app}}}{\tau_{1/2}} \right)^{a_e - 1} \right]^{-1} \quad (3-25)$$

where  $\tau_{1/2}$  is the shear stress at which  $\mu_{\text{app}} = \mu_0/2$ . The exponent  $a_e$  is the shear thinning index, which is a measure of the degree of nonlinearity.

## HOUSKA THIXOTROPY

The Houska thixotropy model takes a similar form as the Herschel–Bulkley–Papanastasiou model, but with a linear dependence of the consistency  $m$  and the yield stress  $\tau_y$  on the structure field  $\xi$ :

$$\tau_y = \tau_{y,0} + \tau_{y,t} \xi, \quad m = m_{y,0} + m_{y,t} \xi$$

where  $m_{y,0}$  and  $\tau_{y,0}$  denote the consistency and yield stress of the fully broken down material, respectively,  $m_{y,0} + m_{y,t}$  and  $\tau_{y,0} + \tau_{y,t}$  indicate the consistency and yield stress of the fully recovered material, respectively.

So, its equation takes the following form:

$$\mu_{\text{app}} = \frac{\tau_{y,0} + \tau_{y,t} \xi}{\dot{\gamma}} [1 - \exp(-m_p \dot{\gamma})] + \left( (m_{y,0} + m_{y,t} \xi) \left( \frac{\dot{\gamma}}{\dot{\gamma}_{\text{ref}}} \right) \right)^{n_f - 1} \quad (3-26)$$

Here, the structure field  $\xi$  obeys the following equation:

$$\frac{d\xi}{dt} = k_f (1 - \xi) - k_b \left( \frac{\dot{\gamma}}{\dot{\gamma}_{\text{ref}}} \right)^{n_b - 1}$$

where  $k_f$  and  $k_b$  denote the rebuild and breakdown coefficients, respectively.

### Thermal Effects

It is also possible to add the thermal effects to the non-Newtonian constitutive models. The following options: **None**, **Arrhenius**, **Williams–Landel–Ferry** (WLF), **Exponential**, and **User defined** are expressed in terms of a thermal function  $\alpha_T$ . The thermal function  $\alpha_T$  is coupled with the non-Newtonian constitutive models by multiplying by  $\mu_0$ ,  $\mu_p$ ,  $m$ ,  $\mu_{\text{DK}}$ ,  $m_{y,0}$ , and  $m_{y,t}$  if they are **User defined**.



### ARRHENIUS

$$\alpha_T = \exp\left[\frac{Q}{R}\left(\frac{1}{T} - \frac{1}{T_0}\right)\right] \quad (3-27)$$

where  $Q$  denotes the activation energy,  $R$  is the universal gas constant,  $T$  and  $T_0$  are the temperature and reference temperature, respectively.

### WILLIAMS-LANDEL-FERRY

$$\log(\alpha_T) = \frac{-C_{1\text{WLF}}(T - T_{\text{WLF}})}{C_{2\text{WLF}} + T - T_{\text{WLF}}} \quad (3-28)$$

where  $C_{1\text{WLF}}$  and  $C_{2\text{WLF}}$  are model constants, and  $T_{\text{WLF}}$  is the reference temperature.

### EXPONENTIAL

$$\alpha_T = \exp[-b(T - T_0)] \quad (3-29)$$

where  $b$  denotes a temperature sensitivity, and  $T_0$  is the reference temperature.

## *Gravity*

---

### DEFINITION

The gravity force is defined from the acceleration of gravity vector,  $\mathbf{g}$ , and the density,  $\rho$ . Under usual conditions and in Cartesian coordinates with the  $z$ -axis in the vertical direction,

$$\mathbf{g} = \begin{pmatrix} 0 \\ 0 \\ -g_{\text{const}} \end{pmatrix}.$$

When gravity is considered, a volume force equal to  $\rho\mathbf{g}$  is included in the momentum equation. For example, for laminar weakly compressible flow, it reads:

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho \mathbf{u} \cdot \nabla \mathbf{u} = \nabla \cdot \left( -p \mathbf{I} + \mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - \frac{2}{3} \mu(\nabla \cdot \mathbf{u}) \mathbf{I} \right) + \mathbf{F} + \rho \mathbf{g} \quad (3-30)$$

Introducing a constant reference density  $\rho_{\text{ref}}$ , and assuming that  $\mathbf{g}$  is homogeneous, this equation is equivalently written:

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho \mathbf{u} \cdot \nabla \mathbf{u} = \nabla \cdot \left( -p \mathbf{I} + \mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - \frac{2}{3} \mu (\nabla \cdot \mathbf{u}) \mathbf{I} + \rho_{\text{ref}} \mathbf{g} \cdot (\mathbf{r} - \mathbf{r}_{\text{ref}}) \right) + \mathbf{F} + (\rho - \rho_{\text{ref}}) \mathbf{g} \quad (3-31)$$

where  $\mathbf{r}$  is the position vector and  $\mathbf{r}_{\text{ref}}$  is an arbitrary reference position vector.

From this equation, it is convenient to define the reduced pressure which accounts for the hydrostatic pressure,  $\tilde{p} = p - \rho_{\text{ref}} \mathbf{g} \cdot (\mathbf{r} - \mathbf{r}_{\text{ref}})$

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho \mathbf{u} \cdot \nabla \mathbf{u} = \nabla \cdot \left( -\tilde{p} \mathbf{I} + \mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - \frac{2}{3} \mu (\nabla \cdot \mathbf{u}) \mathbf{I} \right) + \mathbf{F} + (\rho - \rho_{\text{ref}}) \mathbf{g} \quad (3-32)$$

In Equation 3-32, the gravity force is written  $(\rho - \rho_{\text{ref}}) \mathbf{g}$ .

### INCOMPRESSIBLE FLOW

For incompressible flow the fluid density is assumed to be constant. Hence it is natural to define the reference density,  $\rho_{\text{ref}}$ , such that  $\rho = \rho_{\text{ref}} = \rho(T_{\text{ref}}, p_{\text{ref}})$  which makes it possible to simplify Equation 3-32:

$$\rho_{\text{ref}} \frac{\partial \mathbf{u}}{\partial t} + \rho_{\text{ref}} \mathbf{u} \cdot \nabla \mathbf{u} = \nabla \cdot (-\tilde{p} \mathbf{I} + \mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)) + \mathbf{F}$$

In some cases, even when the flow is modeled as incompressible, buoyancy should be accounted for. Using the approximation of the gravity force based on the thermal expansion coefficient (which is relevant in this case: since the density changes are small, the first order approximation is reasonably accurate), it is possible to rewrite the momentum equation with constant density and a buoyancy force:

$$\rho_{\text{ref}} \frac{\partial \mathbf{u}}{\partial t} + \rho_{\text{ref}} \mathbf{u} \cdot \nabla \mathbf{u} = \nabla \cdot (-\tilde{p} \mathbf{I} + \mu_{\text{ref}} (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)) + \mathbf{F} - \rho_{\text{ref}} \alpha_p (T_{\text{ref}}) (T - T_{\text{ref}}) \mathbf{g}$$

### WEAKLY COMPRESSIBLE FLOW

For weakly compressible flow the assumption is that the density depends only on the temperature. In particular, the pressure dependency of the density is neglected, and the density is evaluated at the reference pressure:  $\rho = \rho(T, p_{\text{ref}})$ . With the relative pressure as dependent variable, Equation 3-30 is used

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho \mathbf{u} \cdot \nabla \mathbf{u} = \nabla \cdot \left( -p \mathbf{I} + \mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - \frac{2}{3} \mu (\nabla \cdot \mathbf{u}) \mathbf{I} \right) + \mathbf{F} + \rho \mathbf{g}$$

With the reduced pressure as dependent variable, [Equation 3-32](#) is used:

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho \mathbf{u} \cdot \nabla \mathbf{u} = \nabla \cdot \left( -\tilde{p} \mathbf{I} + \mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - \frac{2}{3} \mu (\nabla \cdot \mathbf{u}) \mathbf{I} \right) + \mathbf{F} + (\rho - \rho_{\text{ref}}) \mathbf{g}$$

### COMPRESSIBLE FLOW

For compressible flow the density may depend on any other variable, in particular on the temperature and the pressure. Similar equations are used as for the [Weakly Compressible Flow](#) case except that the density is not evaluated at the reference pressure.

### NONISOTHERMAL FLOW COUPLING

For consistency, when the **Nonisothermal Flow** coupling is active, the assumptions made for the single phase flow interface are also made in the heat transfer interface:

- Incompressible flow: the Boussinesq approximation implies that the thermal conductivity and heat capacity are also constant.
- Weakly Compressible flow: the density is evaluated at  $p_{\text{ref}}$  in the heat interface too.
- Compressible flow: no change.

### PRESSURE FORMULATION

When the relative pressure is used (default option) the interface dependent variable represents the relative pressure and the absolute pressure is defined as  $p_A = p_{\text{ref}} + p$ . When the pressure is used to define a boundary condition (for example, when  $p_0$  defines the pressure condition at an outlet), it represents the relative pressure. Hence defining the outlet pressure as  $p_{\text{hydro,approx}} = -\rho_{\text{ref}} \mathbf{g} \cdot (\mathbf{r} - \mathbf{r}_{\text{ref}})$  compensates for the gravity force for an ambient reference pressure of 0 Pa when the density is constant, there is no external force, and provided  $p_{\text{ref}}$ ,  $\mathbf{g}$ , and  $\mathbf{r}_0$  are defined consistently.

When the reduced pressure is used, the interface dependent variable (named  $p$  by default) represents the reduced pressure. The absolute pressure is then defined as  $p_A = p_{\text{ref}} - \rho_{\text{ref}} \mathbf{g} \cdot (\mathbf{r} - \mathbf{r}_{\text{ref}}) + p$ . In this case when the pressure is used to define a boundary condition (for example, to define a pressure condition at an outlet), its value corresponds to the reduced pressure. Hence, the prescribed pressure compensates for an approximate hydrostatic pressure,  $p_{\text{hydro,approx}} = -\rho_{\text{ref}} \mathbf{g} \cdot (\mathbf{r} - \mathbf{r}_{\text{ref}})$ , which is exact only when the density is constant and there is no external force.

## PRESSURE BOUNDARY CONDITION

For an immobile fluid the momentum equation simplifies to  $\nabla \cdot (p\mathbf{I}) = \mathbf{F} + \rho\mathbf{g}$  or

$$\nabla \cdot (-\tilde{p}\mathbf{I}) = \mathbf{F} + (\rho - \rho_{\text{ref}})\mathbf{g} \text{ depending on the pressure formulation.}$$

For incompressible flow, assuming there are no external forces, this leads respectively to  $p = -\rho_{\text{ref}}\mathbf{g} \cdot (\mathbf{r} - \mathbf{r}_{\text{ref}}) + p_0$  or  $\tilde{p} = p_0$ .

For weakly compressible flow and compressible flow, since the density varies, there is no corresponding explicit expression. We have

$$p = \int_{\mathbf{r}_{\text{ref}}}^{\mathbf{r}} \rho \mathbf{g} \cdot d\mathbf{r} + p_0 \text{ and } \tilde{p} = \int_{\mathbf{r}_{\text{ref}}}^{\mathbf{r}} (\rho - \rho_{\text{ref}}) \mathbf{g} \cdot d\mathbf{r} + p_0.$$

In practice, these integrals can be problematic to evaluate. Hence, whenever possible, it is recommended to locate the pressure boundary in a region where the approximate definition of the hydrostatic pressure is applicable, or to define a boundary that is perpendicular to the gravity vector.

If it is not possible and if the pressure conditions cannot be determined, you can use a no viscous stress condition (available in the Open Boundary feature).

### *Theory for the Wall Boundary Condition*

---

See [Wall](#) for the node settings.

## SLIP

The [Slip](#) condition assumes that there are no viscous effects at the slip wall and hence, no boundary layer develops. From a modeling point of view, this is a reasonable approximation if the important effect of the wall is to prevent fluid from leaving the domain. Mathematically, the constraint can be formulated as:

$$\mathbf{u} \cdot \mathbf{n} = 0, \quad (-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T))\mathbf{n} = \mathbf{0}$$

The no-penetration term takes precedence over the Neumann part of the condition and the above expression is therefore equivalent to

$$\mathbf{u} \cdot \mathbf{n} = 0, \quad \mathbf{K}_n - (\mathbf{K}_n \cdot \mathbf{n})\mathbf{n} = \mathbf{0}$$

$$\mathbf{K}_n = \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)\mathbf{n}$$

expressing that there is no flow across the boundary and no viscous stress in the tangential direction.

For a moving wall with translational velocity  $\mathbf{u}_{tr}$ ,  $\mathbf{u}$  in the above equations is replaced by the relative velocity  $\mathbf{u}_{rel} = \mathbf{u} - \mathbf{u}_{tr}$ .

### POROUS SLIP

The **Porous slip** option, which can be chosen in the **Porous treatment of no slip condition** list when **Enable porous media domains** is activated, results in a special treatment of **Wall** boundaries and **Interior Wall** boundaries adjacent to porous domains (**Porous Medium** feature). Namely, similar to the **Navier slip** boundary condition, no penetration and tangential stress conditions are applied at the wall:

$$\mathbf{u} \cdot \mathbf{n} = 0, \quad \mathbf{K}_n - (\mathbf{K}_n \cdot \mathbf{n})\mathbf{n} = -\frac{\mu}{\varepsilon_p} \frac{\mathbf{u}_{slip}}{y_{ps}},$$

$$\mathbf{u}_{slip} = \mathbf{u} - (\mathbf{u} \cdot \mathbf{n})\mathbf{n}, \quad \mathbf{K}_n = \frac{\mu}{\varepsilon_p} (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)\mathbf{n}$$

Here,  $\mathbf{K}_n$  is the viscous wall traction,  $\mathbf{n}$  is the wall normal,  $\mathbf{u}_{slip}$  is the tangential velocity at the wall while real no slip is assumed to be applied at a distance  $d_w$  (half-height of the first cell adjacent to the wall) outside the wall, and  $y_{ps}$  is the porous slip length. An analytical derivation of the velocity profile in the boundary layer where the pressure gradient is balanced by the sum of the Darcy term, the Forchheimer drag, and the viscous term (neglecting convective terms) leads to the following expression for  $y_{ps}$ :

$$y_{ps} = \tilde{l} \left( \frac{1 + X_p e^{-\xi_p}}{1 - X_p e^{-\xi_p}} \right) \left[ \left( \frac{1 + X_p e^{-\xi_p}}{1 + X_p} \right)^2 e^{\xi_p} - 1 \right], \quad \xi_p = \frac{d_w}{\tilde{l}}, \quad \tilde{l} = \frac{l_{pore}}{\sqrt{1 + 2c_\beta}},$$

$$X_p = \frac{\sqrt{1 + 2c_\beta} - \sqrt{1 + \frac{4}{3}c_\beta}}{\sqrt{1 + 2c_\beta} + \sqrt{1 + \frac{4}{3}c_\beta}}, \quad c_\beta = \frac{\sqrt{1 + 4c_D} - 1}{2}, \quad c_D = \frac{\beta_{ND}}{\mu} \frac{\kappa^2}{\mu} |\nabla p - \rho \mathbf{g}|$$

where  $l_{pore} = \sqrt{\kappa/\varepsilon_p}$  is the porous length scale,  $\beta_{ND} = \rho c_F/\sqrt{\kappa}$  is the non-Darcian coefficient ( $c_F$  is the Forchheimer coefficient),  $\mathbf{g}$  is the gravity vector, and  $\xi_p, \tilde{l}, X_p, c_\beta, c_D$ , are intermediate variables. This formula is used when the **Pressure-gradient formulation** is chosen and it uses the pressure gradient at the wall. By default, the **Velocity formulation** is activated and the corresponding formula is:

$$y_{ps} = \tilde{l} (e^{\xi_p} - 1), \quad \xi_p = \frac{d_w}{\tilde{l}}, \quad \tilde{l} = \frac{l_{pore}}{\sqrt{1 + 2c_\beta}},$$

$$c_\beta = \left[ \left( \frac{\beta_{ND} \kappa |\mathbf{u}_{slip}|}{\mu} \frac{l_{pore}}{d_w} \right)^2 / 2 \right]^{1/3}$$

Although this formulation is an approximation, since it uses the slip velocity at the wall to reconstruct the pressure gradient, it is rather accurate. Moreover, the influence of the convective terms is partially accounted for in this formulation.

### SLIDING WALL

The sliding wall option is appropriate if the wall behaves like a conveyor belt; that is, the surface is sliding in its tangential direction. The wall does not have to actually move in the coordinate system.

- In 2D, the tangential direction is unambiguously defined by the direction of the boundary, but the situation becomes more complicated in 3D. For this reason, this boundary condition has slightly different definitions in the different space dimensions.
- For 2D and 2D axisymmetric components, the velocity is given as a scalar  $U_w$  and the condition prescribes

$$\mathbf{u} \cdot \mathbf{n} = 0, \quad \mathbf{u} \cdot \mathbf{t} = U_w$$

where  $\mathbf{t} = (n_y, -n_x)$  for 2D and  $\mathbf{t} = (n_z, -n_r)$  for axial symmetry.

- For 3D components, the velocity is set equal to a given vector  $\mathbf{u}_w$  projected onto the boundary plane:

$$\mathbf{u} = \frac{\mathbf{u}_w - (\mathbf{n} \cdot \mathbf{u}_w)\mathbf{n}}{\|\mathbf{u}_w - (\mathbf{n} \cdot \mathbf{u}_w)\mathbf{n}\|} \|\mathbf{u}_w\|$$

The normalization makes  $\mathbf{u}$  have the same magnitude as  $\mathbf{u}_w$  even if  $\mathbf{u}_w$  is not exactly parallel to the wall.

### Navier Slip

This boundary condition enforces no-penetration at the wall,  $\mathbf{u} \cdot \mathbf{n}_{\text{wall}} = 0$ , and adds a tangential stress

$$\mathbf{K}_{\text{nt}} = -\frac{\mu}{\beta} \mathbf{u}_{\text{slip}}$$

where  $\mathbf{K}_{\text{nt}} = \mathbf{K}_n - (\mathbf{K}_n \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$ ,  $\mathbf{K}_n = \mathbf{K}\mathbf{n}_{\text{wall}}$ , and  $\mathbf{K}$  is the viscous stress tensor.  $\beta$  is a slip length, and  $\mathbf{u}_{\text{slip}} = \mathbf{u} - (\mathbf{u} \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$  is the velocity tangential to the wall. The boundary condition does not set the tangential velocity component to zero; however, the extrapolated tangential velocity component is 0 at a distance  $\beta$  outside the wall.

The **Slip Length** setting is per default set to **Factor of minimum element length**. The slip length  $\beta$  is then defined as  $\beta = f_h h_{\text{min}}$ , where  $h_{\text{min}}$  is the smallest element side

(corresponds to the element size in the wall normal direction for boundary layer elements) and  $f_h$  is a user input.

In cases where the wall movement is nonzero, **Account for the translational wall velocity in the friction force** may be selected to use  $\mathbf{u} - \mathbf{u}_{\text{bnd}} - ((\mathbf{u} - \mathbf{u}_{\text{bnd}}) \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$  instead of  $\mathbf{u}_{\text{slip}}$  in the friction force. Then, the extrapolated tangential velocity component is  $\mathbf{u}_{\text{bnd}}$  at a distance  $\beta$  outside of the wall. Note that the **Velocity of sliding wall**  $\mathbf{u}_w$  is always accounted for in the friction force.

The Navier Slip boundary condition is suitable for walls adjacent to a fluid-fluid interface or a free surface when solving for laminar flow. Applying this boundary condition, the contact line (fluid-fluid-solid interface) is free to move along the wall. Note that in problems with contact lines, the tangential velocity of the wall typically represents the movement of the contact line but the physical wall is not moving. In such cases, **Account for the translational wall velocity in the friction force** should not be checked.

#### *Nonlinear Navier Slip*

This boundary condition enforces no-penetration at the wall,  $\mathbf{u} \cdot \mathbf{n}_{\text{wall}} = 0$ , and adds a tangential stress

$$\mathbf{K}_{\text{nt}} = -\left(\frac{\tau_{\text{ref}}}{\|\mathbf{K}_{\text{nt}}\|}\right)^{m-1} \frac{\mu}{\beta} \mathbf{u}_{\text{slip}}$$

where  $\mathbf{K}_{\text{nt}} = \mathbf{K}_n - (\mathbf{K}_n \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$ ,  $\mathbf{K}_n = \mathbf{K}\mathbf{n}_{\text{wall}}$ , and  $\mathbf{K}$  is the viscous stress tensor.  $\tau_{\text{ref}}$  is a reference value used for scaling and dimensionalization and  $\mathbf{u}_{\text{slip}} = \mathbf{u} - (\mathbf{u} \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$  is the velocity tangential to the wall. The Nonlinear Navier Slip boundary condition is suitable for walls adjacent to a fluid-fluid interface or a free surface when solving for laminar flow. For further information, see the [Navier Slip](#) condition.

#### *Hatzikiriakos Slip*

When Hatzikiriakos slip is selected, a slip velocity is applied when the tangential stress is larger than the specified yield stress,  $\tau_y$ ,

$$\mathbf{u}_{\text{slip}} = \begin{cases} k_1 \sinh([k_2(\|\mathbf{K}_{\text{nt}}\| - \tau_y)]) \frac{\mathbf{K}_{\text{nt}}}{\|\mathbf{K}_{\text{nt}}\|}, & \text{if } \|\mathbf{K}_{\text{nt}}\| > \tau_y \\ 0, & \text{if } \|\mathbf{K}_{\text{nt}}\| \leq \tau_y \end{cases}$$

where  $\mathbf{K}_{\text{nt}} = \mathbf{K}_n - (\mathbf{K}_n \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$ ,  $\mathbf{K}_n = \mathbf{K}\mathbf{n}_{\text{wall}}$ , and  $\mathbf{K}$  is the viscous stress tensor.  $k_1$  (SI unit: m/s) and  $k_2$  (SI unit: 1/Pa) are the slip-velocity and compliance coefficients. Due to the exponential behavior in the slip velocity, the parameters  $\tau_y$ ,  $k_1$ , and  $k_2$  need to be chosen carefully to avoid divergence of the numerical iterations. It is often advisable to start with the linear Navier slip model to get an estimate of the magnitude of the tangential stress.

In cases where the wall movement is nonzero, **Account for the translational wall velocity in the friction force** may be selected to set the slip velocity relative to the boundary velocity  $\mathbf{u}_{\text{bnd}}$ . Note that the **Velocity of sliding wall**  $\mathbf{u}_w$  is always accounted for in the boundary velocity.

The Hatzikiriakos slip condition is suitable for walls adjacent to a fluid-fluid interface or a free surface when solving for laminar flow. Applying this boundary condition, the contact line (fluid-fluid-solid interface) is free to move along the wall. Note that in problems with contact lines, the tangential velocity of the wall typically represents the movement of the contact line but the physical wall is not moving. In such cases, **Account for the translational wall velocity in the friction force** should not be selected.

#### *Asymptotic Slip*

This boundary condition enforces no-penetration at the wall,  $\mathbf{u} \cdot \mathbf{n}_{\text{wall}} = 0$ , and adds a tangential stress

$$\mathbf{K}_{\text{nt}} = -\frac{1}{k_2}(e^{\|\mathbf{u}_{\text{slip}}\|/k_1} - 1)\frac{\mathbf{u}_{\text{slip}}}{\|\mathbf{u}_{\text{slip}}\|}$$

where  $\mathbf{K}_{\text{nt}} = \mathbf{K}_n - (\mathbf{K}_n \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$ ,  $\mathbf{K}_n = \mathbf{K}\mathbf{n}_{\text{wall}}$ , and  $\mathbf{K}$  is the viscous stress tensor.  $k_1$  (SI unit: m/s) and  $k_2$  (SI unit: 1/Pa) are the slip-velocity and compliance coefficients, while  $\mathbf{u}_{\text{slip}} = \mathbf{u} - (\mathbf{u} \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$  is the velocity tangential to the wall.

In cases where the wall movement is nonzero, **Account for the translational wall velocity in the friction force** may be selected to use  $\mathbf{u} - \mathbf{u}_{\text{bnd}} - ((\mathbf{u} - \mathbf{u}_{\text{bnd}}) \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$  instead of  $\mathbf{u}_{\text{slip}}$  in the friction force.

The Asymptotic Slip boundary condition is suitable for walls adjacent to a fluid-fluid interface or a free surface when solving for laminar flow. Applying this boundary condition, the contact line (fluid-fluid-solid interface) is free to move along the wall. Note that in problems with contact lines, the tangential velocity of the wall typically represents the movement of the contact line but the physical wall is not moving. In such cases, **Account for the translational wall velocity in the friction force** should not be selected.



## CONSTRAINT SETTINGS

The wall feature uses three different techniques to constraint the velocity field:

- **Pointwise constraints** is the standard technique to enforce strong constraints in the finite element method. The desired value of the velocity is prescribed at each node point in the mesh. Since the constraint is enforced locally at each node, only local values are affected by the constraint and the constraints are independent of each other. The solvers can therefore eliminate both the constrained degrees of freedom and the constraint force terms, effectively reducing the number of degrees of freedom being solved for.

The main advantage of pointwise constraints is that they enforce the constraint exactly pointwise. This means that they do not introduce any leakage of the velocity across the wall, unless specified. The main disadvantage of pointwise constraints is that they introduce locking effects when trying to impose a no-penetration condition for the velocity,  $\mathbf{u} \cdot \mathbf{n} = 0$ , on curved walls or walls with sharp corners.

- **Weak constraints** is an alternative method to prescribe the velocity. It consists on enforcing the boundary condition for the velocity via *Lagrange multipliers*. Their main advantage is that the Lagrange multiplier can provide an accurate representation of the reaction flux at the wall. Their main disadvantage is that they introduce extra unknowns, and are usually difficult to combine with other constraint methods on adjacent boundaries. Moreover, they may require extra constraints for the Lagrange multipliers. For more information, see [Weak Constraints](#) in the *COMSOL Multiphysics Reference Manual*.
- **Nitsche constraints** use a numerical flux to prescribe the velocity at the wall. They impose the constraint in an integral sense rather than pointwise, and do not suffer from the locking effects introduced by pointwise constraints when trying to prescribe a no penetration condition for the velocity. They are also better behaved when prescribing nonlinear constraints. Their main disadvantage is that the constraint is only imposed approximately, and may produce small leaks. For more information, see [Nitsche Constraints](#).

The following combination of Constraint techniques can be selected in the **Constraint Setting** sections of Wall boundary conditions:

- **Automatic** (default) use different constraint methods depending on whether only the normal component of the velocity is prescribed, such as in the no penetration condition,  $\mathbf{u} \cdot \mathbf{n} = 0$ , imposed, for example, in Slip walls or No Slip walls using Wall

Functions or Automatic Wall Treatment, or both tangential and normal components are prescribed, as is the case of No Slip walls in laminar flow.

Nitsche constraints are used to impose the no penetration condition for Slip walls. When a No Slip condition is prescribed, pointwise constraints are used except for moving walls where Nitsche constraints are used.

- **Pointwise constraints.**
- **Nitsche constraints.**
- **Weak constraints.** They are not available on Interior Walls.
- **Mixed constraints.** This option is only available when both the tangential and normal components of the velocity need to be prescribed. The velocity on the wall normal direction is imposed via pointwise constraints. The constraint for the tangential directions is relaxed, and Nitsche constraints are used instead. This provides improved accuracy and performance when working with coarse boundary layer meshes. For more information, see [Ref. 18](#).

### *Prescribing Inlet and Outlet Conditions*

---

The Navier–Stokes equations can show large variations in mathematical behavior, ranging from almost completely elliptic to almost completely hyperbolic. This has implications when it comes to prescribing admissible boundary conditions. There is also a discrepancy between mathematically valid boundary conditions and practically useful boundary conditions. See [Inlet](#) and [Outlet](#) for the node settings.

#### **INLET CONDITIONS**

An inlet requires specification of the velocity components. The most robust way to do this is to prescribe a velocity field using a Velocity condition.

A common alternative to prescribing the complete velocity field is to prescribe a pressure and all but one velocity component. The pressure cannot be specified pointwise because this is mathematically over-constraining. Instead the pressure can be specified via a stress condition:

$$-p + 2\mu \frac{\partial u_n}{\partial n} = F_n \quad (3-33)$$

where  $\partial u_n / \partial n$  is the normal derivative of the normal velocity component.

[Equation 3-33](#) is prescribed by the Pressure condition in the Inlet and Outlet features and the Normal stress condition in the Open Boundary and Boundary Stress features.

[Equation 3-33](#) is mathematically more stringent compared to specifying the pressure

pointwise and at the same time cannot guarantee that  $p$  obtains the desired value. In practice,  $p$  is close to  $F_n$ , except for low Reynolds number flows where viscous effects are the only effects that balance the pressure. In addition to Equation 3-33, all but one velocity component must be specified. For low Reynolds numbers, this can be specified by a vanishing tangential stress condition:

$$\mu \frac{\partial u_t}{\partial n} = 0$$

which is what the Normal stress condition does. Vanishing tangential stress becomes a less well-posed inlet condition as the Reynolds number increases. The Pressure condition in the Inlet feature therefore requires a flow direction to be prescribed, which provides a well-posed condition independent of Reynolds number.

### OUTLET CONDITIONS

The most common approach is to prescribe a pressure via a normal stress condition on the outlet. This is often accompanied by a vanishing tangential stress condition:

$$\mu \frac{\partial u_t}{\partial n} = 0$$

where  $\partial u_t / \partial n$  is the normal derivative of the tangential velocity field. It is also possible to prescribe  $u_t$  to be zero. The latter option should be used with care since it can have a significant effect on the upstream solution.

The elliptic character of the Navier–Stokes equations mathematically permits specifying a complete velocity field at an outlet. This can, however, be difficult to apply in practice. The reason being that it is hard to prescribe the outlet velocity so that it is consistent with the interior solution at each point. The adjustment to the specified velocity then occurs across an outlet boundary layer. The thickness of this boundary layer depends on the Reynolds number; the higher the Reynolds number, the thinner the boundary layer.

### *Mass Flow*

---

The **Mass flow** boundary condition constrains the mass flowing into the domain across an inlet boundary. The mass flow can be specified in a number of ways.

### POINTWISE MASS FLUX

The pointwise mass flux sets the velocity at the boundary to:

$$\mathbf{u} = -\frac{m_f}{\rho} \mathbf{n}$$

where  $m_f$  is the normal mass flux and  $\rho$  is the density.

### MASS FLOW RATE

The mass flow rate boundary condition sets the total mass flow through the boundary according to:

$$-\int_{\partial\Omega} d_{bc} \rho (\mathbf{u} \cdot \mathbf{n}) dS = m$$

where  $d_{bc}$  (only present in the 2D Cartesian axis system) is the boundary thickness normal to the fluid-flow domain and  $m$  is the total mass flow rate.

In addition to the constraint on the total flow across the boundary, the tangential velocity components are set to zero on the boundary

$$\mathbf{u} \times \mathbf{n} = \mathbf{0} \quad (3-34)$$

### STANDARD FLOW RATE

The standard flow rate boundary condition specifies the mass flow as a standard volumetric flow rate. The mass flow through the boundary is set by the equation:

$$-\int_{\partial\Omega} d_{bc} \frac{\rho}{\rho_{st}} (\mathbf{u} \cdot \mathbf{n}) dS = Q_{sv}$$

where  $d_{bc}$  (only present in the 2D component Cartesian axis system) is the boundary thickness normal to the fluid-flow domain,  $\rho_{st}$  is the standard density, and  $Q_{sv}$  is the standard flow rate. The standard density is defined by one of the following equations:

$$\rho_{st} = \frac{M_n}{V_n}$$

$$\rho_{st} = \frac{p_{st} M_n}{R T_{st}}$$

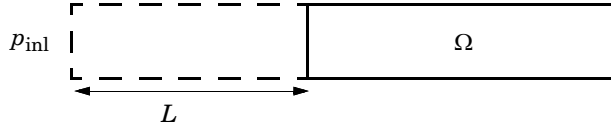
where  $M_n$  is the mean molar mass of the fluid,  $V_n$  is the standard molar volume,  $p_{st}$  is the standard pressure,  $R$  is the universal molar gas constant, and  $T_{st}$  is the standard temperature.

Equation 3-34 or Equation 3-35 is also enforced for compressible and incompressible flow, respectively, ensuring that the normal component of the viscous stress and the tangential component of the velocity are zero at the boundary.

### *Fully Developed Flow (Inlet)*

---

The **Fully developed flow** boundary condition can be understood from the following figure:



The flow to the domain  $\Omega$  is assumed to enter through a straight channel of length  $L$ . The channel is a virtual extrusion of the inlet cross section and a pressure constant pressure  $P_{\text{inl}}$  is applied on the inlet of the virtual channel.

The Fully Developed Flow boundary condition prescribes that the tangential flow component on the boundary is zero:

$$\mathbf{u} - (\mathbf{u} \cdot \mathbf{n})\mathbf{n} = \mathbf{0}$$

The momentum equation for a fully developed flow in the virtual extrusion of the inlet cross section can then be projected onto the inlet boundary with the following weak equation as the result:

$$\int_{\partial\Omega} \left( \left( L \left( -\mu (\nabla_{\mathbf{t}} \mathbf{u} + (\nabla_{\mathbf{t}} \mathbf{u})^T) + \frac{(p + P_{\text{inl}})}{2} \mathbf{I} \right) : (\nabla_{\mathbf{t}} \hat{\mathbf{u}}) \right) - P_{\text{inl}} (\mathbf{n} \cdot \hat{\mathbf{u}}) \right)$$

The exact value of  $L$  is somewhat arbitrary as long as it is not too high or too low.  $L$  is therefore set to ten times the inlet edge length in 2D and to ten times the square root of the inlet area in 2D axisymmetry and in 3D.

The fact that the velocity profile is not prescribed, but rather the solution of a projected weak contribution, means that the actual velocity profile that is obtained on the inlet can deviate from the analytical fully developed flow profile (in cases such an analytical solution exists) if required by the solution inside the computational domain,  $\Omega$ . This is most notably if the flow is strongly curved just downstream of the inlet. This can, for example, be the case if an object is positioned just downstream of the inlet or if the adjacent boundaries are not orthogonal to the inlet. The analytical solution can in

these cases be recovered by including a little bit of the inlet channel in the computational domain.

The inlet pressure,  $P_{\text{inl}}$ , must be solved for and the its equation is a discrete algebraic equation (DAE) for  $P_{\text{inl}}$ . When, for example, the average velocity is specified, the DAE reads

$$(\langle \mathbf{u} \cdot \mathbf{n} \rangle + U_{\text{av}}) \hat{P}_{\text{inl}}$$

where  $\langle \cdot \rangle$  denotes the average over the inlet. Since the equation for  $P_{\text{inl}}$  is a DAE (the equation for  $P_{\text{inl}}$  does not contain  $P_{\text{inl}}$ ), it must be solved coupled to Navier–Stokes and it must be treated by a Vanka presmoothing and postsmoothing if iterative solvers are used.

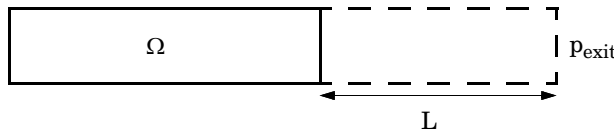
The boundary conditions for the virtual inlet channel are inherited from the boundaries adjacent to the inlet channel. Virtual boundaries adjacent to walls (except slip walls) are treated as no-slip walls. Virtual boundaries adjacent to any other type of boundary are treated as slip-walls (or equivalently as symmetry boundaries). So in the figure above, if the lower boundary of  $\Omega$  is a no-slip wall and the top boundary is a symmetry boundary, the lower boundary of the virtually extruded channel becomes a no-slip wall and the upper boundary becomes a symmetry boundary.

Algebraic turbulence models need to additional equations or constraints. For all other turbulence models, their weak equations in the virtual inlet channel are projected onto the inlet of the computational domain. The projections are regularized to prevent the production to fall to zero, so the solutions of the projects can show slight deviation from the correct solutions close to no-slip walls.

For the Viscoelastic Flow interface, the equations for the elastic stress tensor components in the virtual inlet channel are projected onto the inlet of the computational domain.

### *Fully Developed Flow (Outlet)*

The **Fully developed flow** boundary condition can be understood from the following figure:



The flow to the domain  $\Omega$  is assumed to enter through a straight channel of length  $L$ . The channel is a virtual extrusion of the inlet cross section and a pressure constant pressure  $P_{\text{exit}}$  is applied on the inlet of the virtual channel.

The Fully Developed Flow boundary condition prescribes that the tangential flow component on the boundary is zero:

$$\mathbf{u} - (\mathbf{u} \cdot \mathbf{n})\mathbf{n} = \mathbf{0}$$

The momentum equation for a fully developed flow in the virtual extrusion of the inlet cross section can then be projected onto the inlet boundary with the following weak equation as the result:

$$\int_{\partial\Omega} \left( \left( L \left( -\mu (\nabla_{\mathbf{t}} \mathbf{u} + (\nabla_{\mathbf{t}} \mathbf{u})^T) + \frac{(p + P_{\text{exit}})}{2} \mathbf{I} \right) : (\nabla_{\mathbf{t}} \hat{\mathbf{u}}) \right) - P_{\text{exit}} (\mathbf{n} \cdot \hat{\mathbf{u}}) \right)$$

The exact value of  $L$  is somewhat arbitrary as long as it is not too high or too low.  $L$  is therefore set to ten times the inlet edge length in 2D and to ten times the square root of the inlet area in 2D axisymmetry and in 3D.

The fact that the velocity profile is not prescribed, but rather the solution of a projected weak contribution, means that the actual velocity profile that is obtained on the outlet can deviate from the analytical fully developed flow profile (in cases such an analytical solution exists) if required by the solution inside the computational domain,  $\Omega$ . The inlet pressure,  $P_{\text{exit}}$ , must be solved for and the its equation is a discrete algebraic equation (DAE) for  $P_{\text{exit}}$ . When, for example, the average velocity is specified, the DAE reads

$$(\langle \mathbf{u} \cdot \mathbf{n} \rangle + U_{\text{av}}) \dot{P}_{\text{exit}}$$

where  $\langle \cdot \rangle$  denotes the average over the inlet. Since the equation for  $P_{\text{ext}}$  is a DAE, it must be solved coupled to Navier–Stokes and it must be treated by a Vanka presmooter and postsmooter if iterative solvers are used.

The boundary conditions for the virtual outlet channel are inherited from the boundaries adjacent to the outlet channel. Virtual boundaries adjacent to walls (except slip walls) are treated as no-slip walls. Virtual boundaries adjacent to any other type of boundary are treated as slip-walls (or equivalently as symmetry boundaries).

### Normal Stress Boundary Condition

---

The total stress on the boundary is set equal to a stress vector of magnitude  $f_0$ , oriented in the negative normal direction:

$$\left(-p\mathbf{I} + \left(\mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I}\right)\right)\mathbf{n} = -f_0\mathbf{n}$$

$$(-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T))\mathbf{n} = -f_0\mathbf{n}$$

using the compressible/weakly compressible and the incompressible formulation, respectively.

This implies that the total stress in the tangential direction is zero. This boundary condition implicitly sets a constraint on the pressure which for 2D flows is

$$p = 2\mu\frac{\partial u_n}{\partial n} + f_0 \quad (3-35)$$

If  $\partial u_n/\partial n$  is small, [Equation 3-35](#) states that  $p \approx f_0$ .

The Normal Stress condition is the mathematically correct version of the [Pressure Conditions](#) condition ([Ref. 4](#)), but it is numerically less stable.

### Pressure Boundary Condition

---

For single-phase flow, a mathematically correct natural boundary condition for outlets is

$$\left(-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I}\right)\mathbf{n} = -p_0\mathbf{n} \quad (3-36)$$

$$(-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T))\mathbf{n} = -p_0\mathbf{n} \quad (3-37)$$

using the compressible/weakly compressible and the incompressible formulation, respectively.

This is a normal stress condition together with a no-tangential-stress condition. When  $\mu > 0$ , [Equation 3-36](#) or [Equation 3-37](#) can be supplemented with a tangential velocity condition

$$\mathbf{u} \cdot \mathbf{t} = 0 \quad (3-38)$$



If so, the no-tangential-stress condition is overridden. An issue with Equation 3-36 or Equation 3-37 is that it does not strongly enforce unidirectional flow on the boundary. If the prescribed pressure on an outlet is too high, parts of the outlet can actually have inflow. This is not as much of an issue for the Navier–Stokes equations as it is an issue for scalar transport equations solved along with the Navier–Stokes equations. Hence, when applying the **Pressure** boundary condition at an outlet or inlet you can further constrain the flow. With the **Suppress backflow** option

$$\begin{aligned} \left( -p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla\cdot\mathbf{u})\mathbf{I} \right) \mathbf{n} &= -\hat{p}_0\mathbf{n} \\ (-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)) \mathbf{n} &= -\hat{p}_0\mathbf{n} \\ \hat{p}_0 &\leq p_0 \end{aligned} \quad , \quad (3-39)$$

the normal stress is adjusted to keep

$$\mathbf{u} \cdot \mathbf{n} \geq 0 \quad (3-40)$$

Equation 3-39 effectively means that the prescribed pressure is  $p_0$  if  $\mathbf{u} \cdot \mathbf{n} \geq 0$ , but smaller at locations where  $\mathbf{u} \cdot \mathbf{n} < 0$ . This means that Equation 3-39 does not completely prevent backflow, but the backflow is substantially reduced. Backflow is suppressed also when external forces are acting on the fluid, provided the magnitude of these forces are of the same order as the dynamic pressure at the outlet.

A pressure condition can also be applied at an inlet. In this case, either the normal stress is prescribed

$$\begin{aligned} \mathbf{n}^T \left( -p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla\cdot\mathbf{u})\mathbf{I} \right) &= -\hat{p}_0 \\ \mathbf{n}^T (-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)) &= -\hat{p}_0 \\ \hat{p}_0 &\geq p_0 \end{aligned} \quad (3-41)$$

together with the tangential condition in Equation 3-38, or, a general flow direction is prescribed.

$$\begin{aligned}
\mathbf{r}_{\mathbf{u}}^T \left( -p \mathbf{I} + \mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - \frac{2}{3} \mu (\nabla \cdot \mathbf{u}) \mathbf{I} \right) \mathbf{n} &= -\hat{p}_0 (\mathbf{r}_{\mathbf{u}} \cdot \mathbf{n}) \\
\mathbf{r}_{\mathbf{u}}^T (-p \mathbf{I} + \mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)) \mathbf{n} &= -\hat{p}_0 (\mathbf{r}_{\mathbf{u}} \cdot \mathbf{n}) \\
\hat{p}_0 &\geq p_0 \\
\mathbf{u} - (\mathbf{u} \cdot \mathbf{r}_{\mathbf{u}}) \mathbf{r}_{\mathbf{u}} &= \mathbf{0}, \quad \mathbf{r}_{\mathbf{u}} = \frac{\mathbf{d}_{\mathbf{u}}}{\|\mathbf{d}_{\mathbf{u}}\|}
\end{aligned} \tag{3-42}$$

The “>” option is used with suppress backflow to have  $\mathbf{u} \cdot \mathbf{n} \leq 0$  or  $\mathbf{u} \cdot \mathbf{r}_{\mathbf{u}} \geq 0$ .

For incompressible single-phase flow, it is also allowed to specify the total pressure,  $p_{\text{tot}}$ , instead of the static pressure,  $p_{\text{stat}}$ , on inlet and outlet boundaries. It is more useful, for example, in pump applications. The pressure is then prescribed at the boundaries using Bernoulli’s principle,

$$p = p_{\text{stat}} = p_{\text{tot}} - \frac{1}{2} \rho |\mathbf{u}|^2 \tag{3-43}$$

The equation is imposed with two options: **Average** and **Pointwise**.

In the first option,  $p_{\text{stat}}$  is prescribed by:

$$\mathbf{n}^T (-p \mathbf{I} + \mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)) \mathbf{n} = \mathbf{n}^T (-p_{\text{stat}} \mathbf{I} + \mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)) \mathbf{n} \tag{3-44}$$

where  $p_{\text{tot}}$  and  $|\mathbf{u}|^2$  are averaged over the boundaries using the `aveop` operator:

$$p_{\text{stat}} = \text{aveop}(p_{\text{tot}}) - \frac{1}{2} \rho \cdot \text{aveop}(|\mathbf{u}|^2)$$

For the second option, [Equation 3-43](#) is prescribed pointwise.

See [Inlet](#), [Outlet](#), [Open Boundary](#), and [No Viscous Stress](#) for the individual node settings. Note that some modules have additional theory sections describing options available with that module.

### *Porous Interface*

---

This condition properly accounts for a sudden jump in porosity  $\varepsilon_p$  across boundary between different domains, giving the corresponding jump in stress across the interface. The emergence of a boundary layer on the upstream side of the interface is

also revealed, which provides modifications to the behavior of the tangential velocity near the interface. The feature helps to avoid the need to add a very thin transition region, in which  $\varepsilon_{p,\text{upstream}}$  changes to  $\varepsilon_{p,\text{downstream}}$  continuously, while solving the Brinkman equation in such situations. Below, ‘-’ and ‘+’ (used as subscripts and superscripts) refer to the upstream and downstream side of the interface, respectively.

Whatever the option for **Continuity conditions** is chosen, the velocity conditions are

$$[\rho(\mathbf{u} \cdot \mathbf{n})]_{-}^{+} = 0, \quad \mathbf{u}_t|_{-}^{+} \equiv [\mathbf{u} - (\mathbf{u} \cdot \mathbf{n})\mathbf{n}]_{-}^{+} = 0$$

That is, the mass flux and tangential velocity across the interface are constant, and  $\mathbf{n}$  is assumed to be a normal pointing from ‘-’ to ‘+’. **Velocity and stress** option just imposes usual stress continuity adding a correction for a possible jump in density:

$$\left[ p\mathbf{n} - \mathbf{K}\mathbf{n} + \rho(\mathbf{u} \cdot \mathbf{n}) \left( \left[ \frac{1}{\varepsilon_p} \right]_{\text{av}} \right)^2 \mathbf{u} \right]_{-}^{+} = 0$$

The option **Velocity and porosity-corrected stress** (default) adds a contribution due to the inertial term, which is non-trivial (gives non-zero effect) when  $\varepsilon_p$  jumps across the interface:

$$\left[ p\mathbf{n} - \mathbf{K}\mathbf{n} + \rho(\mathbf{u} \cdot \mathbf{n}) \left( \left[ \frac{1}{\varepsilon_p} \right]_{\text{av}} \right)^2 \mathbf{u} + \rho(\mathbf{u} \cdot \mathbf{n}) \left( \frac{1}{2\varepsilon_p^2} \right) \mathbf{u}_{\text{av}} \right]_{-}^{+} = 0$$

Notice that  $\left[ \frac{1}{\varepsilon_p} \right]_{\text{av}} = \left( \left[ \frac{1}{\varepsilon_p} \right]_{-} + \left[ \frac{1}{\varepsilon_p} \right]_{+} \right) / 2$ ,  $\mathbf{u}_{\text{av}} = (\mathbf{u}_{-} + \mathbf{u}_{+}) / 2$  and so on.

Then, the **Velocity and porosity-corrected stress with loss** option has a ‘pressure loss’ term:

$$\left[ p\mathbf{n} - \mathbf{K}\mathbf{n} + \rho(\mathbf{u} \cdot \mathbf{n}) \left( \left[ \frac{1}{\varepsilon_p} \right]_{\text{av}} \right)^2 \mathbf{u} + \rho(\mathbf{u} \cdot \mathbf{n}) \left( \frac{1}{2\varepsilon_p^2} \right) \mathbf{u}_{\text{av}} \right]_{-}^{+} + p^{(\text{loss})} \mathbf{n} = 0$$

This loss occurs in the downstream direction. Its mechanism is similar to the mechanism of common engineering ‘total pressure loss’. Indeed, ‘expansion’ and ‘contraction’ across the interface experienced by the flow on the microscopic level (due to increasing or decreasing porosity,  $\varepsilon_p$ , respectively), are roughly equivalent to expansion and contraction of flow in a pipe with geometrical variations. Accordingly,

it is assumed that only the normal velocity component is responsible for the loss. In the case of normal flow and negligible viscous term the pipe-analogy is quite exact.

Subsequently, **Pressure loss model** should be specified. Choosing **User-defined loss coefficients** requires values of **Expansion loss coefficient** and **Contraction loss coefficient** ( $K_{\text{exp}} = 1$  and  $K_{\text{contr}} = 0.5$  are default values). The resulting losses on the ‘expansion’ and ‘contraction’ boundaries are:

$$p^{(\text{loss,expansion})} = K_{\text{exp}} \left( \left( \frac{\rho(\mathbf{u} \cdot \mathbf{n})^2}{2\varepsilon_p^2} \right)_{-} \right),$$

$$p^{(\text{loss,contraction})} = K_{\text{contr}} \left( \frac{\rho(\mathbf{u} \cdot \mathbf{n})^2}{2\varepsilon_p^2} \right)_{+}$$

**Borda–Carnot** option results in the following losses:

$$p^{(\text{loss,expansion})} = \frac{\rho_{\text{av}}}{2} \left( \left[ \left( \frac{\mathbf{u} \cdot \mathbf{n}}{\varepsilon_p} \right)_{+} \right] - \left[ \left( \frac{\mathbf{u} \cdot \mathbf{n}}{\varepsilon_p} \right)_{-} \right] \right)^2,$$

$$p^{(\text{loss,contraction})} = \frac{1}{2} \left( \frac{\rho(\mathbf{u} \cdot \mathbf{n})^2}{\varepsilon_p^2} \right)_{+} \left( \frac{1}{c_{\text{vc}}} - 1 \right)^2$$

and requires to choose **Vena contracta model**. The **User defined** option asks for value of **Vena contracta coefficient** ( $c_{\text{vc}} = 0.5$  by default). **Weisbach** option computes the coefficient based on the ratio of porosities across the interface:

$$c_{\text{vc}} = c_{\text{vc,ll}} + (1 - c_{\text{vc,ll}}) \left( \frac{\varepsilon_{\text{p},+}}{\varepsilon_{\text{p},-}} \right)^{n_{\text{vc}}}$$

and requires inputs for **Lower limit** and **Exponent** ( $c_{\text{vc,ll}} = 0.63$  and  $n_{\text{vc}} = 3$  are default values).

Although the velocity conditions formally look the same for all **Continuity conditions** options, the latter two (containing the inertial correction due to the porosity jump) lead to an extra non-trivial effect. Namely, the tangential velocity exponentially evolves over a boundary layer formed on the upstream side of the interface boundary. This boundary layer has a characteristic length equal to

$$l_{-} = \frac{\mu \varepsilon_{\text{p},-}}{\rho(\mathbf{u} \cdot \mathbf{n})}$$

The ratio of the tangential velocities right after the interface (‘downstream’) and outside the boundary layer (‘upstream infinity’) is

$$\frac{|\mathbf{u}_{t,+}|}{|\mathbf{u}_{t,\text{upstream infinity}}|} = \frac{2\varepsilon_{p,+}/\varepsilon_{p,-}}{\varepsilon_{p,+}/\varepsilon_{p,-} + \varepsilon_{p,-}/\varepsilon_{p,+}}$$

In terms of the interstitial velocity,  $\mathbf{u}^{(i)} = \mathbf{u}/\varepsilon_p$ , this results in:

$$\frac{|\mathbf{u}^{(i)}_{t,+}|}{|\mathbf{u}^{(i)}_{t,\text{upstream infinity}}|} = \frac{2}{\varepsilon_{p,+}/\varepsilon_{p,-} + \varepsilon_{p,-}/\varepsilon_{p,+}} \leq 1$$

Obviously, tangential component of the interstitial kinetic energy is lost across the interface. For low viscosity and high normal-flow component, when  $l_u$  is much lower than  $l_{geom}$  (the characteristic geometry length) there are apparent variations in  $\mathbf{u}_t$  and  $\mathbf{u}^{(i)}_t$  in a thin boundary layer on the upstream side of the interface. However, for the cases with high viscosity or nearly interface-parallel flow,  $l_u$  is much larger than  $l_{geom}$ , and any variations in  $\mathbf{u}_t$  and  $\mathbf{u}^{(i)}_t$  are virtually invisible.

Notice that the feature does not take into account possible underresolved porous scale, as is described in [Theory for the Free and Porous Media Flow, Darcy Interface](#).

### *Mass Sources for Fluid Flow*

There are two types of mass sources in a Single-Phase Flow interface: point sources and line sources.



These features require at least one of the following licenses: Battery Design Module, CFD Module, Chemical Reaction Engineering Module, Corrosion Module, Electrochemistry Module, Electrodeposition Module, Fuel Cell & Electrolyzer Module, Microfluidics Module, Pipe Flow Module, Polymer Flow Module, or Subsurface Flow Module.

#### **POINT SOURCE**

A point source is theoretically formed by taking a mass injection/ejection,  $\dot{Q}$  (SI unit:  $\text{kg}/(\text{m}^3 \cdot \text{s})$ ), in a small volume  $\delta V$  and then letting the size of the volume tend to zero while keeping the total mass flux constant. Given a point source strength,  $\dot{q}_p$  (SI unit:  $\text{kg}/\text{s}$ ), this can be expressed as

$$\lim_{\delta V \rightarrow 0} \int_{\delta V} \dot{Q} = \dot{q}_p \quad (3-45)$$

An alternative way to form a point source/sink is to assume that mass is injected/extracted through the surface of a small object. Letting the object surface area tend to zero while keeping the mass flux constant, results in the same point source. For this alternative approach, effects resulting from the physical object volume, such as drag and fluid displacement, need to be neglected.

The weak contribution

$$\dot{q}_p \text{test}(p)$$

is added to a point in the geometry. As can be seen from Equation 3-45,  $\dot{Q}$  must tend to plus or minus infinity as  $\delta V$  tends to zero. This means that in theory the pressure also tends to plus or minus infinity.

Observe that “point” refers to the physical representation of the source. A point source can therefore only be added to points in 3D components and to points on the symmetry axis in 2D axisymmetry components. Other geometrical points in 2D components represent physical lines.

The finite element representation of Equation 3-45 corresponds to a finite pressure in a point with the effect of the point source spread out over a region around the point. The size of the region depends on the mesh and on the strength of the source. A finer mesh gives a smaller affected region, but also a more extreme pressure value. It is important not to mesh too finely around a point source since the resulting pressure can result in unphysical values for the density, for example. It can also have a negative effect on the condition number for the equation system.

#### LINE SOURCE

A line source can theoretically be formed by assuming a source of strength  $\dot{Q}$  (SI unit:  $\text{kg}/(\text{m}^3 \cdot \text{s})$ ), located within a tube with cross-sectional area  $\delta S$  and then letting  $\delta S$  tend to zero, while keeping the total mass flux per unit length constant. Given a line source strength,  $\dot{q}_l$  (SI unit:  $\text{kg}/(\text{m} \cdot \text{s})$ ), this can be expressed as

$$\lim_{\delta S \rightarrow 0} \int_{\delta S} \dot{Q} = \dot{q}_l \quad (3-46)$$

As in the point source case, an alternative approach is to assume that mass is injected/extracted through the surface of a small object. This results in the same mass source,



but requires that effects on the fluid resulting from the physical object volume are neglected.

The weak contribution


$$\dot{q}_{1\text{test}}(p)$$

is added to lines in 3D or to points in 2D (which represent cut-through views of lines). Line sources can also be added to the axisymmetry line in 2D axisymmetry components. It cannot, however, be added to geometrical lines in 2D because they represent physical planes.

As with a point source, it is important not to mesh too finely around the line source.

	For feature node information, see <a href="#">Line Mass Source</a> and <a href="#">Point Mass Source</a> .
	<p>For the Reacting Flow in Porous Media, Diluted Species interface, which is available with the CFD Module, Chemical Reaction Engineering Module, or Battery Design Module, these shared physics nodes are renamed as follows:</p> <ul style="list-style-type: none"> <li>• The <b>Line Mass Source</b> node is available as two nodes, one for the fluid flow (<b>Fluid Line Source</b>) and one for the species (<b>Species Line Source</b>).</li> <li>• The <b>Point Mass Source</b> node is available as two nodes, one for the fluid flow (<b>Fluid Point Source</b>) and one for the species (<b>Species Point Source</b>).</li> </ul>

### Numerical Stability — Stabilization Techniques for Fluid Flow

The momentum equation ([Equation 3-10](#) or [Equation 3-12](#)) is a (nonlinear) convection-diffusion equation. Such equations can easily become unstable if discretized using the Galerkin finite element method. Stabilized finite element methods are usually necessary in order to obtain physical solutions. The stabilization settings are found in the main Fluid Flow interface features. To display this section, click the **Show More Options** button (  ) and select **Stabilization** in the **Show More Options** dialog box.

There are three types of stabilization methods available for Navier–Stokes — *streamline diffusion*, *crosswind diffusion*, and *isotropic diffusion*. Streamline

diffusion and crosswind diffusion are consistent stabilization methods, whereas isotropic diffusion is an inconsistent stabilization method.

For optimal functionality, the exact weak formulations of and constants in the streamline diffusion and crosswind diffusion methods depend on the order of the shape functions (basis functions) for the elements. The values of constants in the streamline diffusion and crosswind diffusion methods follow [Ref. 5](#) and [Ref. 6](#).

### **STREAMLINE DIFFUSION**

For strongly coupled systems of equations, the streamline diffusion method must be applied to the system as a whole rather than to each equation separately. These ideas were first explored by Hughes and Mallet ([Ref. 8](#)) and were later extended to Galerkin least-squares (GLS) applied to the Navier–Stokes equations ([Ref. 9](#)). This is the streamline diffusion formulation that COMSOL Multiphysics supports.

Streamline diffusion is active by default because it is necessary when convection is dominating the flow.

The governing equations for incompressible flow are subject to the Babuška–Brezzi condition, which states that the shape functions (basis functions) for pressure must be of lower order than the shape functions for velocity. If the incompressible Navier–Stokes equations are stabilized by streamline diffusion, it is possible to use equal-order interpolation. Hence, streamline diffusion is necessary when using first-order elements for both velocity and pressure. This applies also if the model is solved using geometric multigrid (either as a solver or as a preconditioner) and at least one multigrid hierarchy level uses linear Lagrange elements.

### **CROSSWIND DIFFUSION**

Crosswind diffusion can also be formulated for systems of equations, and when applied to the Navier–Stokes equations it becomes a shock-capturing operator. COMSOL Multiphysics supports the formulation in [Ref. 9](#) with a shock-capturing viscosity of the Hughes–Mallet type [Ref. 8](#).

Incompressible flows do not contain shock waves, but crosswind diffusion is still useful for introducing extra diffusion in sharp boundary layers and shear layers that otherwise would require a very fine mesh to resolve.

Crosswind diffusion is active by default as it makes it easier to obtain a solution even if the problem is fully resolved by the mesh. Crosswind diffusion also enables the iterative solvers to use inexpensive presmoothers. If crosswind diffusion is deactivated, more expensive preconditioners must be used instead.



## STABILIZATION TIME SCALES

The time-scale tensor for time-dependent problems should in theory depend on a time-scale that many references just set to the time-step taken by the time-solver,  $\Delta t$  (see, for example, [Ref. 9](#) and [Ref. 10](#)). This simple approach does not, however, necessarily reflect on the actual time-scales in the physics. A typical example is reacting flows where the time step is often guided by fast reactions, while the flow develops relatively slowly. The COMSOL Multiphysics software can therefore replace  $\Delta t^2$  in the time-scale tensor with measures of type  $(1/(\Delta \tilde{t})^2)^{-1}$ , which are calculated from projections of weak expressions in a fashion similar to those in [Ref. 11](#). These measures of the time scale are used when **Use dynamic subgrid time scale** is activated.

The **Limit small time steps effect on stabilization time scale** option can be used to ramp down the time-step contribution in the stabilization when the time step becomes much smaller than the convective and diffusive time scales. This option prevents loss of pressure stabilization at such small time steps.

## ISOTROPIC DIFFUSION

Isotropic diffusion adds diffusion to the Navier–Stokes equations. Isotropic diffusion significantly reduces the accuracy of the solution but does a very good job at reducing oscillations. The stability of the continuity equation is not improved.



### Numerical Stabilization and Iterative

---

## *Solvers for Laminar Flow*

---

The Navier–Stokes equations constitute a nonlinear equation system. A nonlinear solver must hence be applied to solve the problem. The nonlinear solver iterates to reach the final solution. In each iteration, a linearized version of the nonlinear system is solved using a linear solver. In the time-dependent case, a time marching method must also be applied. The default suggestions for each of these solver elements are discussed below.

## NONLINEAR SOLVER

The nonlinear solver method depends on if the model solves a stationary or a time-dependent problem.

### *Stationary Solver*

In the stationary case, a fully coupled, damped Newton method is applied. The initial damping factor is low since a full Newton step can be harmful unless the initial values are close to the final solution. The nonlinear solver algorithm automatically regulates the damping factor in order to reach a converged solution.

For advanced models, the automatically damped Newton method might not be robust enough. A pseudo time-stepping algorithm can then be invoked. See [Pseudo Time Stepping for Laminar Flow Models](#).

### *Time-Dependent Solver*

In the time-dependent case, the initial guess for each time step is (loosely speaking) the previous time step, which is a very good initial value for the nonlinear solver. The automatic damping algorithm is then not necessary. The damping factor in the Newton method is instead set to a constant value slightly smaller than one. Also, for the same reason, it suffices to update the Jacobian once per time step.

It is seldom worth the extra computational cost to update the Jacobian more than once per time step. For most models it is more efficient to restrict the maximum time step or possibly lower the damping factor in the Newton method.

## **LINEAR SOLVER**

The linearized Navier–Stokes equation system has saddle point character, unless the density depends on the pressure. This means that the Jacobian matrix has zeros on the diagonal. Even when the density depends on the pressure, the equation system effectively shares many numerical properties with a saddle point system.

For small 2D and 3D models, the default solver suggestion is a direct solver. Direct solvers can handle most nonsingular systems and are very robust and also very fast for small models. Unfortunately, they become slow for large models and their memory requirement scales as somewhere between  $N^{1.5}$  and  $N^2$ , where  $N$  is the number of degrees of freedom in the model. The default suggestion for large 2D and 3D models is therefore the iterative GMRES solver. The memory requirement for an iterative solver optimally scales as  $N$ .

GMRES is accelerated by a multigrid method, per default the smoothed aggregation algebraic multigrid (SAAMG) method. The cost of SAAMG is typically very low compared to the number of GMRES iterations necessary if no multigrid method is used. As the name implies, SAAMG builds its coarser meshes algebraically, so the application requires no additional meshes in order to employ SAAMG. In contrast, the geometric multigrid (GMG) method requires actual meshes. If a sufficient number of

multigrid levels can be constructed, GMG is often faster than SAAMG. GMG is also superior for cluster computations and for shared-memory computations with many cores. When the default linear solver is GMRES, an optional, but deactivated, linear solver node is available where GMRES is accelerated by GMG.

Multigrid methods need smoothers, but the saddle point character of the linear system restricts the number of applicable smoothers. The choices are further restricted by the anisotropic meshes frequently encountered in fluid-flow problems. The efficiency of the smoothers is highly dependent on the numerical stabilization. Iterative solvers perform at their best when both [Streamline Diffusion](#) and [Crosswind Diffusion](#) are active.

The default smoother for P1+P1 elements is SCGS. This is an efficient and robust smoother specially designed to solve saddle point systems on meshes that contain anisotropic elements. The SCGS smoother works well even without crosswind diffusion. SCGS can sometimes work for higher-order elements, especially if **Method** in the SCGS settings is set to **Mesh element lines**. But there is no guarantee for this, so the default smoother for higher order elements is an SOR Line smoother. SOR Line handles mesh anisotropy but does not formally address the saddle point character. It does, however, function in practice provided that streamline diffusion and crosswind diffusion are both active.

A different kind of saddle point character can arise if the equation system contains ODE variables. Some advanced boundary conditions can add equations with such variables. These variables must be treated with the Vanka algorithm. SCGS includes an option to invoke Vanka. Models with higher-order elements must apply SCGS or use the Vanka smoother. The latter is the default suggestion for higher-order elements, but it does not work optimally for anisotropic meshes.

## TIME-DEPENDENT SOLVERS

The default time-dependent solver for Navier–Stokes is the BDF method with maximum order set to two. Higher BDF orders are not stable for transport problems in general nor for Navier–Stokes in particular.

BDF methods have been used for a long time and are known for their stability. However, they can have severe damping effects, especially the lower-order methods. Hence, if robustness is not an issue, a model can benefit from using the generalized- $\alpha$  method instead. Generalized- $\alpha$  is a solver that has properties similar to those of the second-order BDF solver but it is much less diffusive.

Both BDF and generalized- $\alpha$  are per default set to automatically adjust the time step. While this works well for many models, extra efficiency and accuracy can often be gained by specifying a maximum time step. It is also often beneficial to specify an initial time step to make the solver progress smoothly in the beginning of the time series.



- [Time-Dependent Solver](#)
- [Multigrid](#), [Direct](#), [Iterative](#), [SCGS](#), [SOR Line](#), and [Vanka](#)
- [Stationary Solver](#)

### *Pseudo Time Stepping for Laminar Flow Models*

A stationary formulation has per definition no time derivatives and [Equation 3-12](#) reduces to:

$$\rho(\mathbf{u} \cdot \nabla)\mathbf{u} = \nabla \cdot [-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)] + \mathbf{F} \quad (3-47)$$

Solving [Equation 3-47](#) requires a starting guess that is close enough to the final solution. If no such guess is at hand, the fully transient problem can be solved instead. This is, however, a rather costly approach in terms of computational time. An intermediate approach is to add a fictitious time derivative to [Equation 3-47](#):

$$\rho \frac{\mathbf{u} - \text{nojac}(\mathbf{u})}{\Delta \tilde{t}} + \rho(\mathbf{u} \cdot \nabla)\mathbf{u} = \nabla \cdot [-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)] + \mathbf{F}$$

where  $\Delta \tilde{t}$  is a *pseudo time step*. Since  $\mathbf{u} - \text{nojac}(\mathbf{u})$  is always zero, this term does not affect the final solution. It does, however, affect the discrete equation system and effectively transforms a nonlinear iteration into a step of size  $\Delta \tilde{t}$  of a time-dependent solver.

Pseudo time stepping is not active per default. The pseudo time step  $\Delta \tilde{t}$  can be chosen individually for each element based on the local CFL number:

$$\Delta \tilde{t} = \text{CFL}_{\text{loc}} \frac{h}{|\mathbf{u}|}$$

where  $h$  is the mesh cell size. A small CFL number means a small time step. It is practical to start with a small CFL number and gradually increase it as the solution approaches steady state.

If the automatic expression for  $\text{CFL}_{\text{loc}}$  is set to the built-in variable `CFLCMP`, then the automatic setting suggests a PID regulator for the pseudo time step in the default

solver. The PID regulator starts with a small CFL number and increases  $CFL_{loc}$  as the solution comes closer to convergence.

The default manual expression is

$$1.3^{\min(\text{niterCMP}, 9)} + \begin{cases} \text{if}(\text{niterCMP} > 20, 9 \cdot 1.3^{\min(\text{niterCMP} - 20, 9)}, 0) + \\ \text{if}(\text{niterCMP} > 40, 90 \cdot 1.3^{\min(\text{niterCMP} - 40, 9)}, 0) \end{cases} \quad (3-48)$$

The variable `niterCMP` is the nonlinear iteration number. It is equal to one for the first nonlinear iteration.  $CFL_{loc}$  starts at 1.3 and increases by 30% each iteration until it reaches  $1.3^9 \approx 10.6$ . It remains there until iteration number 20 at which it starts to increase until it reaches approximately 106. A final increase after iteration number 40 then takes it to 1060. Equation 3-48 can, for some advanced flows, increase  $CFL_{loc}$  too slowly or too quickly.  $CFL_{loc}$  can then be tuned for the specific application.



For details about the CFL regulator, see [Pseudo Time Stepping](#).

### *Nitsche Constraints*

Some boundary conditions are implemented using Nitsche constraints. These boundary conditions include

- [Wall](#) — Slip, sliding walls, and moving walls (that is, walls with nonzero translational velocity).
- [Periodic Flow Condition](#)
- [Flow Continuity](#)

The formulation used in the Fluid Flow interfaces in single-phase flow interfaces corresponds to the Symmetric Interior Penalty Galerkin method (SIPG). The SIPG method can be regarded to satisfy the boundary conditions in an integral sense rather than pointwise. More information on SIPG can be found in [Ref. 15](#).

In particular, the SIPG formulation includes a penalty parameter that must be large enough for the formulation to be coercive. The higher the value, the better the boundary condition is fulfilled, but a too high value results in an ill-conditioned equation system. The penalty parameter is implemented according to [Ref. 16](#).

## Particle Tracing in Fluid Flow

---

The Particle Tracing Module is available to assist with these types of modeling problems.

It is possible to model particle tracing with COMSOL Multiphysics provided that the impact of the particles on the flow field is negligible. First compute the flow field, and then, as an analysis step, calculate the motion of the particles. The motion of a particle is defined by Newton's second law

$$m \frac{d^2 \mathbf{x}}{dt^2} = \mathbf{F} \left( t, \mathbf{x}, \frac{d\mathbf{x}}{dt} \right)$$

where  $\mathbf{x}$  is the position of the particle,  $m$  the particle mass, and  $\mathbf{F}$  is the sum of all forces acting on the particle. Examples of forces acting on a particle in a fluid are the drag force, the buoyancy force, and the gravity force. The drag force represents the force that a fluid exerts on a particle due to a difference in velocity between the fluid and the particle. It includes the viscous drag, the added mass, and the Basset history term. Several empirical expressions have been suggested for the drag force. One of those is the one proposed by Khan and Richardson (Ref. 13). That expression is valid for spherical particles for a wide range of particle Reynolds numbers. The particle Reynolds number is defined as

$$\text{Re}_p = \frac{|\mathbf{u} - \mathbf{u}_p| 2r \rho}{\mu}$$

where  $\mathbf{u}$  is the velocity of the fluid,  $\mathbf{u}_p$  the particle velocity,  $r$  the particle radius,  $\rho$  the fluid density, and  $\mu$  the dynamic viscosity of the fluid. The empirical expression for the drag force according to Khan and Richardson is

$$\mathbf{F} = \pi r^2 \rho |\mathbf{u} - \mathbf{u}_p| (\mathbf{u} - \mathbf{u}_p) [1.84 \text{Re}_p^{-0.31} + 0.293 \text{Re}_p^{0.06}]^{3.45}$$



The model *Flow Past a Cylinder* (Application Library path **COMSOL\_Multiphysics/Fluid\_Dynamics/cylinder\_flow**) demonstrates how to add and set up particle tracing in a plot group using the **Particle Tracing with Mass** node. It uses the predefined Khan–Richardson model for the drag force and neglects gravity and buoyancy forces.

## References for the Single-Phase Flow, Laminar Flow Interfaces

---

1. G.G. Stokes, *Trans. Camb. Phil. Soc.*, vol. 8, pp. 287–305, 1845.
2. P.M. Gresho and R.L. Sani, *Incompressible Flow and the Finite Element Method, Volume 2: Isothermal Laminar Flow*, John Wiley & Sons, 2000.
3. G.K. Batchelor, *An Introduction To Fluid Dynamics*, Cambridge University Press, 1967.
4. R.L. Panton, *Incompressible Flow*, 2nd ed., John Wiley & Sons, 1996.
5. I. Harari and T.J.R. Hughes, “What are  $C$  and  $h$ ? Inequalities for the Analysis and Design of Finite Element Methods,” *Comput. Methods Appl. Mech. Eng.*, vol. 97, pp. 157–192, 1992.
6. Y. Bazilevs, V.M. Calo, T.E. Tezduyar, and T.J.R. Hughes, “ $\Upsilon Z \beta$  Discontinuity Capturing for Advection-dominated Processes with Application to Arterial Drug Delivery,” *Int. J. Num. Meth. Fluids*, vol. 54, pp. 593–608, 2007.
7. R.B. Bird, W.E. Stewart, and E.N. Lightfoot, *Transport Phenomena*, 2nd ed., John Wiley & Sons, 2007.
8. T.J.R. Hughes and M. Mallet, “A New Finite Element Formulation for Computational Fluid Dynamics: III. The Generalized Streamline Operator for Multidimensional Advective-Diffusive System,” *Comput. Methods Appl. Mech. Eng.*, vol. 58, pp. 305–328, 1986.
9. G. Hauke and T.J.R. Hughes, “A Unified Approach to Compressible and Incompressible Flows,” *Comput. Methods Appl. Mech. Eng.*, vol. 113, pp. 389–395, 1994.
10. G. Hauke, “Simple Stabilizing Matrices for the Computation of Compressible Flows in Primitive Variables,” *Comput. Methods Appl. Mech. Eng.*, vol. 190, pp. 6881–6893, 2001.
11. M.-C. Hsu, Y. Bazilevs, V.M. Calo, T.E. Tezduyar, and T.J.R. Hughes, “Improving stability of stabilized and multiscale formulations in flow simulations at small time steps,” *Comput. Methods Appl. Mech. Eng.*, vol. 199, pp. 828–840, 2010.
12. D.J. Tritton, *Physical Fluid Dynamics*, 2nd ed., Oxford University Press, 1988.
13. J.M. Coulson and J.F. Richardson, “Particle Technology and Separation Processes,” *Chemical Engineering, Volume 2*, Butterworth-Heinemann, 2002.

14. J.L. Guermond, P. Mineev, and J. Shen, “An overview of projection methods for incompressible flows,” *Comput. Methods Appl. Mech. Eng.*, vol. 195, pp. 6011–6045, 2006.
15. B. Rivière, *Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations*, SIAM, 2008.
16. Y. Epshteyn and B. Rivière, “Estimation of penalty parameters for symmetric interior penalty Galerkin methods,” *J. Comput. Appl. Math.*, vol. 206, pp. 843–872, 2007.
17. R.P. Chhabra and J.F. Richardson, *Non-Newtonian Flow and Applied Rheology*, 2nd ed., Elsevier, 2008.
18. Y. Bazilevs and T.J.R. Hughes, “Weak imposition of Dirichlet boundary conditions in fluid mechanics,” *Comput. Fluids*, vol. 36, pp. 12–26, 2007.



# Theory for the Rotating Machinery, Laminar Flow Interface

The theory for the Rotating Machinery, Laminar Flow and Rotating Machinery, Turbulent Flow interfaces is described in this section:

- [About Rotating Machinery](#)
- [Frozen Rotor](#)
- [Setting Up a Rotating Machinery Model](#)
- [References](#)

## *About Rotating Machinery*

---

Both the Rotating Machinery, Laminar Flow and Rotating Machinery, Turbulent Flow interfaces model flow in geometries with rotating parts. For example, stirred tanks, mixers, and pumps.

The Navier–Stokes equations formulated in a rotating coordinate system read ([Ref. 1](#) and [Ref. 2](#))

$$\frac{\partial p}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \quad (3-49)$$

$$\begin{aligned} \rho \frac{\partial \mathbf{v}}{\partial t} + \rho(\mathbf{v} \cdot \nabla) \mathbf{v} + 2\rho \Omega \times \mathbf{v} = \\ \nabla \cdot [-p\mathbf{I} + \boldsymbol{\tau}] + \mathbf{F} - \rho \left( \frac{\partial \Omega}{\partial t} \times \mathbf{r} + \Omega \times (\Omega \times \mathbf{r}) \right) \end{aligned} \quad (3-50)$$

where  $\mathbf{v}$  is the velocity vector in the rotating coordinate system,  $\mathbf{r}$  is the position vector, and  $\Omega$  is the angular velocity vector. The relation between  $\mathbf{v}$  and the velocity vector in the stationary coordinate system is

$$\mathbf{u} = \mathbf{v} + \frac{\partial \mathbf{r}}{\partial t}$$

The Rotating Machinery, Fluid Flow interfaces solve [Equation 3-49](#) and [Equation 3-50](#), but reformulated in terms of a nonrotating coordinate system; that is, they solve for  $\mathbf{u}$ . This is achieved by invoking the [Arbitrary Lagrangian–Eulerian](#)

**Formulation (ALE)** machinery. In rotating domains,  $\mathbf{x} = \mathbf{x}(\Omega, t)$  as prescribed in the Rotating Domain under Definitions. The Navier–Stokes equations on rotating domains then read

$$\frac{\partial \rho}{\partial T} - \frac{\partial \mathbf{x}}{\partial T} \cdot \nabla \rho + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (3-51)$$

$$\rho \left( \frac{\partial \mathbf{u}}{\partial T} - \frac{\partial \mathbf{x}}{\partial T} \cdot \nabla \mathbf{u} \right) + \rho (\mathbf{u} \cdot \nabla) \mathbf{u} = \nabla \cdot [-p \mathbf{I} + \boldsymbol{\tau}] + \mathbf{F} \quad (3-52)$$

The derivative operator  $\partial/\partial T$  is the mesh time derivative of the density and appears in the equation view as `d(rmspf.rho, TIME)`. Analogously,  $\partial \mathbf{U}/\partial T$  is the mesh time derivative of the velocity. The variable `TIME` replaces  $t$  as the variable for time.

The user input for a rotating domain prescribes the angular frequency,  $w$ . To calculate  $\Omega$ , the physics interfaces set up an ODE variable for the angular displacement  $\omega$ . The equation for  $\omega$  is

$$\frac{d\omega}{dt} = w \quad (3-53)$$

$\Omega$ , is defined as  $w$  times the normalized axis of rotation. In 2D, the axis of rotation is the  $z$  direction while it in 3D is specified in the Rotating Domain features. If the model contains several rotating domains, each domain has its own angular displacement ODE variable.

In nonrotating domains, the ordinary Navier–Stokes equations are solved. The rotating and fixed parts need to be coupled together by an identity pair, where a continuity boundary condition is applied.

Wall and Interior Wall boundary conditions apply in the rotating domain provided that the Translational velocity under Wall Movement uses the default Automatic from frame option.



Arbitrary Lagrangian–Eulerian Formulation (ALE) in the *COMSOL Multiphysics Reference Manual*

### *Frozen Rotor*

Equation 3-51 through Equation 3-53 must be solved using a Time Dependent study. This can be computationally expensive. The Rotating Machinery, Fluid Flow interfaces

therefore support the so-called frozen rotor approach. The frozen rotor approach assumes that the flow in the rotating domain, expressed in the rotating coordinate system, is fully developed. Equation 3-49 then reduces to

$$\nabla \cdot (\rho \mathbf{v}) = 0 \quad (3-54)$$

and Equation 3-50 to

$$\begin{aligned} \rho(\mathbf{v} \cdot \nabla) \mathbf{v} + 2\rho\Omega \times \mathbf{v} = \\ \nabla \cdot [-p\mathbf{I} + \boldsymbol{\tau}] + \mathbf{F} - \rho\Omega \times (\Omega \times \mathbf{r}) \end{aligned} \quad (3-55)$$

Frozen rotor is both a study type and an equation form. When solving a rotating machinery model using a Frozen Rotor study step, the Rotating Machinery, Fluid Flow interfaces effectively solve Equation 3-54 and Equation 3-55 in a “rotating” domain, but “rotating” domains do not rotate at all. Boundary conditions remain transformed as if the domains were rotating, but the domains remain fixed, or frozen, in position. As in the time-dependent case, the Rotating Machinery, Fluid Flow interfaces solve for the velocity vector in the stationary coordinate system,  $\mathbf{u}$ , rather than for  $\mathbf{v}$ .

To make Equation 3-51 and Equation 3-52 equivalent to Equation 3-54 and Equation 3-55, the Frozen Rotor study step defines a parameter TIME, which by default is set to zero (TIME appears in the Parameters node under Global Definitions). Equation 3-53 is replaced by

$$\omega = w\text{TIME}$$

Since TIME is a parameter and  $\mathbf{x}$  is a function of TIME,  $\partial\mathbf{x}/\partial T$  evaluates to its correct value. Finally,  $\partial p/\partial T = 0$  and the mesh time derivative of the velocity is replaced by

$$\frac{\partial \mathbf{u}}{\partial T} = \Omega \times \mathbf{u}$$

In nonrotating domains, the ordinary, stationary Navier–Stokes equations are solved. The Frozen Rotor study step invokes a stationary solver to solve the resulting equation system.

The frozen rotor approach can in special cases give the same solution as solving Equation 3-51 through Equation 3-53 to steady state. This is the case if, for example, the whole geometry is rotating, or if the model is invariant with respect to the position of the rotating domain relative to the nonrotating domain. The latter is the case for a fan placed in the middle of a straight, cylindrical duct.

In most cases, however, there is no steady-state solution to the rotating machinery problem. Only a pseudo-steady state where the solution varies periodically around some average solution. In those cases, the frozen rotor approach gives an approximate solution to the pseudo-steady state. The approximation depends on the position in which the rotor is frozen, and the method cannot capture transient effects (see [Ref. 3](#) and [Ref. 4](#)). An estimate of the effect of the rotor position can be obtained by making a parametric sweep over TIME.

The frozen rotor approach is very useful for attaining initial values for time-dependent simulations. Starting from a frozen rotor solution, the pseudo-steady state can be reached within a few revolutions, while starting from  $\mathbf{u} = \mathbf{0}$  can require tens of revolutions. See, for example, [Ref. 5](#).

Only interfaces that explicitly support frozen rotors are included in a Frozen Rotor study step.



Studies and Solvers in the *COMSOL Multiphysics Reference Manual*

---

### *Setting Up a Rotating Machinery Model*

---

The Rotating Machinery, Fluid Flow interfaces primarily handle two types of geometries with rotating parts.

The first type is where the whole geometry rotates. Typical examples are individual parts in turbomachinery and lab-on-a-chip devices. For such cases, the selection for the Rotating Domain under the Definitions node should be all domains in the geometry.

The other type is geometries where it is possible to divide the modeled device into rotationally invariant geometries. The operation can be, for example, to rotate an impeller in a baffled tank, as in [Figure 3-1](#) where the impeller rotates from position 1 to 2.

The first step to set up these type of models is to divide the geometry into two parts, as shown in Step 1a. Draw the geometry using separate domains for fixed and rotating parts. If you intend to do a time-dependent simulation, activate the assembly (using an assembly instead of a union, see [Geometry Modeling and CAD Tools](#) in the *COMSOL Multiphysics Reference Manual*) and create identity pairs, which makes it possible to treat the domains as separate parts in an assembly.

The second step is to remove the nonrotating domains from the Rotating Domain selection (Step 1b).

Once this is done, proceed to the usual steps of setting the fluid properties, boundary conditions. Apply a Flow Continuity to assembly pairs (Step 2a). Then mesh and solve the problem.

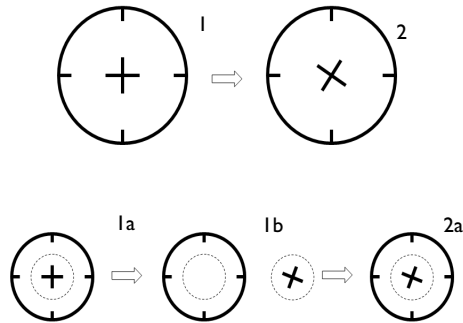


Figure 3-1: The modeling procedure in the Rotating Machinery, Fluid Flow interface.

## References

1. H.P. Greenspan, *The Theory of Rotating Fluids*, Breukelen Press, 1990.
2. G.K. Batchelor, *An Introduction to Fluid Dynamics*, Cambridge University Press, 2000.
3. CFD Online, *Best practice guidelines for turbomachinery CFD*, [https://www.cfd-online.com/Wiki/Best\\_practice\\_guidelines\\_for\\_turbomachinery\\_CFD](https://www.cfd-online.com/Wiki/Best_practice_guidelines_for_turbomachinery_CFD).
4. A. Brucato, M. Ciofalo, F. Grisafi, and G. Micale, “Numerical Prediction of Flow Fields in Baffled Stirred Vessels: A Comparison of Alternative Modelling Approaches”, *Chemical Engineering Science*, vol. 53, no. 21, pp. 3653–3684, 1998.
5. J.-P. Torr , D.F. Fletcher, T. Lasuye, and C. Xuereb, “Single and Multiphase CFD Approaches for Modelling Partially Baffled Stirred Vessels: Comparison of Experimental Data with Numerical Predictions”, *Chemical Engineering Science*, vol. 62, no. 22, pp. 6246–6262, 2007.

# Theory for the Viscoelastic Flow Interface

The theory for the Viscoelastic Flow interface is described in this section:

- [General Viscoelastic Flow Theory](#)
- [Numerical Stability — Stabilization Techniques for Fluid Flow](#)
- [Solvers for Laminar Flow](#)
- [Pseudo Time Stepping for Laminar Flow Models](#)
- [References for the Single-Phase Flow, Laminar Flow Interfaces](#)

## *General Viscoelastic Flow Theory*

---

The Viscoelastic Flow Interface is used to simulate incompressible and isothermal flow of viscoelastic fluids. It solves the continuity equation, the momentum balance equation, and a constitutive equation that defines the extra elastic stress contribution. The continuity and momentum balance can be expressed as

$$\rho \nabla \cdot \mathbf{u} = 0 \quad (3-56)$$

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho (\mathbf{u} \cdot \nabla) \mathbf{u} = \nabla \cdot [-p \mathbf{I} + \boldsymbol{\tau}] + \mathbf{F} \quad (3-57)$$

where  $\boldsymbol{\tau}$  is the extra stress tensor, which is defined as a sum of a viscous and a viscoelastic or elastic contribution as

$$\boldsymbol{\tau} = 2\mu_s \mathbf{S} + \mathbf{T}_e \quad (3-58)$$

where  $\mu_s$  is the solvent viscosity,  $\mathbf{S}$  is the strain-rate tensor, and  $\mathbf{T}_e$  is the elastic (or viscoelastic) stress tensor. To adequately describe a flow of fluid with a complex rheological behavior, the symmetric stress tensor  $\mathbf{T}_e$  is represented as a sum of the individual modes:

$$\mathbf{T}_e = \sum_m \mathbf{T}_{em} \quad (3-59)$$

To close the equation system, the constitutive relation for each mode is required.

The constitutive relation can be formulated using different dependent variables. Several commonly used constitutive models can be written using stress formulation as a hyperbolic partial differential transport equation of the form

$$\frac{1}{2\mu_{em}} \mathbf{f}_{rm}(\mathbf{T}_{em}) + \frac{\lambda_{em}}{2\mu_{em}} \mathbf{T}_{em}^{\nabla} = f_{pm}(\mathbf{T}_{em}) \mathbf{S} \quad (3-60)$$

where the relaxation function  $\mathbf{f}_{rm}$  and the viscosity factor  $f_{pm}$  are model-specific functions of stress,  $\lambda_{em}$  is a relaxation time,  $\mu_{em}$  is a polymer viscosity, and the upper convective derivative operator is defined as

$$\mathbf{T}^{\nabla} \equiv \frac{\partial \mathbf{T}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{T} - [(\nabla \mathbf{u}) \mathbf{T} + \mathbf{T}(\nabla \mathbf{u})^T] \quad (3-61)$$

The first two terms on the right-hand side represent the material derivative, and the other two terms represent the deformation. For more information, see [Ref. 1](#).

#### OLDROYD-B MODEL

For the Oldroyd B model, the relaxation function and the viscosity factor are given by

$$\mathbf{f}_{rm} = \mathbf{T}_{em}, \quad f_{pm} = 1 \quad (3-62)$$

The Oldroyd-B model can be derived from the kinetic theory representing the polymer molecules as suspensions of the Hookean spring in a Newtonian solvent. While demonstrating some basic features of viscoelasticity, the model can only predict a constant shear viscosity and gives unrealistic results for purely extensional flows due to the lack of a mechanism that limits the extensibility.

#### FENE-P MODEL

The finitely extensible nonlinear elastic model (FENE) is based on the kinetic theory that describes the polymer chains as a bead-spring dumbbell and accounts for finite extension of the polymer molecules. The FENE model with Peterlin closure (FENE-P) shows a finite extensibility and a shear-thinning behavior. The expressions for the relaxation function and the viscosity factor are given by

$$\mathbf{f}_{rm} = \left[ 1 + \frac{1}{L_{em}^2} \left( \frac{3}{1 - 3/L_{em}^2} + \frac{\lambda_{em}}{\mu_{em}} \text{tr}(\mathbf{T}_{em}) \right) \right] \mathbf{T}_{em}, \quad f_{pm} = \frac{1}{1 - 3/L_{em}^2} \quad (3-63)$$

where  $L_{em}$  is the extensibility.

### FENE-CR MODEL

Several different closures for FENE types models exists. The expressions for FENE-CR (Finitely Extensible Nonlinear Elastic-Chilcott Rallison) model are given by

$$\begin{aligned}\mathbf{f}_{rm} &= \frac{1}{L_{em}^2 - 3} \left( L_{em}^2 + \frac{\lambda_{em}}{\mu_{em}} \text{tr}(\mathbf{T}_{em}) \right) \mathbf{T}_{em} \\ f_{pm} &= \frac{1}{L_{em}^2 - 3} \left( L_{em}^2 + \frac{\lambda_{em}}{\mu_{em}} \text{tr}(\mathbf{T}_{em}) \right)\end{aligned}\quad (3-64)$$

### ROLIE-POLY MODEL

The Rouse Linear Entangled Polymers (Rolie-Poly) model is a model that derived based on the tube theory. The model incorporates the molecular motion mechanisms of reptation in the tubes, the stretching of polymer chains, and convective constraint release (CCR) due to the release of entanglements as a result of the relaxation of the other chains surrounding the reference chain.

$$\begin{aligned}\mathbf{f}_{rm} &= \mathbf{T}_{em} + 2 \frac{\lambda_{em}}{\lambda_{Rm}} \left( 1 - \frac{1}{\chi_m} \right) \left[ \mathbf{T}_{em} + \frac{\mu_{em}}{\lambda_{em}} \mathbf{I} + \beta_m \chi_m^{2\delta_m} (\mathbf{T}_{em}) \right] \\ f_{pm} &= 1 \\ \chi_m &= \sqrt{1 + \frac{\lambda_{em}}{3\mu_{em}} \text{tr}(\mathbf{T}_{em})}\end{aligned}\quad (3-65)$$

where  $\lambda_{em}$  is the reptation relaxation time,  $\lambda_{Rm}$  is the Rouse relaxation time, parameters  $\beta_m$  and  $\delta_m$  control the convective constrain release.

### GIESEKUS MODEL

The Giesekus model is often used to model the flow of the semi-diluted and concentrated polymers. It adds the quadratic nonlinearity that is attributed to the effect of the hydrodynamic drag induced by the polymer-polymer interactions. The corresponding relaxation function and the viscosity factor are given by

$$\mathbf{f}_{rm} = \left( 1 + \alpha_{em} \frac{\lambda_{em}}{\mu_{em}} \text{tr}(\mathbf{T}_{em}) \right) \mathbf{T}_{em}, \quad f_{pm} = 1 \quad (3-66)$$

where  $\alpha_{em}$  is the dimensionless mobility factor.



### LPTT AND EPTT MODEL

The Phan–Thien–Tanner viscoelastic model is derived from the kinetic theory of an elastic network representing a polymeric melt. The rates of creation and destruction of junctions between the polymer strands depend on stress in the network. The expressions for the relaxation function and the viscosity factor for the linear Phan–Thien–Tanner model (LPTT) are given by

$$\mathbf{f}_{rm} = \left[ 1 + \varepsilon_{em} \frac{\lambda_{em}}{\mu_{em}} \text{tr}(\mathbf{T}_{em}) \right] \mathbf{T}_{em}, \quad f_{pm} = 1 \quad (3-67)$$

where  $\varepsilon_{em}$  is the extensibility. The expressions for the relaxation function for the exponential Phan–Thien–Tanner model (EPTT) is given by

$$\mathbf{f}_{rm} = \exp \left[ \varepsilon_{em} \frac{\lambda_{em}}{\mu_{em}} \text{tr}(\mathbf{T}_{em}) \right] \mathbf{T}_{em} \quad (3-68)$$

### CONFORMATION FORMULATION

The conformation tensor is a macroscopic representation of the polymer molecules structure, for example a representation of the average orientation of the polymer chains. The conformation tensor is a positive definite symmetric tensor that is equal to the identity matrix in the absence of deformation

$$\mathbf{f}_{rm}(\mathbf{C}_{em}) + \lambda_{em} \overset{\nabla}{\mathbf{C}}_{em} = 0 \quad (3-69)$$

where  $\mathbf{f}_{rm}$  is a the relaxation function. The polymer stress is related to the conformation tensor through the strain function

$$\mathbf{T}_{em} = \frac{\mu_{em}}{\lambda_{em}} \mathbf{f}_{sm}(\mathbf{C}_{em}) \quad (3-70)$$

For more information, see [Ref. 3](#). For many popular constitutive models, the strain and relaxation functions are polynomials of  $A$ , usually of first or second orders, with coefficients which may depend of the invariants of  $A$ . The strain and relaxation

functions for the models that are available in the viscoelastic flow interface are given in the [Table 3-2](#).

TABLE 3-2: STRAIN AND RELAXATION FUNCTIONS.

CONSTITUTIVE MODEL	$f_s$	$f_r$
Oldroyd B	$\mathbf{C}_{em} - \mathbf{I}$	$\mathbf{C}_{em} - \mathbf{I}$
FENE-P	$\frac{\mathbf{C}_{em}}{1 - \text{tr}(\mathbf{C}_{em})/L_{em}^2} - \mathbf{I}$	$\frac{\mathbf{C}_{em}}{1 - \text{tr}(\mathbf{C}_{em})/L_{em}^2} - \mathbf{I}$
FENE-CR	$\frac{\mathbf{C}_{em} - \mathbf{I}}{1 - \text{tr}(\mathbf{C}_{em})/L_{em}^2}$	$\frac{\mathbf{C}_{em} - \mathbf{I}}{1 - \text{tr}(\mathbf{C}_{em})/L_{em}^2}$
LPTT	$\mathbf{C}_{em} - \mathbf{I}$	$[1 + \varepsilon_{em} \text{tr}(\mathbf{C}_{em} - \mathbf{I})](\mathbf{C}_{em} - \mathbf{I})$
EPTT	$\mathbf{C}_{em} - \mathbf{I}$	$\exp[\varepsilon_{em} \text{tr}(\mathbf{C}_{em} - \mathbf{I})](\mathbf{C}_{em} - \mathbf{I})$
Giesekus	$\mathbf{C}_{em} - \mathbf{I}$	$\alpha_{em}(\mathbf{C}_{em} - \mathbf{I})^2 + \mathbf{C}_{em} - \mathbf{I}$
Rolie-Poly	$\mathbf{C}_{em} - \mathbf{I}$	$(\mathbf{C}_{em} - \mathbf{I}) + 2\frac{\lambda_{em}}{\lambda_{Rm}}\left(1 - \frac{1}{\chi_m}\right)$ $\left[\mathbf{C}_{em} + \beta_m \chi_m^{2\delta_m}(\mathbf{C}_{em} - \mathbf{I})\right]$

#### CONFORMATION FORMULATION VS ELASTIC TENSOR FORMULATION

Note that, for FENE-CR, FENE-P, and Rolie-Poly models conformation and elastic stress formulations are not fully equivalent. Several terms are not included in the elastic stress formulation. For example, the complete equation for FENE-CR model is

$$\mathbf{f}_{rm} = f_{pm} \left( 1 + \frac{D}{Dt} \left( \frac{1}{f_{pm}} \right) \right) \mathbf{T}_{em}$$

$$f_{pm} = \frac{1}{L_{em}^2 - 3} \left( L_{em}^2 + \frac{\lambda_{em}}{\mu_{em}} \text{tr}(\mathbf{T}_{em}) \right)$$

The time derivative of  $f_{pm}$  is dropped in the elastic formulation in the viscoelastic interface ([Equation 3-64](#)).

**BOUNDARY CONDITIONS**

The theory about boundary conditions is found in the section [Theory for the Single-Phase Flow Interfaces](#). Note that for the viscoelastic models, the extra stress tensor is defined as a sum of a viscous and an elastic contribution:  $\tau = \mathbf{K} + \mathbf{T}_e$ . Therefore, an additional term should be added to the expression for the normal extra stress:  $\mathbf{K}_n = \mathbf{K}n + \mathbf{T}_en$ .

*Numerical Stability — Stabilization Techniques for Fluid Flow*

The constitutive equations ([Equation 3-60](#)) can easily become unstable if discretized using the Galerkin finite element method ([Ref. 2](#)). Stabilized finite element methods are usually necessary in order to obtain physical solutions. The streamline diffusion method is used for the equations for the elastic stress tensor.

The theory about stabilization of the momentum equations [Equation 3-56–Equation 3-57](#) is found in [Numerical Stability — Stabilization Techniques for Fluid Flow](#).



[Numerical Stabilization](#) in the *COMSOL Multiphysics Reference Manual*.

*Solvers for Viscoelastic Flow*

The default solver suggestions for single phase flow interfaces are discussed in [Solvers for Laminar Flow](#). Note that if an iterative solver is used, the separate hybrid preconditioners are used for the velocity group and the viscoelastic stresses.



In the *COMSOL Multiphysics Reference Manual*:

- [Selecting a Preconditioner for an Iterative Linear System Solver](#)

*Pseudo Time Stepping for Viscoelastic Flow Models*

A stationary formulation has per definition no time derivatives. Solving [Equation 3-56](#) through [Equation 3-60](#) requires a starting guess that is close enough to the final solution. If no such guess is at hand, the fully transient problem can be solved instead. This is, however, a rather costly approach in terms of computational time. An

intermediate approach is to add a fictitious time derivative. Further details is found in [Pseudo Time Stepping for Laminar Flow Models](#).

### *References for the Viscoelastic Flow Interfaces*

---

1. R.B. Bird, W.E. Stewart, and E.N. Lightfoot, *Transport Phenomena*, 2nd ed., John Wiley & Sons, 2005.
2. T.J. Craven, J.M. Rees, and W.B. Zimmerman, “Stabilized finite element modelling of Oldroyd-B viscoelastic flow,” *COMSOL Conference 2006*, Birmingham, U.K., 2006.
3. M.A. Alves, P.J.Oliveira, F.T. Pinho, “Numerical Methods for Viscoelastic Fluid Flows”, *Annu. Rev. Fluid Mech.*, vol. 53, pp. 509-541, 2021

# Theory for the Potential Flow Interface

The theory for the Incompressible Potential Flow interface is described in the following section.

## *General Incompressible Potential Flow Theory*

---

The Incompressible Potential Flow interface is used to simulate irrotational, inviscid, and incompressible flow. The continuity equation for incompressible flow can be expressed as a divergence-free constraint for the velocity

$$\nabla \cdot \mathbf{u} = 0 \quad (3-71)$$

Moreover, irrotational flow is described as

$$\nabla \times \mathbf{u} = 0 \quad (3-72)$$

As the curl of the gradient of a scalar field is zero by definition, the velocity can be approximated as the gradient of a scalar velocity potential,  $\phi$ ,

$$\mathbf{u} = -\nabla\phi \quad (3-73)$$

so that [Equation 3-71](#) and [Equation 3-73](#) result in the Laplace equation

$$-\nabla \cdot \nabla\phi = 0 \quad (3-74)$$

The local gauge pressure can be obtained from Bernoulli's equation

$$p = p_{\text{Tot}} - \frac{\rho}{2} \|\nabla\phi\|^2 \quad (3-75)$$


where

$$p_{\text{Tot}} = \frac{\rho}{2} U_{\text{scale}}^2 \quad (3-76)$$

and  $U_{\text{scale}}$  is the user-defined free-stream velocity.



## Multiphase Flow Interfaces

There are several Multiphase Flow interfaces available with the Polymer Flow Module. These are found under the **Fluid Flow>Multiphase Flow** branch (  ) when adding a physics interface. See [Multiphase Flow](#) to help select the physics interface to use.

In this chapter:

- [The Laminar Two-Phase Flow, Level Set and Laminar Two-Phase Flow, Phase Field Interfaces](#)
- [The Two-Phase Flow, Level Set, Brinkman Equations Interface](#)
- [The Laminar Three-Phase Flow, Phase Field Interface](#)
- [The Laminar Two-Phase Flow, Moving Mesh Interface](#)
- [Theory for the Level Set and Phase Field Interfaces](#)
- [Theory for the Three-Phase Flow, Phase Field Interface](#)
- [Theory for the Two-Phase Flow, Moving Mesh Interface](#)

# The Laminar Two-Phase Flow, Level Set and Laminar Two-Phase Flow, Phase Field Interfaces

This section includes the following topics:



- [The Laminar Two-Phase Flow, Level Set Interface](#)
- [The Two-Phase Flow, Level Set Coupling Feature](#)
- [The Wetted Wall Coupling Feature](#)
- [The Interior Wetted Wall Coupling Feature](#)
- [The Laminar Two-Phase Flow, Phase Field Interface](#)
- [The Two-Phase Flow, Phase Field Coupling Feature](#)
- [Domain, Boundary, Point, and Pair Nodes for the Laminar Flow, Two-Phase, Level Set and Phase Field Interfaces](#)



Mathematics, Moving Interfaces

---

## *The Laminar Two-Phase Flow, Level Set Interface*

The **Laminar Two-Phase Flow, Level Set** interface () found under the **Multiphase Flow>Two-Phase Flow, Level Set** branch () is a multiphysics interface designed to track the interface between two immiscible fluids. The flow is assumed to be laminar, that is, to be of low to moderate Reynolds number. The fluids can be incompressible or compressible.

When adding the **Laminar Two-Phase Flow, Level Set** multiphysics interface, a **Laminar Flow** and a **Level Set** interface are added to the Model Builder. In addition, a multiphysics coupling feature, [The Two-Phase Flow, Level Set Coupling Feature](#), is added automatically under the **Multiphysics** node. [The Wetted Wall Coupling Feature](#) is also added, but with empty selection. Other **Wetted Wall** multiphysics coupling features and [The Interior Wetted Wall Coupling Feature](#) can be added when needed to model walls.




Simulations using the Laminar Two-Phase Flow, Level Set interface are always time-dependent since the position of an interface is almost always dependent of its history.



INTERFACE PROPERTIES

Discretization

By default the fluid flow discretization is set to **PI+PI**, and the level set discretization is set to **Linear**. This combination provides exact conservation of the integral of the level set variable when using the nonconservative form of the level set equation.

	<ul style="list-style-type: none"><li>• <a href="#">Theory for the Level Set and Phase Field Interfaces</a></li><li>• <a href="#">Theory of Laminar Flow</a> in the <i>COMSOL Multiphysics Reference Manual</i></li></ul>
---	---

The Two-Phase Flow, Level Set Coupling Feature

The **Laminar Two-Phase Flow, Level Set**,  and **Turbulent Two-Phase Flow, Level Set**,  interfaces contain a multiphysics coupling feature, **Two-Phase Flow, Level Set**, which is added automatically.

The **Two-Phase Flow, Level Set** multiphysics coupling feature defines the density and dynamic viscosity of the fluid used in the **Laminar Flow** interface, and it defines the surface tension on the interface in form of a volume force used in the momentum equation. It also enables the **Level Set** interface to use the velocity field calculated from the **Laminar Flow** interface to transport the interface.

SETTINGS

The **Label** is the default multiphysics coupling feature name.

The **Name** is used primarily as a scope prefix for variables defined by the coupling node. Refer to such variables in expressions using the pattern <name>.<variable\_name>. In order to distinguish between variables belonging to different coupling nodes or 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 multiphysics coupling feature **Two-Phase Flow, Level Set** in the model) is **tpf1**.

## DOMAIN SELECTION

When nodes are added from the context menu, you can select **All domains** (the default) or select **Manual** from the **Selection** list to choose specific domains.

## COUPLED INTERFACES

This section controls which individual interfaces are coupled by the current coupling feature. If a physics interface is deleted and then added to the model again, then in order to reestablish the coupling, you need to choose the correct physics interface again from the **Fluid flow** or **Moving interfaces** lists.

## MODEL INPUTS

Fluid properties of each phase, such as density, viscosity, or the surface tension coefficient, can be defined through user inputs, variables, or by selecting a material. For the latter option, additional inputs, for example temperature or pressure, may be required to define these properties.


### *Temperature*

By default, the **Temperature** model input is set to **Common model input**, and the temperature is controlled from **Default Model Inputs** under **Global Definitions** or by a locally defined **Model Input**. If a Heat Transfer interface is included in the component, it controls the temperature **Common model input**. Alternatively, the temperature field can be selected from another physics interface. All physics interfaces have their own tags (**Name**). For example, if a Heat Transfer in Fluids interface is included in the component, the **Temperature (ht)** option is available for  $T$ .

You can also select **User defined** from the **Temperature** model input in order to manually prescribe  $T$ .

### *Absolute Pressure*

This input appears when a material requires the absolute pressure as a model input. The default **Absolute pressure**  $p_A$  is  $p + p_{\text{ref}}$ , where  $p$  is the dependent pressure variable from the Navier–Stokes or RANS equations, and  $p_{\text{ref}}$  is from the user input defined at the fluid flow physics interface level.

The **Absolute pressure** field can be edited by clicking **Make All Model Inputs Editable** (  ) and entering the desired value in the input field.





Model Inputs and Multiphysics Couplings in the *COMSOL Multiphysics Reference Manual*

---

## MATERIAL PROPERTIES

The effective density and dynamic viscosity can either be **Locally defined** or picked up from a Multiphase Material. When the properties are defined locally, provide the properties for each phase in the respective fluid properties section, and the averaging method in the advanced settings section. If a multiphase material is used, these settings are controlled from the material. Using a multiphase material is advantageous for models coupling two-phase flow with other physics interfaces, such as Heat Transfer in Fluids or Electrostatics. In such cases, the multiphase material can ensure that the effective material properties for these physics interfaces are averaged using the volume fraction from the Level Set physics interface.

Click the **Go to Material** button (  ) to move to the selected material node. Click the **Add Multiphase Material** button (  ) to add a multiphase material with two phases. The added material then becomes the one selected in the **Multiphase Material** list.



Multiphase Material in the *COMSOL Multiphysics Reference Manual*

## FLUID PROPERTIES

A **Fluid Properties** section will be available per phase if the **Material Properties** are set to **Locally defined**. Use the corresponding section to specify the properties of the two fluids. The fluids are denoted **Fluid 1** and **Fluid 2**, respectively.

To specify the properties of **Fluid 1** from a material, select the appropriate material in the **Fluid 1** list. Also make sure that the **Density of fluid 1**  $\rho_1$  and **Dynamic viscosity of fluid 1**  $\mu_1$  are both set to **From Material**.

The density in a material can depend on temperature and/or pressure and these dependencies are automatically replaced by  $p_{\text{ref}}$  and  $T_{\text{ref}}$  for incompressible flows (as specified by the **Compressibility** setting of the fluid flow interface).

The non-Newtonian Power law, Carreau, Carreau–Yasuda, Cross, Cross–Williamson, Bingham–Papanastasiou, Herschel–Bulkley–Papanastasiou, Casson–Papanastasiou, DeKee–Turcotte–Papanastasiou, Robertson–Stiff–Papanastasiou, and Ellis models can alternatively be used to specify the dynamic viscosities of the two fluids.

To instead apply a variable or expression for the density or dynamic viscosity for Fluid A, select **User defined** in the **Density of fluid 1**  $\rho_1$  or the **Dynamic viscosity of fluid 1**  $\mu_1$  list and enter the expression in the corresponding text field.

Similarly, the properties of **Fluid 2** can be specified. The default material is set to **None**.



Care should be taken when using the **Domain Material** setting for the material properties for Fluid 1 and Fluid 2. The material properties are obtained from the domain irrespective of the location of the interface. If two different materials are selected in domains 1 and 2, with the phase boundary initially coincident with the domain boundary, the model has convergence issues once the phase boundary moves away from the domain boundary. This is because a density discontinuity and a viscosity discontinuity occurs at the boundary separating the two fluids. For this reason, selecting the material directly is recommended when setting the material properties for Fluid 1 and Fluid 2.

### ADVANCED SETTINGS

If the **Material Properties** are set to **Locally defined**, you can specify the averaging method for density and dynamic viscosity in this section. The default method for **Density averaging** and **Viscosity averaging** is **Volume average**. In addition to the default method, **Density averaging** can be set to **Heaviside function** or **Harmonic volume average**, while **Viscosity averaging** can be set to **Heaviside function**, **Harmonic volume average**, **Mass average** or **Harmonic mass average**. When a **Heaviside function** is used, enter a value for the corresponding mixing parameter,  $l_p$  or  $l_\mu$ .


### SURFACE TENSION

Select the **Include surface tension force in momentum equation** check box to include the surface tension force in the momentum equation. The surface tension force is not added to porous domains.

Select a **Surface tension coefficient**  $\sigma$  (SI unit: N/m):

- To use a predefined expression, select **Library coefficient, liquid/gas interface** or **Library coefficient, liquid/liquid interface**. Then select an option from the list that displays below (for example, **Water/Air** or **Glycerol/Air**). The predefined correlations are based on the data in [Ref. 7](#), [Ref. 8](#), and [Ref. 9](#).
- For **User defined** enter a value or expression for the surface tension coefficient  $\sigma$  (SI unit: N/m).

## The Wetted Wall Coupling Feature

The **Laminar Two-Phase Flow, Level Set**  interface contains an optional multiphysics coupling boundary feature, **Wetted Wall**. It is an exclusive feature that overrides the **Wall** feature in the Laminar interface as well as the **No Flow** feature in the Level Set interface. It cannot be used in boundaries that have a **Flow Continuity** feature in the flow physics interface or a **Continuity** feature in the level set physics interface.

The **Wetted Wall** boundary condition is suitable for walls in contact with the fluid-fluid interface. When this boundary condition is used, the fluid-fluid interface can move along the wall.

For laminar flow, when the **Wall condition** is set to **Navier slip**, this boundary condition enforces the no-penetration condition  $\mathbf{u} \cdot \mathbf{n}_{\text{wall}} = 0$  and adds a tangential stress on the form

$$\mathbf{K}_{\text{nt}} = -\frac{\mu}{\beta} \mathbf{u}$$

where  $\mathbf{K}_{\text{nt}} = \mathbf{K}_n - (\mathbf{K}_n \cdot \mathbf{n}_{\text{wall}}) \mathbf{n}_{\text{wall}}$ ,  $\mathbf{K}_n = \mathbf{K} \mathbf{n}_{\text{wall}}$ , and  $\mathbf{K}$  is the viscous stress tensor.  $\beta$  is the slip length. For numerical calculations, a suitable choice is  $\beta = h$ , where  $h$  is the mesh element size. The boundary condition does not set the tangential velocity component to zero. However, the extrapolated tangential velocity component is 0 at a distance  $\beta$  outside the wall (see [Figure 4-1](#)). When the **Wall condition** is set to **Nonlinear Navier slip** or **Asymptotic slip**, the frictional force has a nonlinear dependency on the velocity (see [Wall](#)). If the surface tension force is included in the momentum equation in [The Two-Phase Flow, Level Set Coupling Feature](#), the **Wetted Wall** boundary condition also adds the following boundary force to enforce the contact angle:

$$\mathbf{F}_{\theta} = \sigma \delta(\mathbf{n}_{\text{wall}} \cdot \mathbf{n} - \cos \theta_w) \mathbf{n}$$

The contact angle  $\theta_w$  is defined as the angle between the fluid interface and the wall (see [Figure 4-1](#)). For more information, see [Ref. 4](#) and [Ref. 5](#). Note that the contact angle is not enforced on boundaries adjacent to porous domains. In order to properly account for direction and yield stress, when using **Hatzikiriakos slip** in the **Wall condition**, the surface tension contribution has to be combined with the viscous stress in the expression for the slip velocity,

$$\mathbf{u}_{\text{slip}} = - \begin{cases} k_1 \sinh([k_2(\|\mathbf{K}_{\text{nt}}\| - \tau_y)]) \frac{\mathbf{K}_{\text{nt}}}{\|\mathbf{K}_{\text{nt}}\|}, & \text{if } \|\mathbf{K}_{\text{nt}}\| > \tau_y \\ \mathbf{0}, & \text{if } \|\mathbf{K}_{\text{nt}}\| \leq \tau_y \end{cases}$$

$$\mathbf{K}_{\text{nt}} = \mathbf{K}_{\text{n}} - (\mathbf{K}_{\text{n}} \cdot \mathbf{n}_{\text{wall}}) \mathbf{n}_{\text{wall}} - \sigma \delta(\mathbf{n}_{\text{wall}} \cdot \mathbf{n} - \cos \theta_w) (\mathbf{n} \cdot \mathbf{t}_{\text{wall}}) \mathbf{t}_{\text{wall}}$$

where  $\mathbf{K}_{\text{n}} = \mathbf{K} \mathbf{n}_{\text{wall}}$ ,  $\mathbf{K}$  is the viscous stress tensor, and  $\mathbf{t}_{\text{wall}}$  is a unit vector tangential to the wall, in the direction of the projected interface normal.

The contact angle  $\theta_w$  can be defined directly or from Young's equation, which considers the components of the forces in the plane of the surface:

$$\sigma \cos(\theta_w) + \gamma_{s2} = \gamma_{s1} \quad (4-1)$$

where  $\gamma_{s1}$  is the surface energy density on the fluid 1 — solid (wall) interface and  $\gamma_{s2}$  is the surface energy density on the fluid 2 — solid (wall) interface.

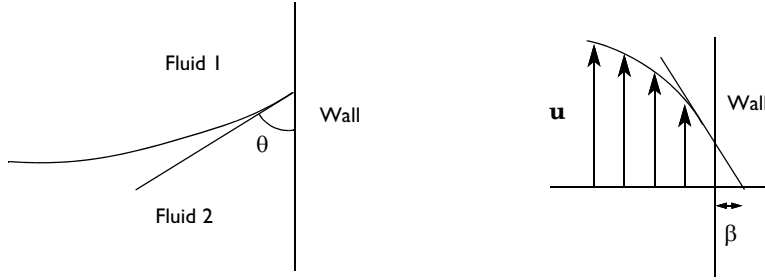


Figure 4-1: Definition of the contact angle  $\theta$  at interface/wall contact points (left) and an illustration of the slip length  $\beta$  (right).

## SETTINGS

The **Label** is the default multiphysics coupling feature name.

The default **Name** (for the first **Wetted Wall** multiphysics coupling feature in the model) is ww1.

## BOUNDARY SELECTION

When nodes are added from the context menu, select **All boundaries** (the default) or select **Manual** from the **Selection** list to choose specific boundaries.

## COUPLED INTERFACES

This section controls which individual interfaces are coupled by the current coupling feature. If a physics interface is deleted and then added to the model again, then in order to reestablish the coupling, you need to choose the correct physics interface again from the **Fluid flow** or **Moving interfaces** lists.

## WALL MOVEMENT

This section contains controls to describe the wall movement relative to the lab (or spatial) frame.

The **Translational velocity** setting controls the translational wall velocity,  $\mathbf{u}_{tr}$ . The list is per default set to **Automatic from frame**. The physics automatically detects if the spatial frame moves. This can for example happen if an ALE interface is present in the model component. If there is no movement  $\mathbf{u}_{tr} = \mathbf{0}$ . If the frame moves,  $\mathbf{u}_{tr}$  becomes equal to the frame movement.

Select **Manual** from **Translational velocity** selection list in order to manually prescribe **Velocity of moving wall**,  $\mathbf{u}_{tr}$ . This can for example be used to model an oscillating wall where the magnitude of the oscillations are very small compared to the rest of the model. Specifying translational velocity manually does not automatically cause the associated wall to move. An additional Moving Mesh interface needs to be added to physically track the wall movement in the spatial reference frame.

**Translational velocity** can also be set to **Zero (Fixed wall)** to force  $\mathbf{u}_{tr} = \mathbf{0}$ . This is typically used in problems where the mesh is deforming or rotating, but we want the boundary to keep zero velocity. For example, in fixed walls tangential to a **Rotating Domain**.

## WETTED WALL

Select the **Wall condition** — **Navier Slip** (the default), **Nonlinear Navier slip**, **Hatzikiriakos slip**, or **Asymptotic slip**. For more details, see [Wall](#).

When using **Navier Slip**, or **Nonlinear Navier slip**, the **Slip Length** setting is per default set to **Factor of minimum element length**. The slip length  $\beta$  is then defined as  $\beta = f_h h_{\min}$ , where  $h_{\min}$  is the smallest element side (corresponds to the element size in the wall normal direction for boundary layer elements) and  $f_h$  is a user input. Select **User defined** from **Slip Length** selection list in order to manually prescribe  $\beta$  (SI unit: m).

If the fluid physics interface includes porous domains, check **Use porous slip wall treatment** to use the porous slip wall treatment to provide tangential stresses and the slip velocity on boundaries adjacent to porous domains. If **Use porous slip wall**

**treatment** is not checked, the condition selected in **Wall condition** is used for porous boundaries.

#### CONTACT ANGLE


Select an option from the **Specify contact angle list** — **Directly** (the default) or **Through Young's equation**. For **Directly** enter a **Contact angle**  $\theta_w$  (SI unit: rad). The default is  $\pi/2$  radians. For **Through Young's equation** enter values or expressions for **Phase 1-Solid surface energy density**  $\gamma_{s1}$  (SI unit: J/m<sup>2</sup>) and **Phase 2-Solid surface energy density**  $\gamma_{s2}$  (SI unit: J/m<sup>2</sup>).



### Theory for the Level Set and Phase Field Interfaces

#### *The Interior Wetted Wall Coupling Feature*

The **Interior Wetted Wall**, multiphysics coupling boundary feature can only be applied on interior boundaries.

It is similar to [The Wetted Wall Coupling Feature](#) available on exterior boundaries of the **Laminar Two-Phase Flow, Level Set**  multiphysics interface, except that it applies on both sides of an interior boundary. It is an exclusive feature which overrides the **Interior Wall** feature in the Laminar interface as well as the **Thin Barrier** feature in the Level Set interface. It is available for laminar flow and turbulent flow with wall functions or automatic wall treatment.

The **Interior Wetted Wall** coupling feature allows discontinuities (velocity, pressure, level set variable) across the boundary and can be used to avoid meshing thin structures in contact with the fluid-fluid interface. When this boundary condition is used, the fluid-fluid interface can move along the wall.

For laminar flow, when the **Wall condition** is set to **Navier slip**, this boundary condition enforces the no-penetration condition  $\mathbf{u} \cdot \mathbf{n}_{\text{wall}} = 0$  on both sides of the wall, and adds a tangential stress on the form

$$\mathbf{K}_{\text{nt}} = -\frac{\mu}{\beta} \mathbf{u}$$

where  $\mathbf{K}_{\text{nt}} = \mathbf{K}_n - (\mathbf{K}_n \cdot \mathbf{n}_{\text{wall}})\mathbf{n}_{\text{wall}}$ ,  $\mathbf{K}_n = \mathbf{K}\mathbf{n}_{\text{wall}}$ , and  $\mathbf{K}$  is the viscous stress tensor.  $\beta$  is the slip length. For numerical calculations, a suitable choice is  $\beta = h$ , where  $h$  is the mesh element size.



If the surface tension force is included in the momentum equation, the **Interior Wetted Wall** boundary condition also adds the following boundary force on both sides of the walls to enforce the contact angle:

$$\mathbf{F}_\theta = \sigma \delta(\mathbf{n}_{\text{wall}} \cdot \mathbf{n} - \cos \theta_w) \mathbf{n}$$

Note that the contact angle is not enforced on boundaries adjacent to porous domains. For more information, see [The Wetted Wall Coupling Feature](#).

## SETTINGS

The **Label** is the default multiphysics coupling feature name.

The default **Name** (for the first **Interior Wetted Wall** multiphysics coupling feature in the model) is `iww1`.

## COUPLED INTERFACES

This section controls which individual interfaces are coupled by the current coupling feature. If a physics interface is deleted and then added to the model again, then in order to reestablish the coupling, you need to choose the correct physics interface again from the **Fluid flow** or **Moving interfaces** lists.

## WALL MOVEMENT

This section contains controls to describe the wall movement relative to the lab (or spatial) frame.

The **Translational velocity** setting controls the translational wall velocity,  $\mathbf{u}_{\text{tr}}$ . The list is per default set to **Automatic from frame**. The physics automatically detects if the spatial frame moves. This can for example happen if an ALE interface is present in the model component. If there is no movement  $\mathbf{u}_{\text{tr}} = \mathbf{0}$ . If the frame moves,  $\mathbf{u}_{\text{tr}}$  becomes equal to the frame movement.

Select **Manual** from **Translational velocity** selection list in order to manually prescribe **Velocity of moving wall**,  $\mathbf{u}_{\text{tr}}$ . This can for example be used to model an oscillating wall where the magnitude of the oscillations are very small compared to the rest of the model. Specifying translational velocity manually does not automatically cause the associated wall to move. An additional Moving Mesh interface needs to be added to physically track the wall movement in the spatial reference frame.

## BOUNDARY SELECTION

When nodes are added from the context menu, select **All boundaries** (the default) or select **Manual** from the **Selection** list to choose specific boundaries.

## WETTED WALL

Select the **Wall condition** — **Navier Slip** (the default), **Nonlinear Navier slip**, **Hatzikiriakos slip**, or **Asymptotic slip**. For more details, see [Wall](#).

When using **Navier Slip**, or **Nonlinear Navier slip**, the **Slip Length** setting is per default set to **Factor of minimum element length**. The slip length  $\beta$  is then defined as  $\beta = f_h h_{\min}$ , where  $h_{\min}$  is the smallest element side (corresponds to the element size in the wall normal direction for boundary layer elements) and  $f_h$  is a user input. Select **User defined** from **Slip Length** selection list in order to manually prescribe  $\beta$  (SI unit: m).

If the fluid physics interface includes porous domains, check **Use porous slip wall treatment** to use the porous slip wall treatment to provide tangential stresses and the slip velocity on boundaries adjacent to porous domains. If **Use porous slip wall treatment** is not checked, the condition selected in **Wall condition** is used for porous boundaries.

## CONTACT ANGLE



Select an option from the **Specify contact angle list** — **Directly** (the default) or **Through Young's equation**. For **Directly** enter a **Contact angle**  $\theta_w$  (SI unit: rad). The default is  $\pi/2$  radians. For **Through Young's equation** enter values or expressions for **Phase 1-Solid surface energy density**  $\gamma_{s1}$  (SI unit: J/m<sup>2</sup>) and **Phase 2-Solid surface energy density**  $\gamma_{s2}$  (SI unit: J/m<sup>2</sup>).



### The Wetted Wall Coupling Feature

---

## *The Laminar Two-Phase Flow, Phase Field Interface*

The **Laminar Two-Phase Flow, Phase Field** interface () found under the **Multiphase Flow>Two-Phase Flow, Phase Field** branch () is a multiphysics interface designed to track the interface between two immiscible fluids. The flow is assumed to be laminar, that is, to be of low to moderate Reynolds number. The fluids can be incompressible or compressible.

When adding the **Laminar Two-Phase Flow, Phase Field** multiphysics interface, a **Laminar Flow** and a **Phase Field in Fluids** interface are added to the Model Builder. In addition, a multiphysics coupling feature, [The Two-Phase Flow, Phase Field Coupling Feature](#), is added automatically under the **Multiphysics** node. Note that unlike the Two-Phase Flow, Level Set interfaces, the wetted wall feature here is not implemented as a separate

multiphase coupling boundary feature but as a default **Wetted Wall** feature in the **Phase Field in Fluids** interface.

	Wetted Wall
---	-------------

Simulations using the Laminar Two-Phase Flow, Phase Field interface are always time-dependent since the position of an interface almost always depends on its history.



**INTERFACE PROPERTIES**

*Discretization*

By default the fluid flow discretization is set to **PI+PI**, and the phase field discretization is set to **Linear**. This combination provides improved conservation of the integral of the phase field variable when using the nonconservative form of the phase field equations.

	Theory for the Two-Phase Flow Interfaces
---	--

*The Two-Phase Flow, Phase Field Coupling Feature*

The **Laminar Two-Phase Flow, Phase Field**,  and **Turbulent Two-Phase Flow, Phase Field**,  interfaces contain a multiphysics coupling feature, **Two-Phase Flow, Phase Field**, which is added automatically.

The **Two-Phase Flow, Phase Field** multiphysics coupling feature defines the density and dynamic viscosity of the fluid used in the **Laminar Flow** interfaces, and it defines the surface tension on the interface in form of a volume force used in the momentum equation. It also enables the **Phase Field in Fluids** interface to use the velocity field calculated from the **Laminar Flow** interface to transport the interface.

**SETTINGS**

The **Label** is the default multiphysics coupling feature name.

The **Name** is used primarily as a scope prefix for variables defined by the coupling node. Refer to such variables in expressions using the pattern <name>.<variable\_name>. In order to distinguish between variables belonging to different coupling nodes or 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 multiphysics coupling feature **Two-Phase Flow, Phase Field** in the model) is `tpf1`.

### DOMAIN SELECTION

When nodes are added from the context menu, you can select **All domains** (the default) or select **Manual** from the **Selection** list to choose specific domains.

### COUPLED INTERFACES

This section controls which individual interfaces are coupled by the current coupling feature. If a physics interface is deleted and then added to the model again, then in order to reestablish the coupling, you need to choose the correct physics interface again from the **Fluid flow** or **Moving interfaces** lists.

### MODEL INPUTS

Fluid properties of each phase, such as density, viscosity, or the surface tension coefficient, can be defined through user inputs, variables, or by selecting a material. For the latter option, additional inputs, for example temperature or pressure, may be required to define these properties.


#### *Temperature*

By default, the **Temperature** model input is set to **Common model input**, and the temperature is controlled from **Default Model Inputs** under **Global Definitions** or by a locally defined **Model Input**. If a Heat Transfer interface is included in the component, it controls the temperature **Common model input**. Alternatively, the temperature field can be selected from another physics interface. All physics interfaces have their own tags (**Name**). For example, if a Heat Transfer in Fluids interface is included in the component, the **Temperature (ht)** option is available for  $T$ .

You can also select **User defined** from the **Temperature** model input in order to manually prescribe  $T$ .

#### *Absolute Pressure*

This input appears when a material requires the absolute pressure as a model input. The default **Absolute pressure**  $p_A$  is  $p + p_{\text{ref}}$ , where  $p$  is the dependent pressure variable from the Navier–Stokes or RANS equations, and  $p_{\text{ref}}$  is from the user input defined at the fluid flow physics interface level.



The **Absolute pressure** field can be edited by clicking **Make All Model Inputs Editable** (  ) and entering the desired value in the input field.



[Model Inputs and Multiphysics Couplings](#) in the *COMSOL Multiphysics Reference Manual*

## MATERIAL PROPERTIES

The effective density and dynamic viscosity can either be **Locally defined** or picked up from a Multiphase Material. When the properties are defined locally, provide the properties for each phase in the respective fluid properties section, and the averaging method in the advanced settings section. If a multiphase material is used, these settings are controlled from the material. Using a multiphase material is advantageous for models coupling two-phase flow with other physics interfaces, such as Heat Transfer in Fluids or Electrostatics. In such cases, the multiphase material can ensure that the effective material properties for these physics interfaces are averaged using the volume fraction from the Phase Field in Fluids physics interface.

Click the **Go to Material** button (  ) to move to the selected material node. Click the **Add Multiphase Material** button (  ) to add a multiphase material with two phases. The added material then becomes the one selected in the **Multiphase Material** list.



[Multiphase Material](#) in the *COMSOL Multiphysics Reference Manual*

## FLUID PROPERTIES

A **Fluid Properties** section will be available per phase if the **Material Properties** are set to **Locally defined**. Use the corresponding section to specify the properties of the two fluids. The fluids are denoted **Fluid 1** and **Fluid 2**, respectively.

To specify the properties of **Fluid 1** from a material, select the appropriate material in the **Fluid 1** list. Also make sure that the **Density of fluid 1**  $\rho_1$  and **Dynamic viscosity of fluid 1**  $\mu_1$  are both set to **From Material**.

The density in a material can depend on temperature and/or pressure and these dependencies are automatically replaced by  $p_{\text{ref}}$  and  $T_{\text{ref}}$  for incompressible flows (as specified by the **Compressibility** setting of the fluid flow interface).

The non-Newtonian Power law, Carreau, Carreau–Yasuda, Cross, Cross–Williamson, Bingham–Papanastasiou, Herschel–Bulkley–Papanastasiou, Casson–Papanastasiou,

DeKee–Turcotte–Papanastasiou, Robertson–Stiff–Papanastasiou, and Ellis models can alternatively be used to specify the dynamic viscosities of the two fluids.

To instead apply a variable or expression for the density or dynamic viscosity for Fluid A, select **User defined** in the **Density of fluid 1**  $\rho_1$  or the **Dynamic viscosity of fluid 1**  $\mu_1$  list and enter the expression in the corresponding text field.

Similarly, the properties of **Fluid 2** can be specified. The default material is set to **None**.



Care should be taken when using the **Domain Material** setting for the material properties for Fluid 1 and Fluid 2. The material properties are obtained from the domain irrespective of the location of the interface. If two different materials are selected in domains 1 and 2, with the phase boundary initially coincident with the domain boundary, the model has convergence issues once the phase boundary moves away from the domain boundary. This is because a density discontinuity and a viscosity discontinuity occurs at the boundary separating the two fluids. For this reason, selecting the material directly is recommended when setting the material properties for Fluid 1 and Fluid 2.

The fluid defined as Fluid 1 affects the wetting characteristics on wetted walls. See the [Wetted Wall](#) node for details.

#### ADVANCED SETTINGS

If the **Material Properties** are set to **Locally defined**, you can specify the averaging method for density and dynamic viscosity in this section. Select the method used for **Density averaging** and **Viscosity averaging**. The default method for both settings is **Volume average**. In addition to the default method, **Density averaging** can be set to **Heaviside function** or **Harmonic volume average**, while **Viscosity averaging** can be set to **Heaviside function**, **Harmonic volume average**, **Mass average** or **Harmonic mass average**. When a **Heaviside function** is used, enter a value for the corresponding mixing parameter,  $l_p$  or  $l_\mu$ .

When the surface tension force is included in the momentum equation, you can select the **Shift surface tension force to the heaviest phase** check box. This can prevent significant spurious oscillations in the velocity field for the lighter phase in cases with a large difference in density between the two phases. The amount of shifting is controlled by the **Smoothing factor**  $d_{s,Fst}$  (default 0.1). The surface tension force is then multiplied by

$$f_s = \frac{2}{(\rho_1 + \rho_2)} \left( \rho_1 H \left( \frac{V_{f,1} - 0.5}{d_{s, Fst}} \right) + \rho_2 H \left( \frac{V_{f,2} - 0.5}{d_{s, Fst}} \right) \right) \quad (4-2)$$

**Shift surface tension force to the heaviest phase** works better when combined with **Heaviside function** for the averaging of density and viscosity.

## SURFACE TENSION

Select the **Include surface tension force in momentum equation** check box to include the surface tension force in the momentum equation. If the surface tension force is included, select **Include surface tension gradient effects in surface tension force** to account for the Marangoni effect due to gradients in the surface tension coefficient.

Select a **Surface tension coefficient**  $\sigma$  (SI unit: N/m):

- To use a predefined expression, select **Library coefficient, liquid/gas interface** or **Library coefficient, liquid/liquid interface**. Then select an option from the list that displays below (for example, **Water/Air** or **Glycerol/Air**). The predefined correlations are based on the data in [Ref. 7](#), [Ref. 8](#), and [Ref. 9](#).
- For **User defined**, enter a value or expression for the surface tension coefficient  $\sigma$  (SI unit: N/m).



Theory for the Level Set and Phase Field Interfaces

## *Domain, Boundary, Point, and Pair Nodes for the Laminar Flow, Two-Phase, Level Set and Phase Field Interfaces*

The Two-Phase Flow, Level Set and Phase Field interfaces are multiphysics interfaces combining Laminar Flow or Turbulent Flow interfaces together with Level Set or Phase Field in Fluids interfaces.

## LAMINAR FLOW

The available physics features for the Laminar Flow interface are listed in the section [Domain, Boundary, Pair, and Point Nodes for Single-Phase Flow](#).

## LEVEL SET

The available physics features for the Level Set interface are listed in the section [Domain, Boundary, and Pair Nodes for the Level Set Interface](#).

## PHASE FIELD IN FLUIDS

The available physics features for the Phase Field in Fluids interface are listed in the section [Domain, Boundary, and Pair Nodes for the Phase Field in Fluids Interface](#).

All domain, boundary, point, and pair nodes are available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or by right-clicking to access the context menu (all users).



# The Two-Phase Flow, Level Set, Brinkman Equations Interface

This Multiphysics interface was designed to track the interface between two immiscible fluids within a porous medium as for example when simulating resin transfer molding.

The tracking is done using the Level Set equation, and the velocity field for this equation can be calculated from the Brinkman Equations.

The immiscible fluids can be incompressible or compressible. The interface is available in 2D, 2D axial symmetric, and 3D.

When adding the **Two-Phase Flow, Level Set, Brinkman Equations** multiphysics interface, a **Brinkman Equations** and a **Level Set** interface are added to the **Model Builder**. In addition, the multiphysics coupling feature **Two-Phase Flow, Level Set** is created automatically under the **Multiphysics** node (see [The Two-Phase Flow, Level Set Coupling Feature](#)). The **Wetted Wall** coupling feature is also added, but with empty selection. Other **Wetted Wall** multiphysics coupling features and the **Interior Wetted Wall** coupling feature can be added when needed to model walls. For more information about these coupling features read [The Wetted Wall Coupling Feature](#) and [The Interior Wetted Wall Coupling Feature](#).

Simulations using the **Two-Phase Flow, Level Set, Brinkman** equations interface are always time-dependent since the position of an interface is almost always dependent of its history.

Note that this multiphysics coupling does not include capillary effects arising inside the porous structure itself as these effects can be neglected for relatively high flow velocities.

Use the following links to get more information about the Brinkman Equations and the Level Set interfaces and the coupling features.



- [The Brinkman Equations Interface](#)
- [Theory for the Brinkman Equations Interface](#)
- [The Laminar Two-Phase Flow, Level Set and Laminar Two-Phase Flow, Phase Field Interfaces](#)
- [Theory for the Level Set and Phase Field Interfaces](#)

# The Laminar Three-Phase Flow, Phase Field Interface

This section includes the following topics:



- [The Laminar Three-Phase Flow, Phase Field Interface](#)
- [The Three-Phase Flow, Phase Field Coupling Feature](#)
- [Domain, Boundary, Point, and Pair Nodes for the Laminar Three-Phase Flow, Phase Field Interface](#)



Mathematics, Moving Interfaces

---

## *The Laminar Three-Phase Flow, Phase Field Interface*

The **Laminar Three-Phase Flow, Phase Field** interface () is found under the **Multiphase Flow>Three-Phase Flow, Phase Field** branch () is a multiphysics interface designed to track the interfaces between three immiscible and incompressible fluids. The flow is assumed to be laminar; that is, a flow characterized by a low-to-moderate Reynolds number. The density of each fluid is assumed to be constant, but there is no restriction on the density differences between the separate fluids.

When adding the **Laminar Three-Phase Flow, Phase Field** multiphysics interface, a **Laminar Flow** and a **Ternary Phase Field** interface are added to the Model Builder. In addition the **Multiphysics** node, including the multiphysics coupling feature **Three-Phase Flow, Phase Field** is added.




[The Multiphysics Branch](#) in the *COMSOL Multiphysics Reference Manual*.

---

Simulations using the Laminar Three-Phase Flow, Phase Field interface are always time-dependent since the position of an interface almost always depends on its history.

## *The Three-Phase Flow, Phase Field Coupling Feature*

---

Use the **Three-Phase Flow, Phase Field**  multiphysics coupling to simulate the flow of a three immiscible fluids while explicitly tracking the interface separating each pair of the fluids.

The Three-Phase Flow interface solves Navier–Stokes equations for the conservation of momentum, and a continuity equation for the conservation of mass. The positions of the interfaces separating the fluid phases are tracked by solving four additional transport equations: two equations governing phase field variables and two equations for the corresponding generalized chemical potentials. The movement of the fluid-fluid interfaces is determined by minimization of free energy.

### **SETTINGS**

The **Label** is the default multiphysics coupling feature name.

The **Name** is used primarily as a scope prefix for variables defined by the coupling node. Refer to such variables in expressions using the pattern <name>.<variable\_name>. In order to distinguish between variables belonging to different coupling nodes or 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 multiphysics coupling feature in the model) is **tfpf1**.

### **DOMAIN SELECTION**

When nodes are added from the context menu, you can select **Manual** (the default) from the **Selection** list to choose specific domains to define the three-phase flow, or select **All domains** as needed.

### **MODEL INPUTS**

Fluid properties of each phase, such as density and viscosity, can be defined through user inputs, variables, or by selecting a material. For the latter option, additional inputs, for example temperature or pressure, may be required to define these properties.

#### *Temperature*


By default, the **Temperature** model input is set to **Common model input**, and the temperature is controlled from [Default Model Inputs](#) under **Global Definitions** or by a locally defined [Model Input](#). If a Heat Transfer interface is included in the component, it controls the temperature **Common model input**. Alternatively, the temperature field can be selected from another physics interface. All physics interfaces have their own

tags (**Name**). For example, if a Heat Transfer in Fluids interface is included in the component, the **Temperature (ht)** option is available for  $T$ .

You can also select **User defined** from the **Temperature** model input in order to manually prescribe  $T$ .

#### *Absolute Pressure*

This input appears when a material requires the absolute pressure as a model input. The default **Absolute pressure**  $p_A$  is  $p + p_{\text{ref}}$ , where  $p$  is the dependent pressure variable from the Navier–Stokes equations, and  $p_{\text{ref}}$  is from the user input defined at the fluid flow physics interface level.

The **Absolute pressure** field can be edited by clicking **Make All Model Inputs Editable** (  ) and entering the desired value in the input field.





[Model Inputs and Multiphysics Couplings](#) in the *COMSOL Multiphysics Reference Manual*

---

### **MATERIAL PROPERTIES**

The effective density and dynamic viscosity can either be **Locally defined** or picked up from a Multiphase Material. When the properties are defined locally, provide the properties for each phase in the respective fluid properties section, and the averaging method in the advanced settings section. If a multiphase material is used, these settings are controlled from the material. Using a multiphase material is advantageous for models coupling two-phase flow with other physics interfaces, such as Heat Transfer in Fluids or Electrostatics. In such cases, the multiphase material can ensure that the effective material properties for these physics interfaces are averaged using the volume fraction from the Ternary Phase Field physics interface.

Click the **Go to Material** button (  ) to move to the selected material node. Click the **Add Multiphase Material** button (  ) to add a multiphase material with three phases. The added material then becomes the one selected in the **Multiphase Material** list.



[Multiphase Material](#) in the *COMSOL Multiphysics Reference Manual*

---

## FLUID PROPERTIES

A **Fluid Properties** section will be available per phase if the **Material Properties** are set to **Locally defined**. Use the corresponding section to specify the properties of all three fluids. The fluids are denoted **Fluid A**, **Fluid B**, and **Fluid C**, respectively.

To specify the properties of **Fluid A** from a material, select the appropriate material in the **Fluid A** list. Also make sure that the **Density of fluid A**  $\rho_A$  and **Dynamic viscosity of fluid A**  $\mu_A$  are both set to **From material**.

The density in a material can depend on temperature and/or pressure and these dependencies are automatically replaced by  $p_{\text{ref}}$  and  $T_{\text{ref}}$  for incompressible flows (as specified by the **Compressibility** setting of the laminar flow interface).

The non-Newtonian Power law, Carreau, Carreau–Yasuda, Cross, Cross–Williamson, Bingham–Papanastasiou, Herschel–Bulkley–Papanastasiou, Casson–Papanastasiou, DeKee–Turcotte–Papanastasiou, Robertson–Stiff–Papanastasiou, and Ellis models can alternatively be used to specify the dynamic viscosities of the three fluids.

To instead apply a variable or expression for the density or dynamic viscosity for Fluid A, select **User defined** in the **Density of fluid A**  $\rho_A$  or the **Dynamic viscosity of fluid A**  $\mu_A$  list and enter the expression in the corresponding text field.

## COUPLED INTERFACES

This section controls which individual interfaces are coupled by the current coupling feature. If a physics interface is deleted and then added to the model again, then in order to reestablish the coupling, you need to choose the correct physics interface again from the **Fluid flow** or **Moving interfaces** lists.



- [Theory for the Level Set and Phase Field Interfaces](#)

### *Domain, Boundary, Point, and Pair Nodes for the Laminar Three-Phase Flow, Phase Field Interface*

The [Laminar Three-Phase Flow, Phase Field Interface](#) is a multiphysics interface combining [The Laminar Flow Interface](#) and [The Ternary Phase Field Interface](#).

## LAMINAR FLOW



The available physics features for [The Laminar Flow Interface](#) are listed in [Domain, Boundary, Pair, and Point Nodes for Single-Phase Flow](#).

## TERNARY PHASE-FIELD

The available physics features for [The Ternary Phase Field Interface](#) are listed in [Domain, Boundary, and Pair Nodes for the Ternary Phase Field Interface](#).

All domain, boundary, point, and pair nodes are available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or by right-clicking to access the context menu (all users).

# The Laminar Two-Phase Flow, Moving Mesh Interface

The **Laminar Two-Phase Flow, Moving Mesh** interface () , found under the **Multiphase Flow>Two-Phase Flow, Moving Mesh** branch () , is a multiphysics interface designed to model laminar two-phase flow of immiscible fluids separated by a moving interface. The velocity field, pressure, and mesh deformation are solved for, and the location of the interface is tracked by the deformed mesh.

When adding the **Laminar Two-Phase Flow, Moving Mesh** multiphysics interface, a **Laminar Flow** interface is added to the component, and a **Moving Mesh** interface with a **Deforming Domain** feature is added under the **Definitions** node. The **Laminar Flow** interface adds the Navier–Stokes equations and provides a way of defining the properties of the fluids. For each of the fluids the density and viscosity must be specified (these are typically taken from the corresponding materials, but can be user defined).

Simulations using the **Laminar Two-Phase Flow, Moving Mesh** interface are always time-dependent since the position of an interface is almost always dependent of its history. There are additional boundary conditions to describe how the fluid-fluid interface moves from the **Laminar Flow** context menu under the option **Fluid Interface Features**:

- [Fluid-Fluid Interface](#)
- [Free Surface](#)
- [Contact Angle](#) (as a subfeature to the two conditions above)



In previous versions of COMSOL Multiphysics (prior to version 5.4), a specific physics interface called **Laminar Two-Phase Flow, Moving Mesh (tpfmm)** was added to the **Model Builder**. Now, a predefined multiphysics coupling approach is used, improving the flexibility and design options for your modeling. For specific details, see [The Multiphysics Branch and Multiphysics Modeling Workflow](#) in the *COMSOL Multiphysics Reference Manual*.

## Select Laminar Flow Properties

---

For a full list of available properties, see [The Single-Phase Flow, Laminar Flow Interface](#) in the *COMSOL Multiphysics Reference Manual*.

### PHYSICAL MODEL

This physics interface deforms the mesh within the domains on either side of the two fluid interfaces to track its movement.

The **Compressibility** defaults to **Incompressible flow** to use the incompressible (constant density) formulation of the Navier–Stokes equations. Select **Compressible flow (Ma<0.3)** to use the compressible formulation.

If flow is occurring at very low Reynolds numbers, the inertial term in the Navier–Stokes equations can be neglected and the linear Stokes equations can be solved on the domain. This flow type is referred to as *creeping flow* or *Stokes flow* and can occur in microfluidics and MEMS devices, where the flow length scales are very small. To make this approximation select the **Neglect inertial term (Stokes flow)** check box, which significantly improves the solver speed.

Enter a **Reference pressure level**  $p_{\text{ref}}$  (SI unit: Pa). The default value is 1 [atm].

### DEPENDENT VARIABLES

The dependent variables (field variables) are for the **Velocity field** and **Pressure**. The names can be changed in the corresponding fields, but the names of fields and dependent variables must be unique within a model.

## Deforming Domain

---

When the **Laminar Two-Phase Flow, Moving Mesh** multiphysics interface is added using the **Model Wizard**, a **Deforming Domain** node is added automatically in the **Model Builder** under the **Definitions>Moving Mesh** node. This node serves to represent domains and boundaries where the mesh can deform. By default, the selection is all domains. Under the **Deforming Domain**, you can also choose the **Mesh smoothing type**, by default set to **Yeoh** with a default **Stiffening factor** of 10. More information of the smoothing type can be found in the [Deforming Domain](#) chapter in the *COMSOL Multiphysics Reference Manual*.

By default, the mesh is fixed at all external boundaries of the geometry. Use the [Symmetry/Roller](#) node to select the external boundaries, where the fluid-fluid interface can slide along the boundary. You can also add other types of boundary



conditions for the mesh motion, for details see [Moving Mesh Features](#) in the *COMSOL Multiphysics Reference Manual*.

## Fluid-Fluid Interface

---

The **Fluid-Fluid Interface** node defines the initial position of a fluid-fluid interface and includes equations to track the evolution of the interface. The [Contact Angle](#) subnode is added by default, and is also available from the context menu (right-click the parent node) or from the **Physics** toolbar, **Attributes** menu.

### MODEL INPUT

Enter a value for the **Temperature**  $T$  (SI unit: K). This option is only necessary when the surface tension coefficient comes from a library of coefficients, as explained below.

### SURFACE TENSION

Select the **Include surface tension force in momentum equation** check box to include the surface tension force in the momentum equation.

The **Surface tension coefficient**  $\sigma$  (SI unit: N/m) can be **User defined**, or select **Library coefficient, liquid/gas interface** (default) or **Library coefficient, liquid/liquid interface**.

- For **Library coefficient, liquid/gas interface** select an option from the list — **Water/Air** (default), **Acetone/Air**, **Acetic acid/Air**, **Ethanol/Air**, **Ethylene glycol/Ethylene glycol vapor**, **Diethyl ether/Air**, **Glycerol/Air**, **Heptane/Nitrogen**, **Mercury/Mercury vapor**, or **Toluene/Air**.
- For **Library coefficient, liquid/liquid interface** select an option from the list — **Benzene/Water, 20°C**, **Corn oil/Water, 20°C**, **Ether/Water, 20°C**, **Hexane/Water, 20°C**, **Mercury/Water, 20°C**, or **Olive oil/Water, 20°C**.



The predefined correlations are based on the data in , , and .

### MASS FLUX

The mass flux setting specifies the mass transfer across the boundary, due to processes such as boiling. The default **Mass Flux**  $\dot{M}_f$  (SI unit: kg/(m<sup>2</sup>·s)) is **User defined**, with a value of 0.

### NORMAL DIRECTION

The normal direction on the selected boundaries can be reversed by selecting **Reverse normal direction**. The direction is indicated by a red arrow in the Graphics window.

	<p>The <b>Fluid-Fluid Interface</b> feature imposes a mesh deformation in the normal direction. On adjacent boundaries, the mesh displacement parallel to those boundaries cannot be fixed if the contact point is moving. In this case a <a href="#">Mesh Slip</a>, <a href="#">Symmetry/Roller</a>, <a href="#">Prescribed Normal Mesh Displacement</a>, or <a href="#">Prescribed Normal Mesh Velocity</a> boundary condition under <b>Definitions&gt;Moving Mesh</b> should be selected. A <a href="#">Mesh Slip</a> condition is automatically used on boundaries that are adjacent to the Fluid-Fluid Interface and do not have a prescribed Moving Mesh boundary condition.</p>
	<p>If a <b>Fluid-Fluid Interface</b> feature is in contact with a solid boundary, the movement of the contact point is not compatible with the no slip condition. For laminar flow the boundary condition with the option should be selected.</p>

### *Free Surface*

The **Free Surface** node defines the initial position of an external fluid interface - that is, an interface in which the viscosity in the fluid outside the domain can be neglected. By applying this feature the motion of the two-phase interface can be tracked. In the tangential direction the free surface motion is governed by the fluid motion in combination with the surface tension of the fluid-fluid interface.

The Free Surface node can only be applied to outer boundaries of a **Deforming Domain** that do not have any boundary condition assigned under **Moving Mesh**.

To control the attachment angle between the free surface and adjacent walls, a [Contact Angle](#) subnode is by default added under the **Free Surface** node. A different contact angle subnode can be added from the context menu (right-click the parent node) or from the **Physics** toolbar, **Attributes** menu.

#### **MODEL INPUT**

Enter a value for the **Temperature**  $T$  (SI unit: K). This option is only necessary when the surface tension coefficient comes from a library of coefficients, as explained below.

## FREE SURFACE

Enter an **External pressure**  $p_{\text{ext}}$  (SI unit: Pa).

## SURFACE TENSION

Select the **Include surface tension force in momentum equation** check box to include the surface tension force in the momentum equation.

The default **Surface tension coefficient**  $\sigma$  (SI unit: N/m) can be **User defined**, or select **Library coefficient, liquid/gas interface** (default) or **Library coefficient, liquid/liquid interface**.

- For **Library coefficient, liquid/gas interface** select an option from the list — **Water/Air** (default), **Acetone/Air**, **Acetic acid/Air**, **Ethanol/Air**, **Ethylene glycol/Ethylene glycol vapor**, **Diethyl ether/Air**, **Glycerol/Air**, **Heptane/Nitrogen**, **Mercury/Mercury vapor**, or **Toluene/Air**.
- For **Library coefficient, liquid/liquid interface** select an option from the list — **Benzene/Water, 20°C**, **Corn oil/Water, 20°C**, **Ether/Water, 20°C**, **Hexane/Water, 20°C**, **Mercury/Water, 20°C**, or **Olive oil/Water, 20°C**.

The predefined correlations are based on the data in [Ref. 5](#), [Ref. 6](#), and [Ref. 7](#).

## MASS FLUX

The mass flux setting specifies the mass transfer across the boundary, due to processes such as boiling. The default **Mass Flux**  $\dot{M}_f$  (SI unit: kg/(m<sup>2</sup>·s)) is **User defined**, with a value of 0.



The **Free Surface** feature imposes a mesh deformation in the normal direction. On adjacent boundaries, the mesh displacement parallel to those boundaries cannot be fixed if the contact point is moving. In this case a [Mesh Slip](#), [Symmetry/Roller](#), [Prescribed Normal Mesh Displacement](#), or [Prescribed Normal Mesh Velocity](#) boundary condition under **Definitions>Moving Mesh** should be selected. A [Mesh Slip](#) condition is automatically used on boundaries that are adjacent to the Free Surface and do not have a prescribed Moving Mesh boundary condition.



If a **Free Surface** feature is in contact with a solid boundary, the movement of the contact point is not compatible with the no slip condition. For laminar flow the **Wall** boundary condition with the **Navier Slip** option should be selected.

## Contact Angle

The **Contact Angle** subnode is available from the context menu (right-click the **Fluid-Fluid Interface Free Surface** parent node) or from the **Physics** toolbar, **Attributes** menu. The condition applies forces on the wall that balance the surface tension for the prescribed contact angle. Note that the contact angle can only be enforced when the surface tension force is included in the momentum equation in the parent feature.

For laminar flow with moving mesh, the **Contact Angle** node should be used on edges or points where a Wall feature with the **Navier Slip** option is prescribed adjacent to the free surface. In this case, the **Navier Slip** option is required for the contact line to move along the wall.

The representation of the contact angle is dependent on the mesh resolution in the region where the fluid-fluid interface attaches to the wall. If, during the simulation, the contact angle is found to fluctuate around a given value, this indicates that the resolution needs to be improved. This can for example be achieved by adding more mesh boundary layers on the wall.

### CONTACT ANGLE

Select an option from the **Specify contact angle list** — **Directly** (the default) or **Through Young's equation**.

- For **Directly** enter a **Contact angle**  $\theta_w$  (SI unit: rad). The default is  $\pi/2$  radians.
- For **Through Young's equation** enter values or expressions for **Phase 1-Solid surface energy density**  $\gamma_{s1}$  (SI unit: J/m<sup>2</sup>) and **Phase 2-Solid surface energy density**  $\gamma_{s2}$  (SI unit: J/m<sup>2</sup>).

### NORMAL WALL VELOCITY

Select whether to **Constrain the wall-normal velocity**. This ensures that the fluid velocity normal to the wall on the contact edge or point is exactly the desired value. This can be helpful when the adjacent wall uses a **Nitsche Constraints** and the no-penetration condition is satisfied in an integral sense rather than pointwise. Moreover, constraining the wall-normal velocity simplifies the equilibrium of forces in the wall-normal

direction. This setting should not be selected when the adjacent boundary has an unknown velocity.



The contact angle  $\theta_w$  is defined between the fluid-fluid interface and the surface of the wall adjacent to phase 1.

---



The moving mesh boundary condition must be compatible with the motion of the wall for the problem to have a valid solution.

---

# Theory for the Level Set and Phase Field Interfaces

The Two-Phase Flow, Level Set and Phase Field interfaces can be used to model the flow of two different, immiscible fluids, where the exact position of the interface between the two fluids is of interest. The fluid-fluid interface is tracked using an auxiliary function on a fixed mesh. These methods account for differences in the two fluids' densities and viscosities and include the effects of surface tension and gravity.

The [Laminar Two-Phase Flow, Level Set and Laminar Two-Phase Flow, Phase Field Interfaces](#) theory is described in this section:

- [Level Set and Phase Field Equations](#)
- [Conservative and Nonconservative Formulations](#)
- [Phase Initialization](#)
- [Numerical Stabilization](#)
- [References for the Level Set and Phase Field Interfaces](#)

## *Level Set and Phase Field Equations*

---

By default, the Level Set and Phase Field interfaces use the incompressible formulation of the Navier–Stokes equations:

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u} \cdot \nabla) \mathbf{u} = \nabla \cdot [-p \mathbf{I} + \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^T)] + \mathbf{F}_g + \mathbf{F}_{st} + \mathbf{F}_{ext} + \mathbf{F} \quad (4-3)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (4-4)$$

Note that [Equation 4-3](#), and [Equation 4-4](#) are solved in the contained interface, **Laminar Flow** or **Turbulent Flow** interface. Note that the form of the continuity equation, appropriate for high density difference mixtures, differs from the definition in [Theory of Laminar Flow](#) in the *COMSOL Multiphysics Reference Manual*.

### USING THE LEVEL SET METHOD

If the level set method is used to track the interface, it adds the following equation:

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = \gamma \nabla \cdot \left( \varepsilon \nabla \phi - \phi(1-\phi) \frac{\nabla \phi}{|\nabla \phi|} \right) \quad (4-5)$$

where  $\gamma$  is the reinitialization parameter (set to 1 by default), and  $\varepsilon$  is the interface thickness controlling parameter (set proportional to  $h_{\max}$  where  $h_{\max}$  is the maximum element size in the component). The density is a function of the level set function. Let  $\rho_1$  and  $\rho_2$  be the constant densities of Fluid 1 and Fluid 2, respectively. Here, Fluid 1 corresponds to the domain where  $\phi < 0.5$ , and Fluid 2 corresponds to the domain where  $\phi > 0.5$ . When **Density averaging** is set to **Volume average**, the density is defined as,

$$\rho = \rho_1 + (\rho_2 - \rho_1)\phi$$

switching to **Heaviside function**, the density is defined as,

$$\rho = \rho_1 + (\rho_2 - \rho_1)H\left(\frac{\phi - 0.5}{l_\rho}\right)$$

where  $H$  is a smooth step function and  $l_\rho$  is a mixing parameter defining the size of the transition zone. When the **Harmonic volume average** is selected,

$$\rho = \frac{1}{(1-\phi)/\rho_1 + \phi/\rho_2}$$

Similarly, the dynamic viscosity can be defined by setting **Viscosity averaging** to **Volume average**,

$$\mu = \mu_1 + (\mu_2 - \mu_1)\phi$$

**Heaviside function**,

$$\mu = \mu_1 + (\mu_2 - \mu_1)H\left(\frac{\phi - 0.5}{l_\mu}\right)$$

**Harmonic volume average**,

$$\mu = \frac{1}{(1-\phi)/\mu_1 + \phi/\mu_2}$$

**Mass average**,

$$\mu = \frac{\rho_1(1-\phi)\mu_1 + \rho_2\phi\mu_2}{\rho}$$

or **Harmonic mass average**,

$$\mu = \frac{\rho}{\rho_1(1-\phi)/\mu_1 + \rho_2\phi/\mu_2}$$

where  $\mu_1$  and  $\mu_2$  are the dynamic viscosities of Fluid 1 and Fluid 2, respectively. For inelastic non-Newtonian fluids,  $\mu_{1,2}$  is replaced by  $\mu_{app1,2}$  in the previous expressions. Further details of the theory for the level set method are in [Ref. 1](#).

#### USING THE PHASE FIELD METHOD

If the phase field method is used to track the interface, it adds the following equations:

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = \nabla \cdot \frac{\gamma \lambda}{\varepsilon^2} \nabla \psi \quad (4-6)$$

$$\psi = -\nabla \cdot \varepsilon^2 \nabla \phi + (\phi^2 - 1)\phi + \left(\frac{\varepsilon^2}{\lambda}\right) \frac{\partial f_{ext}}{\partial \phi} \quad (4-7)$$

where the quantity  $\lambda$  (SI unit: N) is the mixing energy density and  $\varepsilon$  (SI unit: m) is a capillary width that scales with the thickness of the interface. These two parameters are related to the surface tension coefficient,  $\sigma$  (SI unit: N/m), through the equation

$$\sigma = \frac{2\sqrt{2}\lambda}{3\varepsilon}$$

and  $\gamma$  is the mobility parameter which is related to  $\varepsilon$  through  $\gamma = \chi \varepsilon^2$  where  $\chi$  is the mobility tuning parameter (set to 1 by default). The volume fractions of Fluid 1 and Fluid 2 are computed as

$$V_{f,2} = \min(\max([(1+\phi)/2], 0), 1), \quad V_{f,1} = 1 - V_{f,2}$$

where the  $\min$  and  $\max$  operators are used so that the volume fractions have a lower limit of 0 and an upper limit of 1. Let  $\rho_1$  and  $\rho_2$  be the constant densities of Fluid 1 and Fluid 2, respectively. When **Density averaging** is set to **Volume average**, the density is defined as,

$$\rho = \rho_1 V_{f,1} + \rho_2 V_{f,2}$$

switching to **Heaviside function**, the density is defined as,

$$\rho = \rho_1 + (\rho_2 - \rho_1) H\left(\frac{V_{f,2} - 0.5}{l_p}\right)$$



where  $H$  is a smooth step function and  $l_p$  is a mixing parameter defining the size of the transition zone. When the **Harmonic volume average** is selected,

$$\rho = \frac{1}{V_{f,1}/\rho_1 + V_{f,2}/\rho_2}$$

Similarly, the dynamic viscosity can be defined by setting **Viscosity averaging** to **Volume average**,

$$\mu = V_{f,1}\mu_1 + \mu_2 V_{f,2}$$

**Heaviside function**,

$$\mu = \mu_1 + (\mu_2 - \mu_1)H\left(\frac{V_{f,2} - 0.5}{l_\mu}\right)$$

**Harmonic volume average**,

$$\mu = \frac{1}{V_{f,1}/\mu_1 + V_{f,2}/\mu_2}$$

**Mass average**,

$$\mu = \frac{\rho_1 V_{f,1}\mu_1 + \rho_2 V_{f,2}\mu_2}{\rho}$$

or **Harmonic mass average**,

$$\mu = \frac{\rho}{\rho_1 V_{f,1}/\mu_1 + \rho_2 V_{f,2}/\mu_2}$$

where  $\mu_1$  and  $\mu_2$  are the dynamic viscosities of Fluid 1 and Fluid 2, respectively. For inelastic non-Newtonian fluids,  $\mu_{1,2}$  are replaced by  $\mu_{app1,2}$  in the previous expressions.

The mean curvature (SI unit: 1/m) can be computed by entering the following expression:

$$\kappa = 2(1 + \phi)(1 - \phi)\frac{G}{\sigma}$$

where  $G$  is the chemical potential defined as:

$$G = \lambda \left( -\nabla^2 \phi + \frac{\phi(\phi^2 - 1)}{\varepsilon^2} \right) + \frac{\partial f}{\partial \phi}$$

Details of the theory for the phase field method are found in [Ref. 2](#).

### FORCE TERMS

The four forces on the right-hand side of [Equation 4-3](#) are due to gravity, surface tension, a force due to an external contribution to the free energy (using the phase field method only), and a user-defined volume force.

#### *The Surface Tension Force for the Level Set Method*

For the level set method, the surface tension force acting on the interface between the two fluids is:

$$\mathbf{F}_{st} = \sigma \delta \kappa \mathbf{n} + \delta \nabla_s \sigma$$

where  $\sigma$  is the surface tensions coefficient (SI unit: N/m),  $\mathbf{n}$  is the unit normal to the interface, and  $\kappa = -\nabla \cdot \mathbf{n}$  is the curvature.  $\delta$  (SI unit: 1/m) is a Dirac delta function located at the interface.  $\nabla_s$  is the surface gradient operator

$$\nabla_s = (\mathbf{I} - \mathbf{n}\mathbf{n}^T) \nabla$$

The  $\delta$ -function is approximated by a smooth function according to

$$\delta = 6|\nabla \phi| |\phi(1 - \phi)|$$

#### *The Surface Tension Force for the Phase Field Method*

The surface tension force for the phase field method is implemented as a body force

$$\mathbf{F}_{st} = \left( G - \frac{\partial f}{\partial \phi} \right) \nabla \phi$$

where  $G$  is the chemical potential (SI unit: J/m<sup>3</sup>) defined in [Equation 4-3](#) and  $\partial f / \partial \phi$  is a user-defined source of free energy. When **Shift surface tension force to the heaviest phase** is selected the surface-tension force is added as,

$$\mathbf{F}_{st} = \frac{2}{\rho_1 + \rho_2} \left( \rho_1 H \left( \frac{V_{f1} - 0.5}{d_{s,Fst}} \right) + \rho_2 H \left( \frac{V_{f2} - 0.5}{d_{s,Fst}} \right) \right) \left( G - \frac{\partial f}{\partial \phi} \right) \nabla \phi$$

where  $d_{s,Fst}$  is a smoothing factor. This can be used to avoid unphysical acceleration of a light phase near a phase interface caused by the smearing of the phase interface across a few mesh cells.

If **Include surface tension gradient effects in surface tension force** is selected, extra terms are added to account for the Marangoni effect due to gradients in the surface tension coefficient (see [Ref. 6](#)):

$$\mathbf{F}_{st} = \left( G - \frac{\partial f}{\partial \phi} \right) \nabla \phi + \left( \frac{|\nabla \phi|^2}{2 \partial \phi} + \frac{(\phi^2 - 1)^2}{4 \varepsilon^2} \right) \nabla \lambda - (\nabla \lambda \cdot \nabla \phi) \nabla \phi$$

#### *The Gravity Force*

The gravity force is  $\mathbf{F}_g = \rho \mathbf{g}$  where  $\mathbf{g}$  is the gravity vector. Add this as a Gravity feature to the fluid domain.

#### *The User-Defined Volume Force*

When using a Phase Field interface, a force arising due to a user-defined source of free energy is computed according to:

$$\mathbf{F}_{ext} = \left( \frac{\partial f}{\partial \phi} \right) \nabla \phi$$

This force is added when a  $\phi$ -derivative of the external free energy has been defined in the External Free Energy section of the Fluid Properties feature.

### *Conservative and Nonconservative Formulations*

---

When the velocity field is divergence free, you can use either the conservative or the nonconservative formulation of the level set or phase field equation. The conservative form perfectly conserves the mass of each fluid, but the computational time is generally longer.

When the Level Set or Phase Field in Fluids interface are coupled to a fluid flow interface by means of [The Two-Phase Flow, Level Set Coupling Feature](#) or [The Two-Phase Flow, Phase Field Coupling Feature](#), improved conservation of the integral of the level set or phase field variable is obtained using the nonconservative form provided that the discretization order of the level set or phase field variable is equal to or lower than the order of the pressure.

### *Phase Initialization*

---

If the study type Transient with Phase Initialization is used in the model, the level set or phase field variable is automatically initialized. For this study, two study steps are created: Phase Initialization and Time Dependent. The Phase Initialization step solves

for the distance to the initial interface,  $D_{wi}$ . The Time Dependent step then uses the initial condition for the level set function according to the expression

$$\phi_0 = \frac{1}{1 + e^{D_{wi}/\varepsilon}}$$

in domains initially filled with Fluid 1 and

$$\phi_0 = \frac{1}{1 + e^{-D_{wi}/\varepsilon}}$$

in domains initially filled with Fluid 2.

Correspondingly, the phase field method uses the expressions


$$\phi_0 = -\tanh\left(\frac{D_{wi}}{\sqrt{2}\varepsilon}\right)$$

in Fluid 1 and

$$\phi_0 = \tanh\left(\frac{D_{wi}}{\sqrt{2}\varepsilon}\right)$$

in Fluid 2. The initial condition for the help variable is  $\psi_0 = 0$ . These expressions are based on the analytical solution of the steady state solution of [Equation 4-5](#), [Equation 4-6](#), and [Equation 4-7](#) for a straight, nonmoving interface.



If the **Phase Initialization** () study step is being used, for the initialization to work it is crucial that there are two **Initial Values** nodes. One of the Initial Values nodes should use **Specify phase: Fluid 1** and the other **Specify phase: Fluid 2**. The initial fluid-fluid interface is then automatically placed on all interior boundaries between both phases. Note that when the value of **Level set variable** or **Phase field variable** is set to **User defined**, arbitrary user defined initial values have to be provided. In this case, the smooth initialization is not used, and a Time Dependent study should be used instead of [Time Dependent with Phase Initialization](#).




- [The Laminar Two-Phase Flow, Level Set and Laminar Two-Phase Flow, Phase Field Interfaces](#)
- [Studies and Solvers and Time Dependent with Phase Initialization in the COMSOL Multiphysics Reference Manual](#)

---

### *Numerical Stabilization*

---

Three types of stabilization methods are available for the flow (Navier–Stokes) and interface (level set or phase field) equations. Two are consistent stabilization methods — Streamline diffusion and Crosswind diffusion — and one is inconsistent — Isotropic diffusion.

To display this section, click the **Show More Options** button (  ) and select **Stabilization** from the **Show More Options** dialog box.



- [Theory for the Level Set and Phase Field Interfaces](#)
- [Selection Information in the COMSOL Multiphysics Reference Manual](#)

---

### *References for the Level Set and Phase Field Interfaces*

---

1. E. Olsson and G. Kreiss, “A Conservative Level Set Method for Two Phase Flow”, *J. Comput. Phys.*, vol. 210, pp. 225–246, 2005.
2. P. Yue, J.J. Feng, C. Liu, and J. Shen, “A Diffuse-interface Method for Simulating Two-phase Flows of Complex Fluids”, *J. Fluid Mech.*, vol. 515, pp. 293–317, 2004.
3. B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, and G. Zanetti “Modelling Merging and Fragmentation in Multiphase Flows with SURFER.”, *J. Comput. Phys.*, vol. 113, no. 1, pp. 134–147, 1994.
4. Y. D. Shikhmurzaev, “The Moving Contact line on a Smooth Solid Surface”, *Int. J. Multiph. Flow*, vol. 19, no. 4, pp. 589–610, 1993.
5. J. F. Gerbeau and T. Lelièvre, “Generalized Navier Boundary Condition and Geometric Conservation Law for surface tension”, *Comput. Methods Appl. Mech. Eng.*, vol. 198, no. 5-8, pp. 644–656, 2009.
6. C. Liu, J. Shen, J. J. Feng, and P. Yue, “Variational Approach in Two-Phase Flows of Complex Fluids: Transport and Induced Elastic Stress”, Chapter 11 in

*Mathematical Models and Methods in Phase Transitions*, Alan Miranville, ed, Nova Publishers, pp. 259–278, 2005.

7. J. J. Jasper, “The Surface Tension of Pure Liquid Compounds”,  
*J. Phys. Chem. Ref. Data.*, vol. 1, no. 4, pp. 841-1009, 1972.

8. R. C. Weast, ed., “CRC Handbook of Chemistry and Physics”, ed. 69, MIT Press, 1988.

9. A. G. Gaonkar, “Interfacial Tensions of Vegetable Oil/Water Systems: Effect of Oil Purification”, *J. Am. Oil Chem. Soc.*, vol. 66, no. 8, pp. 1090-1092, 1989.

# Theory for the Three-Phase Flow, Phase Field Interface

The Three-Phase Flow, Phase Field interface can be used to model the flow and interaction of three different, immiscible fluids, when the exact positions of the interfaces separating the fluids are of interest. The fluid-fluid interfaces are tracked using a ternary phase field formulation which account for differences in the fluids' densities and viscosities and include the effects of surface tension.

The [Laminar Three-Phase Flow, Phase Field Interface](#) theory is described in the following sections:

- [Governing Equations of the Three-Phase Flow, Phase Field Interface](#)
- [Reference for the Three-Phase Flow, Phase Field Interface](#)

## *Governing Equations of the Three-Phase Flow, Phase Field Interface*

---

### LAMINAR FLOW EQUATIONS

The Three-Phase Flow, Phase Field interface is used to study the evolution of three immiscible fluid phases denoted by fluid *A*, fluid *B*, and fluid *C*. The interface solves the Navier–Stokes equations governing conservation of momentum and mass. The momentum equations solved correspond to

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u} \cdot \nabla) \mathbf{u} = \nabla \cdot \left[ -p \mathbf{I} + \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^T) - \frac{2}{3} \mu(\nabla \cdot \mathbf{u}) \mathbf{I} \right] + \mathbf{F}_{st} + \mathbf{F} \quad (4-8)$$

where  $\mathbf{F}_{st}$  is the surface tension force. The physics interface assumes that the density of each phase is constant. The density may however vary between the phases. In order to accurately handle systems where density of the phases are significantly different, as well as systems with phases of similar density, the following fully compressible continuity equation is solved by the interface

$$\frac{\partial \log \rho}{\partial t} + \nabla \cdot \mathbf{u} + \mathbf{u} \cdot \nabla \log \rho = 0 \quad (4-9)$$

### TERNARY PHASE-FIELD EQUATIONS

In order to track the interfaces between three immiscible fluids, a ternary phase field model based on the work of Boyer and co-workers in [Ref. 1](#) is used. The model solves the following Cahn–Hilliard equations

$$\begin{aligned} \frac{\partial \phi_i}{\partial t} + \nabla \cdot (\mathbf{u} \phi_i) &= \nabla \cdot \left( \frac{M_0}{\Sigma_i} \nabla \eta_i \right) \\ \eta_i &= \frac{4\Sigma_T}{\varepsilon} \sum_{j \neq i} \left( \frac{1}{\Sigma_j} \left( \frac{\partial F}{\partial \phi_i} - \frac{\partial F}{\partial \phi_j} \right) \right) - \frac{3}{4} \varepsilon \Sigma_j \nabla^2 \phi_i \end{aligned} \quad (4-10)$$

governing the phase field variable,  $\phi_i$ , and a generalized chemical potential,  $\eta_i$ , of each phase  $i = A, B, C$ . The phase field variables vary between 0 and 1 and are a measure of the concentration of each phase. The phase field variable for phase  $A$  is one in instances containing only this phase, and zero where there is no phase  $A$ . Since the fluids are immiscible, variations in the phase field variable occur, and define the interface separating two phases. At each point the phase field variables satisfy the following constraint

$$\sum_{i=A,B,C} \phi_i = 1 \quad (4-11)$$

The density of each phase is assumed to be constant which implies that the phase field variable corresponds to the volume fraction of the phase in question. In order to satisfy [Equation 4-11](#), two sets of the equations shown in [Equation 4-10](#) are solved, namely those for phase  $A$  and phase  $B$ . The phase field variable, and correspondingly the mass fraction, for fluid  $C$  is computed from [Equation 4-11](#).

The density and viscosity of the fluid mixture used in [Equation 4-3](#) and [4-4](#) are defined as:

$$\begin{aligned} \rho &= \rho_A \phi_A + \rho_B \phi_B + \rho_C \phi_C \\ \mu &= \mu_A \phi_A + \mu_B \phi_B + \mu_C \phi_C \end{aligned} \quad (4-12)$$

#### Free Energy

The free energy of the three phase system is defined as a function of the phase field variables in the manner of:



$$F = \sigma_{AB}\phi_A^2\phi_B^2 + \sigma_{AC}\phi_A^2\phi_C^2 + \sigma_{BC}\phi_B^2\phi_C^2 + \phi_A\phi_B\phi_C(\Sigma_A\phi_A + \Sigma_B\phi_B + \Sigma_C\phi_C) + \Lambda\phi_A^2\phi_B^2\phi_C^2 \quad (4-13)$$

Here the  $\sigma_{ij}$  denotes the surface tension coefficient of the interface separating phase  $i$  and  $j$ , and the capillary parameters  $\Sigma_i$  are defined for each phase in the manner of:

$$\begin{aligned} \Sigma_A &= \sigma_{AB} + \sigma_{AC} - \sigma_{BC} \\ \Sigma_B &= \sigma_{AB} + \sigma_{BC} - \sigma_{AC} \\ \Sigma_C &= \sigma_{BC} + \sigma_{AC} - \sigma_{AB} \end{aligned} \quad (4-14)$$

$\Lambda$  in the free energy is a function or parameter specifying the additional free bulk energy of the system. By default  $\Lambda$  is zero. In this case it can be seen that the free energy in Equation 4-13 represents the mixing energy, since only interfaces between two phases (where two phase field variables varies between the limiting values) contributes to the free energy.

#### *Ternary Phase Field Parameters*

In Equation 4-10,  $\varepsilon$  (SI unit: 1/m) is a parameter controlling the interface thickness,  $M_0$  (SI unit: m<sup>3</sup>/s) is a molecular mobility parameter, and the parameter  $\Sigma_T$  is defined as

$$\frac{3}{\Sigma_T} = \frac{1}{\Sigma_A} + \frac{1}{\Sigma_B} + \frac{1}{\Sigma_C} \quad (4-15)$$

When using the interface it is recommended that the parameter  $\varepsilon$  is given a value in the same order as the elements in the regions passed by the fluid-fluid interfaces. The mobility determines the time scale of the Cahn–Hilliard diffusion and must be large enough to retain a constant interfacial thickness, but small enough to avoid damping the convective transport. In order to ensure that the mobility is in the correct range, it is recommended to apply a mobility parameter such that the following holds approximately

$$M_0 = L_c U_c \varepsilon \cdot 10^{-2} \quad (4-16)$$

where  $L_c$  and  $U_c$  are the characteristic length and velocity scales of the system at hand.

#### **SURFACE TENSION**

The surface tension force applied in the momentum equations (Equation 4-3) as a body force computed from the generalized chemical potentials:

$$\mathbf{F}_{\text{st}} = \sum_{i=A,B,C} \eta_i \nabla \phi_i \quad (4-17)$$

---

*Reference for the Three-Phase Flow, Phase Field Interface*

I. F. Boyer, C. Lapuerta, S. Minjeaud, B. Piar, and M. Quintard,  
 “Cahn-Hilliard/Navier–Stokes model for the simulation of three-phase flows,”  
*Transport in Porous Media*, vol. 28, pp. 463–484, 2010.

# Theory for the Two-Phase Flow, Moving Mesh Interface

The [Laminar Two-Phase Flow, Moving Mesh Interface](#) can be used to model the flow of two different, immiscible fluids, when the exact position of the interface is of interest. The interface position is tracked by a moving mesh, with boundary conditions that account for surface tension and wetting, as well as mass transport across the interface.

In this section:

- [Domain Level Fluid Flow](#)
- [About the Moving Mesh](#)
- [About the Fluid Interface Boundary Conditions](#)
- [Contact Angle Boundary Conditions](#)
- [References for the Two-Phase Flow, Moving Mesh Interface](#)

## *Domain Level Fluid Flow*

---

Two Phase Flow Moving Mesh interface is a predefined physics interface coupling between a Single-Phase Flow interface and the Moving Mesh interface. Within the domains corresponding to individual phases the fluid flow is solved using the Navier–Stokes equations as in the Laminar Flow interface. The general form for the isothermal Navier–Stokes equations is:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0$$

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u} \cdot \nabla) \mathbf{u} = \nabla \cdot [-p \mathbf{I} + \boldsymbol{\tau}] + \mathbf{F}$$

where:

- $\rho$  is the density (SI unit:  $\text{kg}/\text{m}^3$ )
- $\mathbf{u}$  is the velocity vector (SI unit:  $\text{m}/\text{s}$ )
- $p$  is pressure (SI unit:  $\text{Pa}$ )
- $\mathbf{F}$  is the volume force vector (SI unit:  $\text{N}/\text{m}^3$ )

- $\tau$  is the viscous stress tensor (SI unit: Pa).

For a Newtonian fluid the viscous stress tensor,  $\tau$ , is given by:

$$\tau = 2\mu\mathbf{S} - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I} \quad (4-18)$$

The dynamic viscosity  $\mu$  (SI unit: Pa·s) is allowed to depend on the thermodynamic state but not on the velocity field.



## Theory for the Single-Phase Flow Interfaces

---

### *About the Moving Mesh*

The mesh within the fluid flow domain is deformed to account for the movement of the interface between the two fluids. The software perturbs the mesh nodes so they conform with the moving interface and with other moving or stationary boundaries in the model. The boundary displacement is propagated throughout the domain to obtain a smooth mesh deformation everywhere. This is done by solving equations for the mesh displacements (a Laplace, Winslow, or hyperelastic smoothing equation). Taking two dimensions as an example, a location in the deformed mesh with coordinates  $(x, y)$  can be related to its coordinates in the original undeformed mesh  $(X, Y)$  by a function of the form:

$$x = x(X, Y, t), \quad y = y(X, Y, t)$$

The original, undeformed, mesh is referred to as the *material frame* (or reference frame) whilst the deformed mesh is called the *spatial frame*. COMSOL Multiphysics also defines *geometry* and *mesh frames*, which are coincident with the material frame for this physics interface.

The fluid flow equations (along with other coupled equations such as electric fields or chemical species transport) are solved in the spatial frame in which the mesh is perturbed. The movement of the phase boundary is therefore accounted for in these interfaces.



## Deformed Geometry and Moving Mesh in the *COMSOL Multiphysics Reference Manual*

---

## About the Fluid Interface Boundary Conditions

The interfaces between phases are tracked with specific boundary conditions for the mesh displacement and the fluid flow. Two options are available — [Free Surface](#) and [Fluid-Fluid Interface](#). The Free Surface boundary condition is appropriate when the viscosity of the external fluid is negligible compared to that of the internal fluid. In this case the pressure of the external fluid is the only parameter required to model the fluid and the flow is not solved for in the external fluid. For a fluid-fluid interface the flow is solved for both phases.

### FLUID-FLUID INTERFACE

The boundary conditions applied at an interface between two immiscible fluids, fluid 1 and fluid 2 (see [Figure 4-2](#)), are given by ([Ref. 1](#)):

$$\mathbf{u}_1 = \mathbf{u}_2 + \left( \frac{1}{\rho_1} - \frac{1}{\rho_2} \right) M_f \mathbf{n}_i \quad (4-19)$$

$$\mathbf{n}_i \cdot \boldsymbol{\tau}_2 = \mathbf{n}_i \cdot \boldsymbol{\tau}_1 + \mathbf{f}_{st} \quad (4-20)$$

$$\mathbf{u}_{\text{mesh}} = \left( \mathbf{u}_1 \cdot \mathbf{n}_i - \frac{M_f}{\rho_1} \right) \mathbf{n}_i \quad (4-21)$$

where  $\mathbf{u}_1$  and  $\mathbf{u}_2$  are the velocities of the fluids 1 and 2 respectively,  $\mathbf{u}_{\text{mesh}}$  is the velocity of the mesh at the interface between the two fluids,  $\mathbf{n}_i$  is the normal of the interface (outward from the domain of fluid 1),  $\boldsymbol{\tau}_1$  and  $\boldsymbol{\tau}_2$  are the total stress tensors in domains 1 and 2 respectively,  $\mathbf{f}_{st}$  is the force per unit area due to the surface tension and  $M_f$  is the mass flux across the interface (SI unit:  $\text{kg}/(\text{m}^2\text{s})$ ).

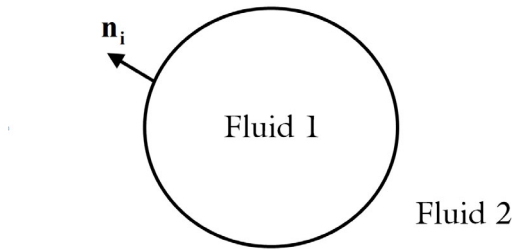


Figure 4-2: Definition of fluid 1 and fluid 2 and the interface normal.

The tangential components of Equation 4-19 enforce a no-slip condition between the fluids at the boundary. In the absence of mass transfer across the boundary, Equation 4-19 and Equation 4-21 ensure that the fluid velocity normal to the boundary is equal to the velocity of the interface. When mass transfer occurs these equations result from conservation of mass and are easily derived in the frame where the boundary is stationary.

The components of the total stress tensor,  $\tau_{uv}$ , represent the  $u^{\text{th}}$  component of the force per unit area perpendicular to the  $v$  direction.  $\mathbf{n} \cdot \boldsymbol{\tau} = \mathbf{n}_v \tau_{uv}$  (using the summation convention) is therefore interpreted as the force per unit area acting on the boundary - in general this is not normal to the boundary. Equation 4-20 therefore expresses the force balance on the interface between the two fluids.



Two boundary conditions (Equation 4-19 and Equation 4-20) are necessary to couple the two domains as there are two separate sets of Navier–Stokes equations, one set for each of the domains.

The force due to the surface tension is given by the following expression:

$$\mathbf{f}_{st} = \sigma(\nabla_s \cdot \mathbf{n}_i) \mathbf{n}_i - \nabla_s \sigma \quad (4-22)$$

where  $\nabla_s$  is the surface gradient operator ( $\nabla_s = (\mathbf{I} - \mathbf{n}_i \mathbf{n}_i^T) \nabla$  where  $\mathbf{I}$  is the identity matrix) and  $\sigma$  is the surface tension at the interface.

In the two dimensional case it is straightforward to see the physical origin of Equation 4-22. Consider an element of the surface of unit depth into the page as shown in Figure 4-3. The normal force per unit area  $\mathbf{F}_n$  on the element in the limit  $\delta s \rightarrow 0$  is:

$$\mathbf{F}_n \delta s = (\sigma + \delta \sigma) \sin(\delta \theta) \rightarrow \sigma \delta \theta$$

Because  $R \delta \theta \rightarrow \delta s$  in the limit,

$$\mathbf{F}_n = \frac{\sigma}{R}$$

The quantity  $\nabla_s \cdot \mathbf{n}_i$  in the first term on the right-hand side of Equation 4-22 is related to the mean curvature,  $\kappa$ , of the surface by the equation  $\kappa = -\nabla_s \cdot \mathbf{n}_i$ . In two dimensions the mean curvature  $\kappa = -1/R$  so  $\nabla_s \cdot \mathbf{n}_i = 1/R$ . The first term in Equation 4-22 is therefore the normal force per unit area acting on the boundary due to the surface tension.

The tangential force per unit area,  $\mathbf{F}_t$ , acting on the interface in [Figure 4-3](#) can be obtained from the force balance along the direction of  $\delta s$  in the limit  $\delta s \rightarrow 0$ :

$$\mathbf{F}_t \delta s = (\sigma + \delta\sigma) \cos(\delta\theta) - \sigma \rightarrow (\sigma + \delta\sigma) - \sigma$$

$$\mathbf{F}_t \rightarrow \frac{(\sigma + \delta\sigma) - \sigma}{\delta s}$$

This is equivalent to the second term on the right of [Equation 4-22](#).

To obtain additional insight into the boundary condition, it is helpful to rewrite [Equation 4-20](#) as

$$\mathbf{n}_i \cdot ((p_1 - p_2) \mathbf{I} - \mu_1 (\nabla \mathbf{u}_1 - (\nabla \mathbf{u}_1)^T) + \mu_2 (\nabla \mathbf{u}_2 - (\nabla \mathbf{u}_2)^T)) = \sigma (\nabla_s \cdot \mathbf{n}_i) \mathbf{n}_i - \nabla_s \sigma$$

assuming Newtonian fluids with viscosities  $\mu_1$  and  $\mu_2$  for fluids 1 and 2, respectively, and that  $p_1$  and  $p_2$  are the pressures in the respective fluids adjacent to the boundary. This equation expresses the equality of two vector quantities. It is instructive to consider the components perpendicular and tangential to the boundary. In the direction of the boundary normal

$$p_1 - p_2 + \mathbf{n}_i \cdot (\mu_2 (\nabla \mathbf{u}_2 - (\nabla \mathbf{u}_2)^T) - \mu_1 (\nabla \mathbf{u}_1 - (\nabla \mathbf{u}_1)^T)) \cdot \mathbf{n}_i = \sigma (\nabla_s \cdot \mathbf{n}_i) \quad (4-23)$$

whereas in the tangential direction,  $\mathbf{t}_i$ ,

$$\mathbf{n}_i \cdot (\mu_2 (\nabla \mathbf{u}_2 - (\nabla \mathbf{u}_2)^T) - \mu_1 (\nabla \mathbf{u}_1 - (\nabla \mathbf{u}_1)^T)) \cdot \mathbf{t}_i = |\nabla_s \sigma| \quad (4-24)$$

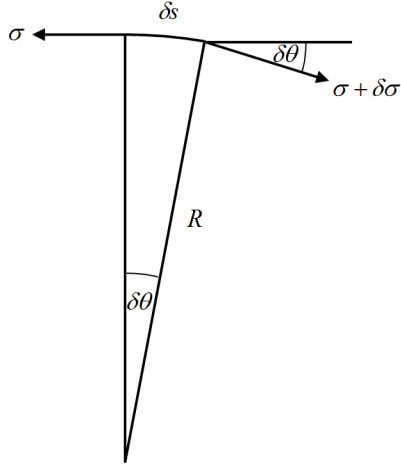


Figure 4-3: Diagrammatic representation of the forces acting on an element of a two-dimensional curved surface.

For the case of a static interface, Equation 4-23 expresses the pressure difference across the interface that results from its curvature. For a moving interface, the surface tension balances the difference in the total normal stress on either side of the interface.

Equation 4-24 involves only velocity gradients and the gradient of the surface tension. The implication of this is that surface tension gradients always drive motion; this is known as the *Marangoni effect*.

### FREE SURFACE

It is often the case that the viscosity of fluid 1 is significantly greater than that of fluid 2 (for example for a liquid-vapor interface). In this case, the viscosity terms in the total stress from fluid 2 can be neglected and Equation 4-20 becomes:

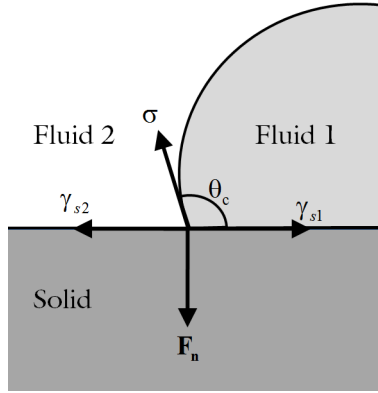
$$\mathbf{n}_i \cdot \boldsymbol{\tau}_1 = -p_{\text{ext}} \mathbf{n}_i + \mathbf{f}_{st} \quad (4-25)$$

The outer fluid (fluid 2) now enters the equation system only through the pressure term and the system can be represented by a domain consisting solely of the fluid 1 domain with an expression (or constant value) for the external pressure,  $p_{\text{ext}}$ , in the fluid 2 domain. Since the velocity in fluid 2 does not affect fluid 1, fluid 2 does not need to be explicitly modeled and Equation 4-19 can be dropped.



## Contact Angle Boundary Conditions

At a three-phase boundary, it is necessary to add force terms on the boundary to ensure that the fluid maintains a consistent contact angle. Additionally the [Wall](#) boundary condition with the [Navier Slip](#) option must be used on the walls. The forces acting on the contact point are applied to the model by the `node`.



*Figure 4-4: Forces per unit length acting on a fluid-fluid interface at a three phase boundary with a solid wall. The surface tension force per unit length,  $\sigma$ , is balanced by a reaction force per unit length at the surface,  $\mathbf{F}_n$ , and by the forces generated by the surface energies of the two phases at the interface:  $\gamma_{s1}$  and  $\gamma_{s2}$ .*

In equilibrium, the surface tension forces and the normal restoring force from the surface are in balance at a constant contact angle ( $\theta_c$ ), as shown in [Figure 4-4](#). This equilibrium is expressed by Young's equation, which considers the components of the forces in the plane of the surface:

$$\sigma \cos(\theta_c) + \gamma_{s1} = \gamma_{s2} \quad (4-26)$$

where  $\sigma$  is the surface tension force between the two fluids,  $\gamma_{s1}$  is the surface energy density on the fluid 1 — solid interface and  $\gamma_{s2}$  is the surface energy density on the fluid 2 — solid interface.

There is still debate in the literature as to precisely what occurs in nonequilibrium situations (for example, drop impact) when the physical contact angle deviates from the contact angle specified by a simple application of Young's equation. A simple approach is to assume that the unbalanced part of the in plane Young Force acts on the fluid to move the contact angle toward its equilibrium value ([Ref. 2](#)). COMSOL

Multiphysics employs this approach as it is physically motivated and is consistent with the allowed form of the boundary condition from thermodynamics (Ref. 3 and Ref. 4).

The normal force balance at the solid surface is handled by the wall boundary condition, which automatically prevents fluid flow across the solid boundary by the application of a constraint force. The contact angle feature applies a force on the fluid at the interface,  $\mathbf{f}_{wf}$ , with magnitude:

$$\mathbf{f}_{wf} = \sigma(\cos(\theta_c) - \cos(\theta))\mathbf{m}_s$$

where  $\theta$  is the actual contact angle and  $\mathbf{m}_s$  is the binormal to the solid surface, as defined in Figure 4-5.

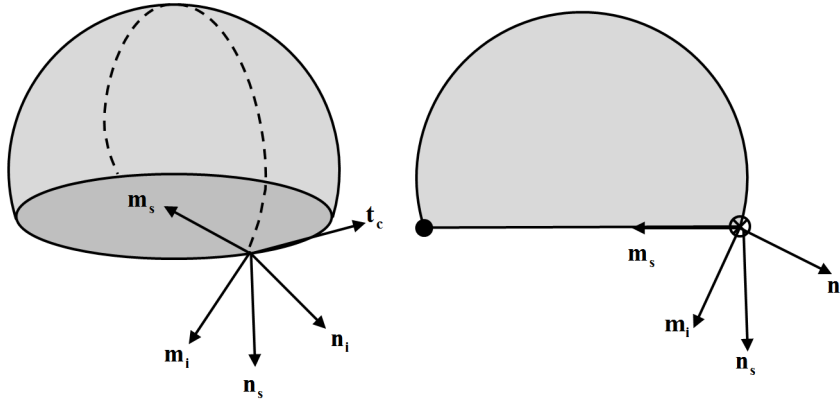


Figure 4-5: Diagram showing normal and tangential vectors defined on the interface between two fluids and a solid surface in 3D (left) and 2D (right). The following vectors are defined:  $\mathbf{n}_i$ , the fluid-fluid interface unit normal;  $\mathbf{n}_s$ , the solid surface unit normal;  $\mathbf{t}_c$ , the tangent to the three-phase contact line; and  $\mathbf{m}_i$  and  $\mathbf{m}_s$ , the two unit binormals, which are defined as  $\mathbf{m}_i = \mathbf{t}_c \times \mathbf{n}_i$  and  $\mathbf{m}_s = \mathbf{t}_c \times \mathbf{n}_s$ , respectively.


#### References for the Two-Phase Flow, Moving Mesh Interface

1. R. Scardovelli and S. Zaleski, “Direct Numerical Simulation of Free Surface and Interfacial Flow,” *Annu. Rev. Fluid Mech.*, vol. 31, pp. 567–603, 1999.
2. J. Gerbeau and T. Lelièvre, “Generalized Navier Boundary Condition and Geometric Conservation Law for Surface Tension,” *Comput. Methods Appl. Mech. Eng.*, vol. 198, pp. 644–656, 2009.

3. W. Ren and E. Weinan, “Boundary Conditions for the Moving Contact Line Problem”, *Phys. Fluids*, vol. 19, p. 022101, 2007.
4. W. Ren and D. Hu, “Continuum Models for the Contact Line Problem”, *Phys. Fluids*, vol. 22, p. 102103, 2010.
5. J. J. Jasper, “The Surface Tension of Pure Liquid Compounds”, *J. Phys. Chem. Ref. Data.*, vol. 1, no. 4, pp. 841-1009, 1972.
6. R. C. Weast, ed., “CRC Handbook of Chemistry and Physics”, ed. 69, MIT Press, 1988.
7. A. G. Gaonkar, “Interfacial Tensions of Vegetable Oil/Water Systems: Effect of Oil Purification”, *J. Am. Oil Chem. Soc.*, vol. 66, no. 8, pp. 1090-1092, 1989.





# Porous Media and Subsurface Flow Interfaces

This chapter discusses physics interfaces under the **Fluid Flow>Porous Media and Subsurface Flow** branch (  ).

In this chapter:

- [The Brinkman Equations Interface](#)
- [The Free and Porous Media Flow, Brinkman Interface](#)
- [Theory for the Brinkman Equations Interface](#)
- [Theory for the Free and Porous Media Flow, Brinkman Interface](#)

# The Brinkman Equations Interface


The **Brinkman Equations (br)** interface (  ), found under the **Porous Media and Subsurface Flow** branch (  ) when adding a physics interface, is used to compute fluid velocity and pressure fields of single-phase flow in porous media in the laminar flow regime. The physics interface extends Darcy's law to describe the dissipation of the kinetic energy by viscous shear, similar to the Navier–Stokes equations. Fluids with varying density can be included at Mach numbers below 0.3. Also the viscosity of a fluid can vary, for example, to describe non-Newtonian fluids. To simplify the equations, select the Stokes–Brinkman flow feature to reduce the dependence on inertial effects when the Reynolds number is significantly less than 1. The physics interface can be used for stationary and time-dependent analyses.

The main node is the Porous Medium feature, which adds the Brinkman equations and provides interfaces for defining the fluid material and the porous matrix.

When this physics interface is added, the following default nodes are also added in the **Model Builder — Porous Medium, Wall** (the default boundary type, using **No slip** as the default boundary condition), and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions and volume forces. You can also right-click **Brinkman Equations** to select physics features from the context menu.

The boundary conditions are essentially the same as for the Laminar Flow interface. Differences exist for the following boundary types: Outlet, Symmetry, Open Boundary, and Boundary Stress where the viscous part of the stress is divided by the porosity to appear as

$$\frac{1}{\varepsilon_p} \left\{ \mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - \frac{2}{3} \mu (\nabla \cdot \mathbf{u}) \mathbf{I} \right\}$$

In the *COMSOL Multiphysics Reference Manual* see [Table 2-4](#) for links to common sections such as **Discretization**, **Consistent Stabilization**, **Inconsistent Stabilization**, and **Advanced Settings**, all accessed by clicking the **Show** button (  ) and choosing the applicable option. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

## 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 `br`.

## PHYSICAL MODEL

This node specifies the properties of the Brinkman Equations interface, which describe the overall type of fluid flow model.

### *Compressibility*

By default the physics interface uses the **Incompressible flow** formulation of the Brinkman equations to model constant density flow. Alternatively, from the **Compressibility** list select **Weakly compressible flow**, which should be used when the pressure dependency of the density can be neglected, or **Compressible flow (Ma<0.3)** to solve for the full compressible flow of which, however, the Mach number must be below 0.3.

### *Swirl Flow*

For 2D axisymmetric models, select the **Swirl flow** check box to include the swirl velocity component, that is the velocity component  $u_\varphi$  in the azimuthal direction. While  $u_\varphi$  can be nonzero, there can be no gradients in the  $\varphi$  direction.

Note that this feature is only available for specific modules. See <https://www.comsol.com/products/specifications/> for a detailed overview.

### *Neglect Inertial Term (Stokes–Brinkman Flow)*

The **Neglect inertial term (Stokes–Brinkman)** check box is selected by default to model flow at low Reynolds numbers for which the inertial term can be neglected. This results in the linear Stokes–Brinkman equations.

### *Include Gravity*

Gravity is not included by default. Select the **Include gravity** check box to activate the acceleration of gravity. This automatically adds a global **Gravity** feature node to the interface model tree, and the buoyancy force is included in the Equations.

Also, when the **Include gravity** check box is selected, the **Use reduced pressure** option changes the pressure formulation from using the total pressure (default) to using the reduced pressure. This option is suitable for configurations where the density changes

are very small; otherwise, the default formulation can be used. For more information, see [Gravity](#).

#### *Porous Treatment of No Slip Condition*

Choose how the **No Slip** condition on **Wall** boundaries and **Interior Wall** boundaries adjacent to **Porous Medium** should be treated. The options are **Standard no slip formulation** and **Porous slip** (default). The latter option provides a unified treatment when the porous matrix is fully resolved as well as when it is under resolved ensuring a smooth transition between regions with different resolutions; see No Slip under [Wall](#) in the [The Creeping Flow and Laminar Flow Interfaces](#) section.

#### *Reference Pressure Level*

Enter a **Reference pressure level**  $p_{\text{ref}}$  (SI unit: Pa). The default value is 1 [atm].

#### *Reference Temperature Level*

Enter a **Reference temperature level**  $T_{\text{ref}}$  (SI unit: K). The default value is 293.15 [K].

#### *Reference Position*

If **Include gravity** is selected, a **Reference position**  $r_{\text{ref}}$  (SI unit: m) can be specified which is then used for the calculation of the hydrostatic pressure.


### **TURBULENCE**

Turbulent flow can be simulated by changing the **Turbulence model type** to **RANS** (Reynolds-Averaged Navier–Stokes). If turbulent flow is activated, you can set the **Turbulence model** either to one of the algebraic turbulence models, **Algebraic yPlus** (the default) or **L-VEL**, or to one of the following two-equation turbulence models:  $k$ - $\epsilon$ , Low Reynolds Number  $k$ - $\epsilon$ . The **Wall treatment** is set to **Automatic** (default) or **Low Re** while **Wall functions** is the only option for  $k$ - $\epsilon$ . For more information about turbulence modeling, see [Theory for the Turbulent Flow Interfaces](#) in the *CFD Module User's Guide*.

### **DEPENDENT VARIABLES**


The following dependent variables (fields) are defined for this physics interface — the **Velocity field**  $\mathbf{u}$  (SI unit: m/s) and its components, and the **Pressure**  $p$  (SI unit: Pa).

### **ADVANCED SETTINGS**

To display this section, click the **Show More Options** button (  ) and select **Advanced Physics Options** in the **Show More Options** dialog box. Normally these settings do not need to be changed.




The **Use pseudo time stepping for stationary equation form** option adds pseudo time derivatives to the equation when the **Stationary equation** form is used in order to speed up convergence. When selected, a **CFL number expression** should also be defined. For the default **Automatic** option, the local CFL number (from the Courant–Friedrichs–Lewy condition) is determined by a PID regulator.

<div>  </div>	<ul style="list-style-type: none"> <li>• <a href="#">Domain, Boundary, Point, and Pair Nodes for the Brinkman Equations Interface</a></li> <li>• <a href="#">Theory for the Brinkman Equations Interface</a></li> <li>• <a href="#">Pseudo Time Stepping for Laminar Flow Models and Pseudo Time Stepping</a></li> <li>• <a href="#">Numerical Stability — Stabilization Techniques for Fluid Flow</a></li> <li>• <a href="#">Nitsche Constraints</a></li> </ul>
--	--

### *Domain, Boundary, Point, and Pair Nodes for the Brinkman Equations Interface*

The [Brinkman Equations Interface](#) has the following domain, boundary, point, and pair nodes, listed in alphabetical order, 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).

<div>  </div>	<p>In general, to add a node, go to the <b>Physics</b> toolbar, no matter what operating system you are using. Subnodes are available by clicking the parent node and selecting it from the <b>Attributes</b> menu.</p>
--	---

These nodes are described in this section:

- [Porous Medium](#)
  - [Initial Values](#)
  - [Mass Source](#)
  - [Volume Force](#)
  - [Fluid Properties](#)

The following nodes (listed in alphabetical order) are described for the Laminar Flow interface :

- [Flow Continuity](#)
- [Inlet](#)
- [Line Mass Source](#)
- [Boundary Stress](#)
- [Outlet](#)
- [Open Boundary](#)
- [Periodic Flow Condition](#)
- [Point Mass Source](#)
- [Pressure Point Constraint](#)
- [Symmetry](#)
- [Wall](#)



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.

## *Porous Medium*

The **Porous Medium** node adds the Brinkman equations: [Equation 4-34](#) and [Equation 4-35](#) (excluding any mass sources). It has two default subfeatures — the [Fluid](#) subfeature where the fluid density  $\rho$  (SI unit:  $\text{kg}/\text{m}^3$ ) and dynamic viscosity  $\mu$  (SI unit:  $\text{Pa}\cdot\text{s}$ ) are defined and the [Porous Matrix](#) subfeature where the permeability  $\kappa$  (SI unit:  $\text{m}^2$ ) and porosity  $\varepsilon_p$  (dimensionless) are specified.

### COORDINATE SYSTEM SELECTION

Select a coordinate system from the **Coordinate system** list for the interpretation of directions in anisotropic material properties. The default is the **Global coordinate system**, and the list contains any additional orthonormal coordinate system (except boundary coordinate systems) added under the Definitions node.

The subnodes inherit these coordinate system settings. In particular, the **Permeability** (in the **Porous Matrix** subnode) should be set according to the coordinate system selected in this section.

## FLOW MODEL

Define if the flow follows a linear or nonlinear pressure-velocity relationship.

- **Darcian flow** (default) defines a linear relationship characteristic for low Reynolds number laminar flows.
- **Non-Darcian flow** defines a nonlinear relationship due to inertial or turbulence effects.

## POROUS MEDIUM TURBULENCE MODEL

If in the parent feature node a turbulence model other than **Algebraic yPlus** or **L-VEL** is selected, define a model which complements the basic turbulence model and accounts for the influence of the porous matrix on turbulent flow in porous media.

- For **Default** and **Nakayama-Kuwahara** specify the **Coefficient of subgrid turbulence generation by porous matrix**  $C_{\text{stpm}}$  which by default is set to 0.212.
- For **Pedras-de Lemos**, specify the **Choice of the Pedras-de Lemos model coefficient**. When **User defined** is chosen, specify the **Coefficient of subgrid turbulence generation by porous matrix**  $C_{\text{stpm}}$  with default value 0.212. Choosing **Original** results in  $C_{\text{stpm}} = 0.28 \cdot \sqrt{\varepsilon_p}$ , while choosing **Recalibrated** results in  $C_{\text{stpm}} = 0.18$ .

## DISTANCE EQUATION

This section is available for the turbulence models **Algebraic yPlus** or **L-VEL**. Select how the **Reference length scale**  $l_{\text{ref}}$  (SI unit: m) is defined — **Automatic** (default) or **Manual**:

- For **Automatic** the wall distance is automatically evaluated to one tenth of the shortest side of the geometry bounding box. This is usually quite accurate but it can sometimes give a too high value if the geometry consists of several slim entities. In such cases, define the reference length scale manually.
- For **Manual** it defines a different value or expression for the length scale. The default is 1 m.

$l_{\text{ref}}$  controls the result of the distance equation. Objects that are much smaller than  $l_{\text{ref}}$  are effectively be diminished while the distance to objects much larger than  $l_{\text{ref}}$  are accurately represented.

## *Fluid*

This node defines the material properties of the fluid of the **Porous Medium** parent node.

## MODEL INPUT

This section contains fields and values that are inputs for expressions defining material properties. If such user-defined property groups are added, the model inputs appear here.

## FLUID PROPERTIES

Specify the **Density**  $\rho$  and the **Dynamic viscosity**  $\mu$  (SI unit: Pa·s) of the fluid. The defaults use values **From material**, for **User defined** enter a value or expression.

The dynamic viscosity describes the relationship between the shear stresses and the shear rate in a fluid. Intuitively, water and air have a low viscosity, and substances often described as thick, such as oil, have a higher viscosity.

## *Porous Matrix*

---

This node defines the porosity and material properties of the solid matrix of the **Porous Medium** parent node.

## MODEL INPUT

This section contains fields and values that are inputs for expressions defining material properties. If such user-defined property groups are added, the model inputs appear here.

## MATRIX PROPERTIES

The **Porosity**  $\varepsilon_p$  (a dimensionless number between 0 and 1) uses by default the value **From material**. For **User defined** the default value is 1.

Select a **Permeability model** to specify the capacity of the porous material to transmit flow. Depending on the **Flow model** selection in the parent **Porous Medium** feature node you can choose between the following options:

- **Permeability** (the default) to directly enter the permeability of the porous matrix or **Kozeny-Carman** to define the permeability from the porosity and mean **Particle diameter** for **Darcian** flow or
- **Forchheimer** or **Ergun** for **Non-Darcian** flow.

Note that some of these options are only available for certain modules. For a detailed overview of the functionality available in each product, visit <https://www.comsol.com/products/specifications/>.

The default **Permeability**  $\kappa$  (SI unit:  $\text{m}^2$ ) uses the value **From material**, as defined by the **Porous material** list. For **User defined** select **Isotropic** to define a scalar value or **Diagonal** or **Symmetric** to define a tensor value.

For **Kozeny-Carman**, enter the mean **Particle diameter**  $d_p$  (SI unit: m), the default value is 0.5 mm.

In case a **Non-Darcian** flow model is chosen in the parent node, **Forchheimer** is the default selection for the **Permeability model**. For **Forchheimer**, the default **Permeability**  $\kappa$  (SI unit:  $\text{m}^2$ ) uses the value **From material**. Furthermore, the dimensionless **Forchheimer parameter**  $c_F$  can be defined. The default value is 0.55.

If **Ergun** is selected, enter the mean **Particle diameter**  $d_p$  (default value: 0.5 mm). The permeability  $\kappa$  is then calculated using [Equation 2-4](#).



- [About Darcian and Non-Darcian Flow](#) in the *Porous Media Flow Module User's Guide*
- [Permeability Models](#) in the *Porous Media Flow Module User's Guide*

## Mass Source

The **Mass Source** node adds a mass source (or mass sink)  $Q_m$  to the right-hand side of the continuity equation: [Equation 5-1](#). This term accounts for mass deposit and/or mass creation in porous domains. The physics interface assumes that the mass exchange occurs at zero velocity.

$$\frac{\partial}{\partial t}(\epsilon_p \rho) + \nabla \cdot (\rho \mathbf{u}) = Q_m \quad (5-1)$$

### DOMAIN SELECTION

Only Porous Matrix domains are available.

### MASS SOURCE

Enter a value or expression for the **Source term**  $Q_{br}$  (SI unit:  $\text{kg}/(\text{m}^3 \cdot \text{s})$ ). The default is  $0 \text{ kg}/(\text{m}^3 \cdot \text{s})$ .

## Volume Force

Use the **Volume Force** node to specify the force  $\mathbf{F}$  on the right-hand side of [Equation 5-2](#). It then acts on each fluid element in the specified domains.

$$\frac{\rho}{\varepsilon_p} \left( \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \frac{\mathbf{u}}{\varepsilon_p} \right) = -\nabla p + \nabla \cdot \left[ \frac{1}{\varepsilon_p} \left\{ \mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - \frac{2}{3} \mu (\nabla \cdot \mathbf{u}) \mathbf{I} \right\} \right] - \left( \kappa^{-1} \mu + \frac{Q_m}{\varepsilon_p^2} \right) \mathbf{u} + \mathbf{F} \quad (5-2)$$

## VOLUME FORCE

Enter the components of **Volume force**  $\mathbf{F}$  (SI unit: N/m<sup>3</sup>).

## Initial Values

---

The **Initial Values** node adds initial values for the velocity field and the pressure that can serve as an initial condition for a transient simulation or as an initial guess for a nonlinear solver.

## INITIAL VALUES

Enter initial values or expressions for the **Velocity field**  $\mathbf{u}$  (SI unit: m/s) and the **Pressure**  $p$  (SI unit: Pa). The default values are 0 m/s and 0 Pa, respectively.

## Fluid Properties

---

The **Fluid Properties** node adds the momentum and continuity equations to solve for free flow in nonporous domains. The node also provides an interface for defining the material properties of the fluid.

## MODEL INPUTS

Fluid properties, such as density and viscosity, can be defined through user inputs, variables, or by selecting a material. For the latter option, additional inputs — for example, temperature or pressure — may be required to define these properties.

### Temperature


By default, the single-phase flow interfaces are set to model isothermal flow. Hence, the **Temperature** is **User defined** and defaults to 293.15 K. If a Heat Transfer interface is included in the component, the temperature may alternatively be selected from this physics interface. All physics interfaces have their own tags (**Name**). For example, if a Heat Transfer in Fluids interface is included in the component, the **Temperature (ht)** option is available.

### *Absolute Pressure*

This input appears when a material requires the absolute pressure as a model input. The absolute pressure is used to evaluate material properties, but it also relates to the value of the calculated pressure field. There are generally two ways to calculate the pressure when describing fluid flow: either to solve for the absolute pressure or for a pressure (often denoted gauge pressure) that relates to the absolute pressure through a reference pressure.

The choice of pressure variable depends on the system of equations being solved. For example, in a unidirectional incompressible flow problem, the pressure drop over the modeled domain is probably many orders of magnitude smaller than the atmospheric pressure, which, when included, may reduce the stability and convergence properties of the solver. In other cases, such as when the pressure is part of an expression for the gas volume or the diffusion coefficients, it may be more convenient to solve for the absolute pressure.

The default **Absolute pressure**  $p_A$  is  $p + p_{\text{ref}}$  where  $p$  is the dependent pressure variable from the Navier–Stokes equations, and  $p_{\text{ref}}$  is from the user input defined at the physics interface level. When  $p_{\text{ref}}$  is nonzero, the physics interface solves for a gauge pressure. If the pressure field instead is an absolute pressure field,  $p_{\text{ref}}$  should be set to 0.

The **Absolute pressure** field can be edited by clicking **Make All Model Inputs Editable** (  ) and entering the desired value in the input field.

### **FLUID PROPERTIES**

If density variations with respect to pressure are to be included in the computations, the flow must be set to compressible.

The **Dynamic viscosity**  $\mu$  describes the relationship between the shear rate and the shear stresses in a fluid. Intuitively, water and air have low viscosities, and substances often described as thick (such as oil) have higher viscosities.

### *Gravity*

---

The **Gravity** global feature is automatically added when **Include gravity** is selected at the interface level in the **Physical Model** settings. It defines the gravity forces from the **Acceleration of gravity** value. When a turbulence model that solves for the turbulent kinetic energy,  $k$ , is used, the option **Include buoyancy-induced turbulence** is available. When selected, the **Buoyancy contribution** is by default set to **Automatic from multiphysics**. Contributions are only obtained from multiphysics couplings that

support buoyancy-induced turbulence, such as Nonisothermal Flow. If the **Buoyancy contribution** is switched to **User defined**, a text field for the **Turbulent Schmidt number** appears. See also [Theory for Buoyancy-Induced Turbulence](#) in the *CFD Module User's Guide*.



This feature requires a specific license. For a detailed overview of the functionality available in each product visit <https://www.comsol.com/products/specifications/>.

#### ACCELERATION OF GRAVITY

The **Acceleration of gravity** (SI unit m/s, default value  $-g_{\text{const}}\mathbf{e}_z$  in 2D axial symmetry and 3D and  $-g_{\text{const}}\mathbf{e}_y$  in 2D) is used to define the gravity forces. It should be a global quantity.



# The Free and Porous Media Flow, Brinkman Interface

The **Free and Porous Media Flow, Brinkman (fp)** interface (  ) is found under the **Porous Media and Subsurface Flow** branch (  ) when adding a physics interface. It is used to compute fluid velocity and pressure fields of single-phase flow where free flow is connected to porous media. The Free and Porous Media Flow, Brinkman interface is used over at least two different domains: a free channel and a porous medium. The physics interface is well suited for transitions between slow flow in porous media, governed by the Brinkman equations, and fast flow in channels described by the Navier–Stokes equations. Fluids with varying density can be included at Mach numbers below 0.3. Also the viscosity of a fluid can vary, for example, to describe non-Newtonian fluids. The physics interface can be used for stationary and time-dependent analyses.

When this physics interface is added, the following default nodes are also added in the **Model Builder** — **Fluid Properties**, **Wall**, and **Initial Values**. Then, from the **Physics** toolbar, add a **Porous Medium** node to be used on the domain selection corresponding to the porous media, or add other nodes that implement, for example, boundary conditions and volume forces. You can also right-click **Free and Porous Media Flow, Brinkman** 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 `fp`.

## PHYSICAL MODEL

### *Compressibility*

By default the physics interface uses the **Incompressible flow** formulation of the Navier–Stokes and Brinkman equations to model constant density flow. If required, select

**Weakly compressible flow** from the **Compressibility** list, to account for small variations in the density, which are only dependent on the temperature (nonisothermal flow) or **Compressible flow (Ma<0.3)** for fully compressible flow. However, for the flow modeled with this physics interface, the Mach number must be below 0.3.

#### *Swirl Flow*

For 2D axisymmetric components, select the **Swirl flow** check box to include the swirl velocity component, that is the velocity component  $u_\phi$  in the azimuthal direction. While  $u_\phi$  can be nonzero, there can be no gradients in the  $\phi$  direction. Note that this feature is only available for specific modules. Visit <https://www.comsol.com/products/specifications/> for a detailed overview.

#### *Neglect Inertial Term*

Select the **Neglect inertial term (Stokes flow)** check box if the inertial forces are small compared to the viscous forces.

#### *Enable porous media domains*

The **Enable porous media domains** check box is selected by default to solve the Brinkman equations in porous domains. If it is unchecked, the **Porous Medium** node is no longer available in this interface.

#### *Include Gravity*

Gravity is not included by default. Select the **Include gravity** check box to activate the acceleration of gravity. This automatically adds a global **Gravity** feature node to the interface model tree, and the buoyancy force is included in the Equations.

Also, when the **Include gravity** check box is selected, the **Use reduced pressure** option changes the pressure formulation from using the total pressure (default) to using the reduced pressure. This option is suitable for configurations where the density changes are very small; otherwise, the default formulation can be used. For more information, see [Gravity](#).

#### *Porous Treatment of No Slip Condition*

Choose how the **No Slip** condition on internal boundaries should be treated. The options are **Standard no slip formulation** and **Porous slip** (default). The latter option provides a unified treatment when the porous matrix is fully resolved as well as when it is under resolved ensuring a smooth transition between regions with different resolutions; see No Slip under [Wall](#) in the [The Creeping Flow and Laminar Flow Interfaces](#) section.

#### *Reference Pressure Level*

Enter a **Reference pressure level**  $p_{\text{ref}}$  (SI unit: Pa). The default value is 1 [atm].

Reference Temperature

Enter a **Reference temperature**  $T_{\text{ref}}$  (SI unit: K). The default value is 293.15 [K].

Reference Position

If **Include gravity** is selected, a **Reference position**  $r_{\text{ref}}$  (SI unit: m) can be specified which is then used for the calculation of the hydrostatic pressure.

TURBULENCE

Turbulent flow can be simulated by changing the **Turbulence model type** to **RANS** (Reynolds-Averaged Navier–Stokes) or **Large Eddy Simulation** (which is only available in 3D). If turbulent flow is activated, you can choose from different **Turbulence models** and options for **Wall treatment**. For a description of the different turbulence models, wall treatment options, and turbulence model parameters see [Theory for the Turbulent Flow Interfaces](#) in the *CFD Module User’s Guide*.

DEPENDENT VARIABLES

The following dependent variables (fields) are defined for this physics interface — the **Velocity field**  $\mathbf{u}$  (SI unit: m/s) and its components, and the **Pressure**  $p$  (SI unit: Pa).



- [Domain, Boundary, Point, and Pair Nodes for the Free and Porous Media Flow, Brinkman Interface](#)
- [Theory for the Free and Porous Media Flow, Brinkman Interface](#)

*Domain, Boundary, Point, and Pair Nodes for the Free and Porous Media Flow, Brinkman Interface*

[The Free and Porous Media Flow, Brinkman Interface](#) has the following domain, boundary, point, and pair nodes, listed in alphabetical order, 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.

- Fluid Properties
- Gravity
- Initial Values
- Porous Medium
- Mass Source
- Volume Force
- Wall

Note that some features are only available with certain COMSOL products (see <https://www.comsol.com/products/specifications/>).

The following nodes (listed in alphabetical order) are described for the Laminar Flow interface:

- No Viscous Stress
- Flow Continuity
- Inlet
- Line Mass Source
- Outlet
- Open Boundary
- Periodic Flow Condition
- Point Mass Source
- Pressure Point Constraint
- Symmetry



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.

## Fluid Properties

Use the **Fluid Properties** node to define the fluid material, density, and dynamic viscosity.

### FLUID PROPERTIES

The default **Fluid material** uses the **Domain material** (the material defined for the domain). Select another material as needed.

The default **Density**  $\rho$  (SI unit:  $\text{kg}/\text{m}^3$ ) uses values **From material** based on the **Fluid material** selection. For **User defined** enter another value or expression. The default is  $0 \text{ kg}/\text{m}^3$ .

The **Dynamic viscosity**  $\mu$  (SI unit: Pa·s) uses values **From material** based on the **Fluid material** selection. For **User defined** enter another value or expression. The default is 0 Pa·s.

## *Porous Medium*

---

The **Porous Medium** node adds the Brinkman equations (excluding any mass sources) to the interface. It has two default subfeatures — the **Fluid** subfeature where the fluid density  $\rho$  (SI unit: kg/m<sup>3</sup>) and dynamic viscosity  $\mu$  (SI unit: Pa·s) are defined and the **Porous Matrix** subfeature where the permeability  $\kappa$  (SI unit: m<sup>2</sup>) and porosity  $\varepsilon_p$  (dimensionless) are specified.

### **FLOW MODEL**

Define if the flow follows a linear or nonlinear pressure-velocity relationship.

- **Darcian flow** (default) defines a linear relationship characteristic for low Reynolds number laminar flows.
- **Non-Darcian flow** defines a nonlinear relationship due to inertial or turbulence effects.

### **POROUS MEDIUM TURBULENCE MODEL**

Defines a model which complements the basic turbulence model and accounts for the influence of the porous matrix on turbulent flow in Porous Media.

- **Default** and **Nakayama-Kuwahara** specify **Coefficient of subgrid turbulence generation by porous matrix**  $C_{\text{stpm}}$  which by default is set to 0.212.
- **Pedras-de Lemos** asks to specify **Choice of the Pedras-de Lemos model coefficient**. When **User defined** is chosen, specify **Coefficient of subgrid turbulence generation by porous matrix**  $C_{\text{stpm}}$  with default value 0.212. Choosing **Original** results in  $C_{\text{stpm}} = 0.28 \cdot \sqrt{\varepsilon_p}$ , while choosing **Recalibrated** results in  $C_{\text{stpm}} = 0.18$ .

## *Fluid*

---

This node defines the material properties of the fluid of the **Porous Medium** parent node.

## MODEL INPUT

This section contains fields and values that are inputs for expressions defining material properties. If such user-defined property groups are added, the model inputs appear here.

## FLUID PROPERTIES

Specify the **Density**  $\rho$  and the **Dynamic viscosity**  $\mu$  (SI unit: Pa·s) of the fluid. The defaults use values **From material**, for **User defined** enter a value or expression.

The dynamic viscosity describes the relationship between the shear stresses and the shear rate in a fluid. Intuitively, water and air have a low viscosity, and substances often described as thick, such as oil, have a higher viscosity.

## *Porous Matrix*

---

This node defines the porosity and material properties of the solid matrix of the **Porous Medium** parent node.

## MODEL INPUT

This section contains fields and values that are inputs for expressions defining material properties. If such user-defined property groups are added, the model inputs appear here.

## MATRIX PROPERTIES

The **Porosity**  $\varepsilon_p$  (a dimensionless number between 0 and 1) uses by default the value **From material**. For **User defined** the default value is 1 which is equivalent to free flow.

Select a **Permeability model** to specify the capacity of the porous material to transmit flow. Depending on the **Flow model** selection in the parent **Porous Medium** feature node you can choose between the following options:

- **Permeability** (the default) to directly enter the permeability of the porous matrix or **Kozeny-Carman** to define the permeability from the porosity and mean **Particle diameter** for **Darcian** flow or
- **Forchheimer** or **Ergun** for **Non-Darcian** flow.

Note that some of these options are only available for certain modules. For a detailed overview of the functionality available in each product, visit <https://www.comsol.com/products/specifications/>.

The default **Permeability**  $\kappa$  (SI unit:  $\text{m}^2$ ) uses the value **From material**, as defined by the **Porous material** list. For **User defined** select **Isotropic** to define a scalar value or **Diagonal** or **Symmetric** to define a tensor value.

For **Kozeny-Carman**, enter the mean **Particle diameter**  $d_p$  (SI unit: m), the default value is 0.5 mm.

In case a **Non-Darcian** flow model is chosen in the parent node, **Forchheimer** is the default selection for the **Permeability model**. For **Forchheimer**, the default **Permeability**  $\kappa$  (SI unit:  $\text{m}^2$ ) uses the value **From material**. Furthermore, the dimensionless **Forchheimer parameter**  $c_F$  can be defined. The default value is 0.55.

If **Ergun** is selected, enter the mean **Particle diameter**  $d_p$  (default value: 0.5 mm). The permeability  $\kappa$  is then calculated using [Equation 2-4](#).



- [About Darcian and Non-Darcian Flow](#) in the *Porous Media Flow Module User's Guide*
- [Permeability Models](#) in the *Porous Media Flow Module User's Guide*

### Mass Source

Enter a value or expression for an optional mass source (or sink) **Source term**  $Q_m$  (SI unit:  $\text{kg}/(\text{m}^3 \cdot \text{s})$ ). This term accounts for mass deposit or mass creation within porous domains. The physics interface assumes that the mass exchange occurs at zero velocity.

### Volume Force

The **Volume Force** node specifies the force  $\mathbf{F}$  on the right-hand side of the Navier–Stokes or Brinkman equations, depending on whether a [Fluid Properties](#) or [Porous Medium](#) node is active on the domain. It then acts on each fluid element in the specified domains.

#### VOLUME FORCE

Enter the components of the **Volume force**  $\mathbf{F}$  (SI unit:  $\text{N}/\text{m}^3$ ).

## *Initial Values*

---

The **Initial Values** node adds initial values for the velocity field and the pressure that can serve as an initial condition for a transient simulation or as an initial guess for a nonlinear solver.

### **INITIAL VALUES**

Enter initial values or expressions for the **Velocity field  $\mathbf{u}$**  (SI unit: m/s) and for the **Pressure  $p$**  (SI unit: Pa). The default values are 0 m/s and 0 Pa, respectively.

## *Gravity*

---

The **Gravity** global feature is automatically added when **Include gravity** is selected at the interface level in the **Physical Model** settings. It defines the gravity forces from the **Acceleration of gravity** value. When a turbulence model that solves for the turbulent kinetic energy,  $k$ , is used, the option **Include buoyancy-induced turbulence** is available. When selected, the **Buoyancy contribution** is by default set to **Automatic from multiphysics**. Contributions are only obtained from multiphysics couplings that support buoyancy-induced turbulence, such as Nonisothermal Flow. If the **Buoyancy contribution** is switched to **User defined**, a text field for the **Turbulent Schmidt number** appears. See also [Theory for Buoyancy-Induced Turbulence](#) in the *CFD Module User's Guide*.

This feature requires a specific license. For a detailed overview of the functionality available in each product visit <https://www.comsol.com/products/specifications/>.

### **ACCELERATION OF GRAVITY**

The **Acceleration of gravity** (SI unit m/s, default value  $-g_{\text{const}}\mathbf{e}_z$  in 2D axial symmetry and 3D and  $-g_{\text{const}}\mathbf{e}_y$  in 2D) is used to define the gravity forces. It should be a global quantity.

## *Wall*

---

The **Wall** node includes a set of boundary conditions describing fluid-flow conditions at stationary, moving, and leaking walls.



**BOUNDARY CONDITION**

Select a **Boundary condition** for the wall.

- [No Slip](#)<sup>1</sup>
- [Leaking Wall](#)
- [Slip](#)

*No Slip*

**No slip** is the default boundary condition for a stationary solid wall for laminar flow (and SST, Low Re k-ε, Algebraic yPlus, L-VEL, and Spalart–Allmaras turbulence models). The condition prescribes  $\mathbf{u} = 0$ ; that is, the fluid at the wall is not moving.


*Slip*

The **Slip** option prescribes a no-penetration condition,  $\mathbf{u} \cdot \mathbf{n} = 0$ . It is implicitly assumed that there are no viscous effects at the slip wall and hence, no boundary layer develops. From a modeling point of view, this can be a reasonable approximation if the main effect of the wall is to prevent fluid from leaving the domain.

*Leaking Wall*

This boundary condition may be used to simulate a wall where fluid is leaking into or leaving the domain with the velocity  $\mathbf{u} = \mathbf{u}_1$  through a perforated wall. The components of the **Fluid velocity**  $\mathbf{u}_1$  on the leaking wall should be specified.

**CONSTRAINT SETTINGS**

This section is displayed by clicking the **Show** button (  ) and selecting **Advanced Physics Options**.



- [Theory for the Wall Boundary ConditionThe Moving Mesh Interface in the COMSOL Multiphysics Reference Manual](#)

# Theory for the Brinkman Equations Interface

The [Brinkman Equations Interface](#) theory is described in this section:

- [About the Brinkman Equations](#)
- [Brinkman Equations Theory](#)
- [References for the Brinkman Equations Interface](#)

## *About the Brinkman Equations*

---

The *Brinkman equations* describe fluids in porous media for which the momentum transport within the fluid due to shear stresses is of importance. This mathematical model extends Darcy's law to include a term that accounts for the viscous transport in the momentum balance, and it treats both the pressure and the flow velocity vector as independent variables. Use the Free and Porous Media Flow, Brinkman interface to model combinations of porous media and free flow domains. These types of problems are often encountered in applications such as monolithic reactors and fuel cells

In porous domains, the flow variables and fluid properties are defined at any point inside the medium by means of averaging of the actual variables and properties over a certain volume surrounding the point. This control volume must be small compared to the typical macroscopic dimensions of the problem, but it must be large enough to contain many pores and solid matrix elements.

Porosity is defined as the fraction of the control volume that is occupied by pores. Thus, the porosity can vary from zero for pure solid regions to unity for domains of free flow.

The physical properties of the fluid, such as density and viscosity, are defined as *intrinsic volume averages* that correspond to a unit volume of the pores. Defined this way, they present the relevant physical parameters that can be measured experimentally, and they are assumed to be continuous with the corresponding parameters in the adjacent free flow.

The flow velocity is defined as a *superficial volume average*, and it corresponds to a unit volume of the medium including both the pores and the matrix. It is sometimes called the Darcy velocity, defined as the volume flow rate per unit cross section of the

medium. Such a definition makes the velocity field continuous across the boundaries between porous regions and regions of free flow.

### *Brinkman Equations Theory*

---

The dependent variables in the Brinkman equations are the Darcy velocity and the pressure. The flow in porous media is governed by a combination of the continuity equation and the momentum equation, which together form the Brinkman equations:

$$\frac{\partial}{\partial t}(\varepsilon_p \rho) + \nabla \cdot (\rho \mathbf{u}) = Q_m \quad (5-3)$$

$$\begin{aligned} \frac{\rho}{\varepsilon_p} \left( \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \frac{\mathbf{u}}{\varepsilon_p} \right) = \\ -\nabla p + \nabla \cdot \left[ \frac{1}{\varepsilon_p} \left\{ \mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - \frac{2}{3} \mu (\nabla \cdot \mathbf{u}) \mathbf{I} \right\} \right] - \left( \kappa^{-1} \mu + \frac{Q_m}{\varepsilon_p^2} \right) \mathbf{u} + \mathbf{F} \end{aligned} \quad (5-4)$$

In these equations:

- $\mu$  (SI unit: kg/(m·s)) is the dynamic viscosity of the fluid
- $\mathbf{u}$  (SI unit: m/s) is the velocity vector
- $\rho$  (SI unit: kg/m<sup>3</sup>) is the density of the fluid
- $p$  (SI unit: Pa) is the pressure
- $\varepsilon_p$  is the porosity
- $\kappa$  (SI unit: m<sup>2</sup>) is the permeability of the porous medium, and
- $Q_m$  (SI unit: kg/(m<sup>3</sup>·s)) is a mass source or sink

Influence of gravity and other volume forces can be accounted for via the force term  $\mathbf{F}$  (SI unit: kg/(m<sup>2</sup>·s<sup>2</sup>)).

When the **Neglect inertial term (Stokes flow)** check box is selected, the term  $(\mathbf{u} \cdot \nabla)(\mathbf{u}/\varepsilon_p)$  on the left-hand side of [Equation 5-2](#) is disabled.

The mass source,  $Q_m$ , accounts for mass deposit and mass creation within the domains. The mass exchange is assumed to occur at zero velocity.

The Forchheimer and Ergun drag options add a viscous force proportional to the square of the fluid velocity,  $\mathbf{F} = -\rho\beta|\mathbf{u}|\mathbf{u}$ , on the right-hand side of [Equation 5-2](#); see [References for the Darcy's Law Interface](#) in the *CFD Module User Guide*.

In case of a flow with variable density, Equation 5-1 and Equation 5-2 must be solved together with the equation of state that relates the density to the temperature and pressure (for instance the ideal gas law).

For incompressible flow, the density stays constant in any fluid particle, which can be expressed as

$$\frac{\partial}{\partial t}(\varepsilon_p \rho) + \mathbf{u} \cdot \nabla \rho = 0$$

and the continuity equation (Equation 5-1) reduces to

$$\rho \nabla \cdot \mathbf{u} = Q_m$$

### *References for the Brinkman Equations Interface*

---

1. D. Nield and A. Bejan, *Convection in Porous Media*, 3rd ed., Springer, 2006.
2. M. Le Bars and M.G. Worster, “Interfacial Conditions Between a Pure Fluid and a Porous Medium: Implications for Binary Alloy Solidification,” *J. Fluid Mech.*, vol. 550, pp. 149–173, 2006.

# Theory for the Free and Porous Media Flow, Brinkman Interface

The [Free and Porous Media Flow, Brinkman Interface](#) uses the Navier–Stokes equations to describe the flow in open regions, and the Brinkman equations to describe the flow in porous regions.

The same fields,  $\mathbf{u}$  and  $p$ , are solved for in both the free flow domains and in the porous domains. This means that the pressure in the free fluid and the pressure in the pores is continuous over the interface between a free flow domain and a porous domain. It also means that continuity is enforced between the fluid velocity in the free flow and the Darcy velocity in the porous domain. This treatment is one of several possible models for the physics at the interface. Examples of other models can be found in [Ref. 1](#).

The continuity in  $\mathbf{u}$  and  $p$  implies a stress discontinuity at the interface between a free-flow domain and a porous domain. The difference corresponds to the stress absorbed by the rigid porous matrix, which is a consequence implicit in the formulations of the Navier–Stokes and Brinkman equations.

## *Reference for the Free and Porous Media Flow, Brinkman Interface*

---

1. M.L. Bars and M.G. Worster, “Interfacial Conditions Between a Pure Fluid and a Porous Medium: Implications for Binary Alloy Solidification,” *J. Fluid Mech.*, vol. 550, pp. 149–173, 2006.



## Mathematics, Moving Interfaces

The Level Set and Phase Field in Fluids Moving Interfaces are available under the **Mathematics>Moving Interface** branch (  ) when adding interfaces.

- [The Level Set Interface](#)
- [The Phase Field in Fluids Interface](#)
- [The Ternary Phase Field Interface](#)
- [Theory for the Level Set Interface](#)
- [Theory for the Phase Field in Fluids Interface](#)
- [Theory for the Ternary Phase Field Interface](#)



# The Level Set Interface

In this section:

- [The Level Set Interface](#)
- [The Level Set in Porous Media Interface](#)
- [Domain, Boundary, and Pair Nodes for the Level Set Interface](#)

## *The Level Set Interface*

---

The **Level Set (Is)** interface (  ), found under the **Mathematics>Moving Interface** branch (  ) when adding an interface, is used to track moving interfaces in fluid-flow models by solving a transport equation for the level set function. Simulations using the **Level Set** interface are always time dependent because the position of an interface almost always depends on its history.

The main node is the [Level Set Model](#) feature, which adds the level set equation and provides an interface for defining the level set properties and the velocity field.

When this physics interface is added, the following default nodes are also added in the **Model Builder — Level Set Model, No Flow** (the default boundary condition), **Initial Values**, and **Initial Values, Fluid 2**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions. You can also right-click **Level Set** 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 `ls`.


### STABILIZATION

To display this sections, click the **Show More Options** button (  ) and select **Stabilization** in the **Show More Options** dialog box.



There are two stabilization methods available — **Streamline diffusion** and **Crosswind diffusion**. Both are active by default and should remain selected for optimal performance.

**ADVANCED SETTINGS**

To display this sections, click the **Show More Options** button (  ) and select **Advanced Physics Options** in the **Show More Options** dialog box.


Specify the **Convective term** as **Nonconservative form** (default) or **Conservative form**.

**DISCRETIZATION**



By default, the Level Set interface uses **Linear** elements.

**DEPENDENT VARIABLES**

The dependent variable (field variable) is the **Level set variable** phi. The name can be changed but the names of fields and dependent variables must be unique within a model.

	<ul style="list-style-type: none"><li>• <a href="#">Conservative and Nonconservative Form</a></li><li>• <a href="#">Domain, Boundary, and Pair Nodes for the Level Set Interface</a></li><li>• <a href="#">Theory for the Level Set Interface</a></li></ul>
---	---

*The Level Set in Porous Media Interface*

The **Level Set in Porous Media (ls)** interface (  ), found under the **Mathematics>Moving Interface** branch (  ) when adding an interface, is used to track moving interfaces in fluid-flow through porous media models by solving a transport equation for the level set function. Simulations using the **Level Set in Porous Media** interface are always time dependent since the position of an interface almost always depends on its history.

The main node is the Porous Medium [Level Set Model](#) feature, which adds the level set equation and provides interfaces for defining the level set properties, the velocity field and the porosity of the porous matrix.

When this physics interface is added, the following default nodes are also added in the **Model Builder** — **Porous Medium, No Flow** (the default boundary condition), **Initial Values**, and **Initial Values, Fluid 2**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions. You can also right-click **Level Set in Porous Media** to select physics features from the context menu.

## SETTINGS

The rest of the settings are the same as for the **Level Set** interface.

### *Domain, Boundary, and Pair Nodes for the Level Set Interface*

has the following domain, boundary, and pair nodes described.

- [Continuity](#)
- [Initial Values](#)
- [Inlet](#)
- [Level Set Model](#)
- [No Flow](#)
- [Open Boundary](#)
- [Outlet](#)<sup>1</sup>
- [Periodic Condition](#) (see [Periodic Boundary Conditions](#) in the *COMSOL Multiphysics Reference Manual*)
- [Porous Matrix](#)
- [Porous Medium](#)
- [Symmetry](#)<sup>1</sup>
- [Thin Barrier](#)

<sup>1</sup> Described for the Laminar Flow interface.



Boundary conditions for axial symmetry boundaries are not required. For the symmetry axis at  $r = 0$ , the software automatically provides a suitable boundary condition and adds an **Axial Symmetry** node that is valid on the axial symmetry boundaries only.



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.

### *Level Set Model*

If the **Level Set Model** node is a domain feature under the [The Level Set Interface](#), it adds the following transport equation governing a level set function  $\phi$ :

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = \gamma \nabla \cdot \left( \varepsilon_{ls} \nabla \phi - \phi(1 - \phi) \frac{\nabla \phi}{|\nabla \phi|} \right)$$

If it is a domain subfeature under the [Porous Medium](#) domain feature, it adds the following equation

$$\varepsilon_p \frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = \gamma \nabla \cdot \left( \varepsilon_{ls} \nabla \phi - \phi(1 - \phi) \frac{\nabla \phi}{|\nabla \phi|} \right)$$

In both cases the **Level Set Model** node provides the options to define the associated level set parameters and the velocity field.

#### LEVEL SET PARAMETERS

Enter a value or expression for the **Reinitialization parameter**  $\gamma$  (SI unit: m/s). The default is 1 m/s.

Enter a value or expression for the **Parameter controlling interface thickness**  $\varepsilon_{ls}$  (SI unit: m). The default expression is `1s.ep_default`, a value proportional to the maximum element size with a proportionality factor dependent on the ratio of the maximum and minimum element sizes in the domain.

#### CONVECTION

Enter values or expressions for the components ( $u$ ,  $v$ , and  $w$  in 3D, for example) of the **Velocity field**  $\mathbf{u}$  (SI unit: m/s). The applied velocity field transports the level set function through convection.

### *Porous Medium*

---

The **Porous Medium** node adds the level set equation. It has two default subfeatures — the [Level Set Model](#) feature where the reinitialization parameter  $\gamma$  (SI unit: m/s) and the parameter controlling the interface thickness  $\varepsilon_{ls}$  (SI unit: m) are defined, and the [Porous Matrix](#) feature where the porosity  $\varepsilon_p$  (dimensionless) is specified.

### *Porous Matrix*

---

This node contains inputs for the material properties of the solid matrix of the [Porous Medium](#) parent node.

#### MATRIX PROPERTIES

The **Porosity**  $\varepsilon_p$  (a dimensionless number between 0 and 1) uses by default the value **From material**. For **User defined** the default is 1.

## Initial Values

---

Use the **Initial Values** node to define the initial values of the level set variable.


### INITIAL VALUES

The initial positions of the two fluids, and consequently the separating interface, can be specified in two ways. The level set function can be specified explicitly, using any type of variable or expression. Alternatively the fluid occupying the current domain selection can be specified. Using the former method, a Time Dependent study step can directly be solved. Using the latter method, a Phase Initialization study step is needed in order to initialize the level set function across the fluid-fluid interface.

Specify the **Level set variable** as one of the following:

- To specify the fluid occupying the domain, select **Specify phase** (default). Then select the check box **Fluid 1** ( $\phi = 0$ ) or **Fluid 2** ( $\phi = 1$ ).
- For **User defined** enter a value or expression for the level set variable  $\phi$ .



If the **Phase Initialization** () study step is being used, for the initialization to work it is crucial that there are two **Initial Values** nodes. One of the Initial Values nodes should use **Specify phase: Fluid 1** ( $\phi = 0$ ) and the other **Specify phase: Fluid 2** ( $\phi = 1$ ). The initial fluid-fluid interface is then automatically placed on all interior boundaries between both phases. The smooth initialization is not used in domains with **User defined** initial values.

See .

## Inlet

---

### LEVEL SET CONDITION

Specify **Level Set Condition** according to one of the following options:

- **Fluid 1** ( $\phi = 0$ )
- **Fluid 2** ( $\phi = 1$ )
- **Specify level set function explicitly**

When the **Specify level set function explicitly** is selected, the level set function  $\phi$  must be specified explicitly. The value must be in the range from 0 to 1, where the default is 0.

## *No Flow*

---

The **No Flow** node adds a boundary condition that represents boundaries where there is no flow across the boundary. This is the default boundary condition.

## *Open Boundary*

---

Use this node to set up transport across boundaries where both convective inflow and outflow can occur. On the parts of the boundary where fluid flows into the domain, an exterior value of the level set function is prescribed. On the remaining parts, where fluid flows out of the domain, a condition equivalent to the **Outflow** node is instead prescribed.

The direction of the flow across the boundary is typically calculated by a fluid flow interface and is provided as a model input to the Level Set interface.

### **LEVEL SET CONDITION**

Specify **Level Set Condition** according to one of the following options:

- **Fluid 1** ( $\phi = 0$ )
- **Fluid 2** ( $\phi = 1$ )
- **Specify level set function explicitly**

When the **Specify level set function explicitly** is selected, the level set function  $\phi$  must be specified explicitly. The value must be in the range from 0 to 1, where the default is 0.

## *Thin Barrier*

---

The **Thin Barrier** feature is available on interior boundaries and introduces a discontinuity in the level set variable across the boundary. A **No Flow** condition is prescribed on each side. This node can be used to avoid meshing thin structures.

## *Continuity*

---

The **Continuity** condition is suitable for pairs where the boundaries match; it prescribes that the level set variable is continuous across the pair.

The parts of the boundary where a source boundary lacks a corresponding destination boundary and vice versa will get conditions from other boundary conditions under the physics interface acting as fallback boundary conditions on the nonoverlapping parts.

By default, the **No Flow** node is used as fallback unless another feature has been selected for the boundary.



About Identity and Contact Pairs

---

#### LOCATION IN USER INTERFACE



Right-click the physics interface and select **Pairs>Continuity**.

*Ribbon*

Physics tab with the physics interface selected:

**Pairs>Continuity**

# The Phase Field in Fluids Interface

The **Phase Field in Fluids (pf)** interface (  ), found under the **Mathematics>Moving Interface** branch (  ) when adding a physics interface, is used to track moving fluid-fluid interfaces by solving two transport equations, one for the phase field variable,  $\phi$ , and one for the mixing energy density,  $\psi$ . The position of the interface is determined by minimizing the free energy.

The main node is the **Phase Field Model** feature, which adds the phase field equations and provides an interface for defining the phase field model properties.

When this physics interface is added, the following default nodes are also added in the **Model Builder** — **Phase Field Model**, **Wetted Wall** (the default boundary condition), **Initial Values**, and **Initial Values, Fluid 2**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions. You can also right-click **Phase Field in Fluids** 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 `pf`.

## ADVANCED SETTINGS

To display this sections, click the **Show More Options** button (  ) and select **Advanced settings options** in the **Show More Options** dialog box.

Specify the **Convective term** as **Nonconservative form** (default) or **Conservative form**.

## DISCRETIZATION

By default, the Phase Field in Fluids interface uses **Linear** elements.

## DEPENDENT VARIABLES

This interface defines the dependent variables (fields) **Phase field variable**  $\phi$  and **Phase field help variable**  $\psi$ . If required, edit the name, but dependent variables must be unique within a model.



- [Conservative and Nonconservative Form](#)
- [Domain, Boundary, and Pair Nodes for the Phase Field in Fluids Interface](#)
- [Theory for the Phase Field in Fluids Interface](#)

### *Domain, Boundary, and Pair Nodes for the Phase Field in Fluids Interface*

includes the following domain, boundary, and pair nodes, listed in alphabetical order, 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.

- [Continuity](#)
- [Initial Values](#)
- [Inlet](#)
- [Interior Wetted Wall](#)
- [Outlet](#)
- [Periodic Condition](#) (see [Periodic Boundary Conditions](#) in the *COMSOL Multiphysics Reference Manual*)
- [Phase Field Model](#)
- [Symmetry](#)
- [Wetted Wall](#)



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 **Phase Field Model** node adds the equations described in [The Equations for the Phase Field Method](#). The node defines the associated phase field parameters including surface tension and interface thickness.

#### PHASE FIELD PARAMETERS

Define the following phase field parameters. Enter a value or expression for the:

- **Surface tension coefficient**  $\sigma$  (SI unit: N/m).

**Parameter controlling interface thickness**  $e_{pf}$  (SI unit: m). The default expression is `pf.ep_default`, a value proportional to the maximum element size with a proportionality factor dependent on the ratio of the maximum and minimum element sizes in the domain.

- **Mobility tuning parameter**  $\chi$  (SI unit: m·s/kg). This parameter determines the time scale of the Cahn–Hilliard diffusion and it thereby also governs the diffusion-related time scale for the interface. Select **User defined** (default) or **Calculate from velocity**. For **User defined** enter a value or expression for the mobility tuning parameter  $\chi$ . The default is 1 m·s/kg, which is a good starting point for most models.

For **Calculate from velocity** enter a value for the expected velocity at the interface  $U$  (SI unit: m/s).  $\chi$  is then computed as:

$$\chi = \frac{U h_{\max}}{3\sqrt{2}\sigma\epsilon}$$

where  $h_{\max}$  is the largest mesh cell size.



Keep the  $\chi$  parameter value large enough to maintain a constant interface thickness but still low enough to not damp the convective motion. A too high mobility can also lead to excessive diffusion of droplets.

#### EXTERNAL FREE ENERGY

Add a source of external free energy to the phase field equations. This modifies the last term on the right-hand side of the equation:

$$\psi = -\nabla \cdot \epsilon^2 \nabla \phi + (\phi^2 - 1)\phi + \left(\frac{\epsilon^2}{\lambda}\right) \frac{\partial f}{\partial \phi}$$

The external free energy  $f$  (SI unit:  $\text{J}/\text{m}^3$ ) is a user-defined free energy. In most cases, the external free energy can be set to zero. Manually differentiate the expression for the external free energy with respect to  $\phi$  and then enter it into the  **$\phi$ -derivative of external free energy** field  $\partial f/\partial \phi$ .

## CONVECTION

Enter values or expressions for the components ( $u$ ,  $v$ , and  $w$  in 3D, for example) of the **Velocity field  $u$**  (SI unit:  $\text{m}/\text{s}$ ). The applied velocity field transports the phase field variables through convection.

## Initial Values

---

Use the **Initial Values** node to define the initial values of the phase field variable.


## INITIAL VALUES

The initial positions of the two fluids, and consequently the fluid-fluid interface, can be specified in two ways. The phase field variable can be specified explicitly, using any type of variable or expression. Alternatively the fluid occupying the current domain selection can be specified. Using the former method, a Time Dependent study step can directly be solved. Using the latter method, a Phase Initialization study step is needed in order to initialize the phase field function across the fluid-fluid interface.

Specify the **Phase field variable** as one of the following:

- To specify the fluid occupying the domain, select **Specify phase** (default). Then select the check box **Fluid 1** ( $\phi = -1$ ) or **Fluid 2** ( $\phi = 1$ ).
- For **User defined** enter a value or expression for the phase field variable  $\phi$ .



If the **Phase Initialization** () study step is being used, for the initialization to work it is crucial that there are two **Initial Values** nodes. One of the Initial Values nodes is set to **Specify phase: Fluid 1** ( $\phi = -1$ ) and the other to **Specify phase: Fluid 2** ( $\phi = 1$ ). The initial fluid-fluid interface is then automatically placed on all interior boundaries between both phases. The smooth initialization is not used in domains with **User defined** initial values.

**SETTINGS**

Specify **Phase field Condition** according to one of the following options:

- **Fluid 1** ( $\phi = -1$ )
- **Fluid 2** ( $\phi = 1$ )
- **Specify phase field function explicitly**

When **Specify phase field function explicitly** is selected, the phase field function  $\phi$  must be specified in the entire domain. The value must be in the range from  $-1$  to  $1$ , where the default is  $1$ .

*Wetted Wall*

---

The **Wetted Wall** node is the default boundary condition representing wetted walls. Along a wetted wall the contact angle for the fluid,  $\theta_w$ , is specified, and across it, the mass flow is zero. This is prescribed by

$$\mathbf{n} \cdot \varepsilon^2 \nabla \phi = \varepsilon^2 \cos(\theta_w) |\nabla \phi|$$

in combination with

$$\mathbf{n} \cdot \frac{\gamma_\lambda^2}{\varepsilon} \nabla \psi = 0$$

The contact angle  $\theta_w$  can be defined directly or from Young's equation, which considers the components of the forces in the plane of the surface:

$$\sigma \cos(\theta_w) + \gamma_{s2} = \gamma_{s1} \quad (6-1)$$

where  $\gamma_{s1}$  is the surface energy density on the fluid 1 — solid (wall) interface and  $\gamma_{s2}$  is the surface energy density on the fluid 2 — solid (wall) interface.

**WETTED WALL**

Select an option from the **Specify contact angle list** — **Directly** (the default) or **Through Young's equation**. For **Directly** enter a value or expression for the **Contact angle**  $\theta_w$  (SI unit: rad). The default value is  $\pi/2$  radians. For **Through Young's equation** enter values or expressions for **Phase 1-Solid surface energy density**  $\gamma_{s1}$  (SI unit: J/m<sup>2</sup>) and **Phase 2-Solid surface energy density**  $\gamma_{s2}$  (SI unit: J/m<sup>2</sup>).

### *Interior Wetted Wall*

---

The **Interior Wetted Wall** feature is available on interior boundaries. It is similar to the [Wetted Wall](#) condition except that it applies on both sides of an interior boundary and introduces a discontinuity of the phase field variables across the boundary. This node can be used to avoid meshing thin structures.

Along an interior wetted wall the contact angle for the fluid,  $\theta_w$ , is specified on both sides of the boundary, and across it, the mass flow is zero.

#### **WETTED WALL**

Enter a value or expression for the **Contact angle**  $\theta_w$ . The default value is  $\pi/2$  rad.

### *Continuity*

---

The **Continuity** condition is suitable for pairs where the boundaries match; it prescribes that the phase field variables are continuous across the pair.

The parts of the boundary where a source boundary lacks a corresponding destination boundary and vice versa will get conditions from other boundary conditions under the physics interface acting as fallback boundary conditions on the nonoverlapping parts. By default, the [Wetted Wall](#) node is used as fallback unless another feature has been selected for the boundary.



About Identity and Contact Pairs

---

#### **LOCATION IN USER INTERFACE**



Right-click the physics interface and select **Pairs>Continuity**.

#### *Ribbon*

Physics tab with the physics interface selected:

**Pairs>Continuity**

# The Ternary Phase Field Interface

The **Ternary Phase Field (terpf)** interface (  ), found under the **Mathematics>Moving Interface** branch (  ) when adding a physics interface, is used to track moving interfaces separating three different phases. The interface solves four transport equations: two equations governing phase field variables,  $\phi_A$  and  $\phi_B$ , and two equations for the corresponding generalized chemical potentials,  $\eta_A$  and  $\eta_B$ . The position of the interfaces are determined by minimization of the free energy.

The phase field variables are defined such that they represent the volume fraction of the respective phase. The volume fraction of the third phase not solved for, but computed from the natural constraint that the sum of the volume fractions of all phases is one.

The main node of the interface is the [Phase Field Model](#) feature, which adds the phase field equations and provides an interface for defining the phase field model properties such as surface tensions and parameters controlling the interface thickness.

When this physics interface is added, the following default nodes are also added in the **Model Builder** — **Mixture Properties**, **Wetted Wall** and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions. You can also right-click **Ternary Phase Field** 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 `terpf`.

## DEPENDENT VARIABLES

This interface defines the dependent variables (fields):

- **Phase field variable, fluid A** `phiA`
- **Phase field variable, fluid B** `phiB`

- **Generalized chemical potential of fluid A**  $\eta_A$
- **Generalized chemical potential of fluid B**  $\eta_B$

If required, edit the any of the names, but dependent variables must be unique within a model.



## Theory for the Ternary Phase Field Interface

---

### *Domain, Boundary, and Pair Nodes for the Ternary Phase Field Interface*

---

The **Ternary Phase Field Interface** includes the following domain, boundary, and pair nodes, listed in alphabetical order, available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or by right-clicking the interface node to access the context menu (all users).

- [Initial Values](#)
- [Inlet](#)
- [Outlet](#)
- [Phase Field Model](#)
- [Symmetry](#)
- [Wetted Wall](#)

### *Mixture Properties*

---

The **Mixture Properties** node adds the equations described in . The node defines the associated phase field parameters, including the surface tension coefficients and parameters controlling the thickness of the interfaces.

#### **MODEL INPUTS**

The surface tension coefficients can be defined through user inputs, variables, or by selecting a material. For the latter option, an additional input, the temperature, may be required to define these properties.

#### *Temperature*

By default, the **Temperature** model input is set to **Common model input**, and the temperature is controlled from [Default Model Inputs](#) under **Global Definitions** or by a locally defined [Model Input](#). If a Heat Transfer interface is included in the component, it controls the temperature **Common model input**. Alternatively, the temperature field

can be selected from another physics interface. All physics interfaces have their own tags (**Name**). For example, if a Heat Transfer in Fluids interface is included in the component, the **Temperature (ht)** option is available for  $T$ .

You can also select **User defined** from the **Temperature** model input in order to manually prescribe  $T$



## PHASE FIELD PARAMETERS

Define the following phase field parameters. Enter a value or expression for the:

**Parameter controlling interface thickness**  $\epsilon$ . The default expression is `terpf.ep_default*sqrt(2)`, a value proportional to the maximum element size with a proportionality factor dependent on the ratio of the maximum and minimum element sizes in the domain.

- **Mobility tuning parameter**  $M_0$ . This parameter determines the time scale of the Cahn–Hilliard diffusion and it thereby also governs the diffusion-related time scale for the interface. Keep the mobility tuning parameter value large enough to maintain a constant interface thickness but still low enough to not damp the convective motion. A too high mobility can also lead to excessive diffusion of droplets. By default,  $M_0$  is set to  $1 \cdot 10^{-4} \text{ m}^3/\text{s}$ , which is a good starting point for most models.
- **Additional free bulk energy**  $\Lambda$ . When needed, add a user-defined expression for the additional bulk energy.

## SURFACE TENSION

Specify the surface tension coefficients for the three types of interfaces present.

- To use a predefined expression, select **Library coefficient, liquid/gas interface** or **Library coefficient, liquid/liquid interface**. Then select an option from the list that displays below (**Water/Air**, **Glycerol/Air**, and so forth).
- For **User defined**, enter a value or expression for the surface tension coefficient.

The predefined correlations are based on the data in [Ref. 3](#), [Ref. 4](#), and [Ref. 5](#).

## CONVECTION

Enter values or expressions for the components ( $u$ ,  $v$ , and  $w$  in 3D, for example) of the **Velocity field**  $u$ . The applied velocity field transports the phase field variables through convection.

## *Initial Values*

---

The **Initial Values** node adds initial values for the phase field variables that can serve as initial conditions for a transient simulation.

## INITIAL VALUES

Enter initial values or expressions for **Phase field variable A** and the **Phase field variable B** corresponding to the mass fraction of the respective phase. The value must be in the range from 0 to 1.

## *Inlet*

---

This condition should be used on boundaries for which there is a net convective flow of the phases into the adjacent domain.

## INLET

Specify phase field variables, corresponding to the volume fractions, of phase  $A$  and phase  $B$ . The value must be in the range from 0 to 1. The volume fraction of phase  $C$  will be evaluated such that the sum of all mass fractions equals one.

Mathematically this boundary condition imposes

$$\phi_i = \phi_{i,0}, \mathbf{n} \cdot \frac{M_0}{\Sigma_i} \nabla \eta_i = 0$$

for phases  $i = A$  and  $B$ .

## *Outlet*

---

This condition should be used on boundaries for which there is a net convective outflow from the domain.

Mathematically this boundary condition imposes

$$\mathbf{n} \cdot \frac{M_0}{\Sigma_i} \nabla \eta_i = 0$$



for phases  $i = A$  and  $B$ .

### *Symmetry*

---

Use this condition on boundaries that represent a geometrical symmetry line across which the flow of the fluid phases is zero.

Mathematically this boundary condition imposes

$$\mathbf{n} \cdot \frac{M_0}{\Sigma_i} \nabla \eta_i = 0$$

for phases  $i = A$  and  $B$ .

### *Wetted Wall*

---

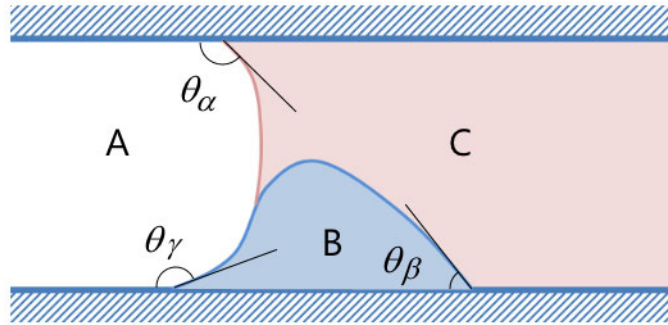
Use this node to represent a solid wall along which the contact angles between the separating interfaces and the wall should be prescribed.

#### **WETTED WALL**

Enter values or expressions for the contact angles for the three interfaces types:

- The **Contact angle of interface from phase A to phase C**  $\theta_\alpha$ .
- The **Contact angle of interface from phase B to phase C**  $\theta_\beta$ .
- The **Contact angle of interface from phase A to phase B**  $\theta_\gamma$ .

The definition of the contact angles with respect to the wall are shown in the figure below.



*Figure 6-1: Definitions of the contact angels in the Wetted Wall feature of the Ternary Phase Field interface.*

# Theory for the Level Set Interface

Fluid flow with moving interfaces or boundaries occur in a number of different applications, such as *fluid-structure interaction*, multiphase flows, and flexible membranes moving in a liquid. One way to track moving interfaces is to use a level set method. A certain contour line of the globally defined function, the level set function, then represents the interface between the phases. For the fluid-fluid interface can be advected with an arbitrary velocity field.

In this section:

- [The Level Set Method](#)
- [Conservative and Nonconservative Form](#)
- [Initializing the Level Set Function](#)
- [Variables For Geometric Properties of the Interface](#)
- [Reference for the Level Set Interface](#)

## *The Level Set Method*

---

The *level set method* is a technique to represent moving interfaces or boundaries using a fixed mesh. It is useful for problems where the computational domain can be divided into two domains separated by an interface. Each of the two domains can consist of several parts. [Figure 6-2](#) shows an example where one of the domains consists of two separated parts. The interface is represented by a certain level set or isocontour of a globally defined function, the level set function  $\phi$ . In COMSOL Multiphysics,  $\phi$  is a smooth step function that equals zero (0) in one domain and one (1) in the other. Across the interface, there is a smooth transition from zero to one. The interface is

defined by the 0.5 isocontour, or level set, of  $\phi$ . Figure 6-3 shows the level set representation of the interface in Figure 6-2.

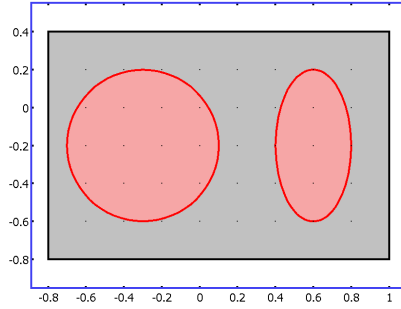


Figure 6-2: An example of two domains divided by an interface. In this case, one of the domains consists of two parts. Figure 6-3 shows the corresponding level set representation.

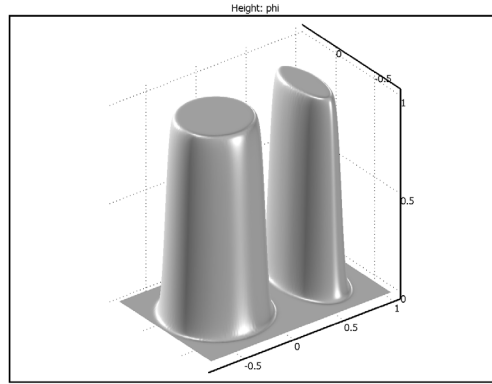


Figure 6-3: A surface plot of the level set function corresponding to Figure 6-2.

The physics interface solves Equation 6-2 in order to move the interface with the velocity field  $\mathbf{u}$ :

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = \gamma \nabla \cdot \left( \epsilon_{ls} \nabla \phi - \phi(1 - \phi) \frac{\nabla \phi}{|\nabla \phi|} \right) \quad (6-2)$$

The terms on the left-hand side give the correct motion of the interface, while those on the right-hand side are necessary for numerical stability. The parameter,  $\epsilon_{ls}$ , determines the thickness of the region where  $\phi$  varies smoothly from zero to one and is typically of the same order as the size of the elements of the mesh. By default,  $\epsilon_{ls}$  is

constant within each domain and proportional to the largest value of the mesh size,  $h$ , within the domain. The parameter  $\gamma$  determines the amount of reinitialization or stabilization of the level set function. It needs to be tuned for each specific problem. If  $\gamma$  is too small, the thickness of the interface might not remain constant and oscillations in  $\phi$  can appear because of numerical instabilities. On the other hand, if  $\gamma$  is too large the interface moves incorrectly. A suitable value for  $\gamma$  is the maximum magnitude of the velocity field  $\mathbf{u}$ .

### *Conservative and Nonconservative Form*

---

If the velocity is divergence free, that is, if

$$\nabla \cdot \mathbf{u} = 0 \quad (6-3)$$

the volume (area for 2D problems) bounded by the interface should be conserved if there is no inflow or outflow through the boundaries. To obtain exact numerical conservation in cases where Equation 6-3 is not fully satisfied, switch to the conservative form

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\mathbf{u} \phi) = \gamma \nabla \cdot \left( \varepsilon_{\text{ls}} \nabla \phi - \phi(1 - \phi) \frac{\nabla \phi}{|\nabla \phi|} \right) \quad (6-4)$$

in Settings window for .

Using the conservative level set form, exact numerical conservation of the integral of  $\phi$  is obtained. However, the nonconservative form is better suited for numerical calculations and usually converges more easily. The nonconservative form, which is the default form, only conserves the integral of the level set function approximately, but this level of conservation is sufficient for most applications.

### *The Level Set Equation in Porous Domains*

---

In a porous domain, the equation that determines the evolution of the interface is

$$\frac{\partial \phi}{\partial t} + \mathbf{u}_{\text{pore}} \cdot \nabla \phi = \gamma \nabla \cdot \left( \varepsilon_{\text{ls}} \nabla \phi - \phi(1 - \phi) \frac{\nabla \phi}{|\nabla \phi|} \right) \quad (6-5)$$

Here  $\mathbf{u}_{\text{pore}}$  is the pore velocity. Instead using the Darcy velocity  $\mathbf{u} = \mathbf{u}_{\text{pore}} / \varepsilon_{\text{p}}$ , multiplying the equation with the porosity  $\varepsilon_{\text{p}}$ , and rescaling  $\gamma$  such that it again can be

set proportional to the maximum magnitude of the velocity field  $\mathbf{u}$ , the equation can be rewritten as:

$$\varepsilon_p \frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = \gamma \nabla \cdot \left( \varepsilon_{ls} \nabla \phi - \phi(1 - \phi) \frac{\nabla \phi}{|\nabla \phi|} \right) \quad (6-6)$$

This is the nonconservative form of the level set equation that is solved in the Porous Medium feature.

To obtain exact numerical conservation of the integral of  $\varepsilon_p \phi$ , switch to the conservative form

$$\frac{\partial}{\partial t} (\varepsilon_p \phi) + \nabla \cdot (\mathbf{u} \phi) = \gamma \nabla \cdot \left( \varepsilon_{ls} \nabla \phi - \phi(1 - \phi) \frac{\nabla \phi}{|\nabla \phi|} \right) \quad (6-7)$$

The conservative form can be derived from the nonconservative form if it is assumed that the velocity field  $\mathbf{u}$  satisfies the following continuity equation:

$$\frac{\partial \varepsilon_p}{\partial t} + \nabla \cdot \mathbf{u} = 0 \quad (6-8)$$

### *Initializing the Level Set Function*

---

If the study type Transient with Phase Initialization is used in the model, the level set variable is first initialized so that it varies smoothly between zero and one over the interface. For that study, two study steps are created, Phase Initialization and Time Dependent. The Phase Initialization step solves for the distance to the initial interface,  $D_{wi}$ . The Time Dependent step then uses the initial condition for the level set function according to the following expression:


$$\phi_0 = \frac{1}{1 + e^{-D_{wi}/\varepsilon}}$$

in domains initially outside the interface and

$$\phi_0 = \frac{1}{1 + e^{D_{wi}/\varepsilon}}$$

in domains initially inside the interface. Here, inside refers to domains where  $\phi < 0.5$  and outside refers to domains where  $\phi > 0.5$ .



If the **Transient with Phase Initialization** () study is being used, then, for the initialization to work, it is crucial that there are two **Initial Values** nodes. One of the Initial Values nodes should use **Specify phase: Fluid 1** ( $\phi = 0$ ) and the other **Specify phase: Fluid 2** ( $\phi = 1$ ). The initial fluid-fluid interface is then automatically placed on all interior boundaries between both phases. The smooth initialization is not used in domains with **User defined** initial values.



- [Studies and Solvers](#) and [Time Dependent with Initialization](#) in the *COMSOL Multiphysics Reference Manual*

### *Variables For Geometric Properties of the Interface*

Geometric properties of the interface are often needed. The unit normal to the interface is given by

$$\mathbf{n} = \frac{\nabla\phi}{|\nabla\phi|} \Big|_{\phi=0.5} \quad (6-9)$$

This variable is available in the physics interface as the interface normal `ls.intnorm`.

### *Reference for the Level Set Interface*

1. E. Olsson and G. Kreiss, “A Conservative Level Set Method for Two Phase Flow,” *J. Comput. Phys.*, vol. 210, pp. 225–246, 2005.

# Theory for the Phase Field in Fluids Interface

The Phase Field in Fluids Interface theory is described in this section:

- [About the Phase Field Method](#)
- [The Equations for the Phase Field Method](#)
- [Conservative and Nonconservative Forms](#)
- [Additional Sources of Free Energy](#)
- [Initializing the Phase Field Function](#)
- [Variables and Expressions](#)
- [Reference for the Phase Field in Fluids Interface](#)

## *About the Phase Field Method*

---

The phase field method offers an attractive alternative to more established methods for solving multiphase flow problems. Instead of directly tracking the interface between two fluids, the interfacial layer is governed by a phase field variable,  $\phi$ . The surface tension force is added to the Navier–Stokes equations as a body force by multiplying the chemical potential of the system by the gradient of the phase field variable.

The evolution of the phase field variable is governed by the *Cahn–Hilliard equation*, which is a 4th-order PDE. The Phase Field in Fluids interface decomposes the Cahn–Hilliard equation into two second-order PDEs.

For the level set method, the fluid interface is simply advected with the flow field. The Cahn–Hilliard equation, on the other hand, does not only convect the fluid interface, but it also ensures that the total energy of the system diminishes correctly. The phase field method thus includes more physics than the level set method.

The free energy of a system of two immiscible fluids consists of mixing, bulk distortion, and anchoring energy. For simple two-phase flows, only the mixing energy is retained, which results in a rather simple expression for the free energy.

## *The Equations for the Phase Field Method*

---

The free energy is a functional of a dimensionless phase field parameter,  $\phi$ :



$$F(\phi, \nabla\phi, T) = \int \left( \frac{1}{2} \varepsilon^2 |\nabla\phi|^2 + f(\phi, T) \right) dV = \int f_{\text{tot}} dV$$

where  $\varepsilon$  is a measure of the interface thickness. Equation 6-10 describes the evolution of the phase field parameter:

$$\frac{\partial\phi}{\partial t} + (\mathbf{u} \cdot \nabla)\phi = \nabla \cdot \gamma \nabla \left( \frac{\partial f_{\text{tot}}}{\partial \phi} - \nabla \cdot \frac{\partial f_{\text{tot}}}{\partial \nabla\phi} \right) \quad (6-10)$$

where  $f_{\text{tot}}$  (SI unit: J/m<sup>3</sup>) is the total free energy density of the system, and  $\mathbf{u}$  (SI unit: m/s) is the velocity field for the advection. The right-hand side of Equation 6-10 aims to minimize the total free energy with a relaxation time controlled by the mobility  $\gamma$  (SI unit: m<sup>3</sup>·s/kg).

The free energy density of an isothermal mixture of two immiscible fluids is the sum of the mixing energy and elastic energy. The mixing energy assumes the Ginzburg–Landau form:

$$f_{\text{mix}}(\phi, \nabla\phi) = \frac{1}{2} \lambda |\nabla\phi|^2 + \frac{\lambda}{4\varepsilon} (\phi^2 - 1)^2$$

where  $\phi$  is the dimensionless phase field variable, defined such that the volume fraction of the components of the fluid are  $(1 + \phi)/2$  and  $(1 - \phi)/2$ . The quantity  $\lambda$  (SI unit: N) is the mixing energy density and  $\varepsilon$  (SI unit: m) is a capillary width that scales with the thickness of the interface. These two parameters are related to the surface tension coefficient,  $\sigma$  (SI unit: N/m), through the equation

$$\sigma = \frac{2\sqrt{2}\lambda}{3\varepsilon} \quad (6-11)$$

The PDE governing the phase field variable is the Cahn–Hilliard equation:

$$\frac{\partial\phi}{\partial t} + \mathbf{u} \cdot \nabla\phi = \nabla \cdot \gamma \nabla G \quad (6-12)$$

where  $G$  (SI unit: Pa) is the chemical potential and  $\gamma$  (SI unit: m<sup>3</sup>·s/kg) is the mobility. The mobility determines the time scale of the Cahn–Hilliard diffusion and must be large enough to retain a constant interfacial thickness but small enough so that the convective terms are not overly damped. In COMSOL Multiphysics the mobility is determined by a mobility tuning parameter that is a function of the interface thickness  $\gamma = \chi \varepsilon^2$ . The chemical potential is:

$$G = \lambda \left[ -\nabla^2 \phi + \frac{\phi(\phi^2 - 1)}{\varepsilon^2} \right] \quad (6-13)$$

The Cahn–Hilliard equation forces  $\phi$  to take a value of 1 or  $-1$  except in a very thin region on the fluid-fluid interface. The Phase Field in Fluids interface breaks Equation 6-12 up into two second-order PDEs:

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = \nabla \cdot \frac{\gamma \lambda}{\varepsilon^2} \nabla \psi \quad (6-14)$$

$$\psi = -\nabla \cdot \varepsilon^2 \nabla \phi + (\phi^2 - 1)\phi \quad (6-15)$$

### *Conservative and Nonconservative Forms*

---

If the velocity field is divergence free, that is, if

$$\nabla \cdot \mathbf{u} = 0 \quad (6-16)$$

the volume (area for 2D problems) bounded by the interface should be conserved if there is no inflow or outflow through the boundaries. To obtain exact numerical conservation in cases where Equation 6-16 is not fully satisfied, switch to the conservative form

$$\frac{\partial \phi}{\partial t} + \nabla \cdot \mathbf{u} \phi = \nabla \cdot \frac{\gamma \lambda}{\varepsilon^2} \nabla \psi$$

Using the conservative phase field form, exact numerical conservation of the integral of  $\phi$  is obtained. However, the nonconservative form is better suited for numerical calculations and usually converges more easily. The nonconservative form, which is the default form, only conserves the integral of the phase field function approximately, but this is sufficient for most applications.

When the Phase Field in Fluids interface is coupled to a fluid flow interface by means of , conservation of the integral of  $\phi$  using the nonconservative form can be improved provided that the discretization order of the phase field variables is equal to or lower than the order of the pressure.

### *Additional Sources of Free Energy*

---

In some cases, the expression for the free energy can include other sources. It is possible to incorporate these by modifying [Equation 6-15](#):

$$\Psi = -\nabla \cdot \varepsilon^2 \nabla \phi + (\phi^2 - 1)\phi + \left(\frac{\varepsilon^2}{\lambda}\right) \frac{\partial f}{\partial \phi} \quad (6-17)$$

where  $f$  is a user-defined free energy (SI unit: J/m<sup>3</sup>).



The expression for the external free energy must be manually differentiated with respect to  $\phi$  and then entered into the  $\partial f / \partial \phi$  field. In most cases, the external free energy is zero.

---

### *Initializing the Phase Field Function*

---

If the study type Transient with Phase Initialization is used in the model, the phase field variable is first initialized so that it varies smoothly between zero and one over the interface. For this study, two study steps are created, Phase Initialization and Time Dependent. The Phase Initialization step solves for the distance to the initial interface,  $D_{wi}$ . The Time Dependent step then uses the initial condition for the phase field function according to the expression


$$\phi_0 = -\tanh\left(\frac{D_{wi}}{\sqrt{2}\varepsilon}\right)$$

in Fluid 1 and

$$\phi_0 = \tanh\left(\frac{D_{wi}}{\sqrt{2}\varepsilon}\right)$$

in Fluid 2. These expressions are based on a steady, analytic solution to [Equation 6-14](#) and [Equation 6-15](#) for a straight, nonmoving interface.



If the **Transient with Phase Initialization** () study is being used, for the initialization to work it is crucial that there are two **Initial Values** nodes. One of the Initial Values nodes is set to **Specify phase: Fluid 1** ( $\phi = -1$ ) and the other to **Specify phase: Fluid 2** ( $\phi = 1$ ). The initial fluid-fluid interface is then automatically placed on all interior boundaries between both phases. The smooth initialization is not used in domains with **User defined** initial values.



[Studies and Solvers](#) in the *COMSOL Multiphysics Reference Manual*

### *Variables and Expressions*

The physics interface defines the chemical potential, which can be rewritten in terms of the dependent variable  $\psi$ ,

$$G = \frac{\lambda\psi}{\varepsilon^2}$$

and the surface tension force  $\mathbf{F} = G\nabla\phi$ .

Geometric properties of the interface are often needed. The unit normal to the interface is given by

$$\mathbf{n} = \frac{\nabla\phi}{|\nabla\phi|} \Big|_{\phi=0.5} \quad (6-18)$$

This variable is available in the physics interface as the interface normal `pf.intnorm`.

The mean curvature (SI unit: 1/m) of the interface can be computed by entering the following expression:

$$\kappa = 2(1+\phi)(1-\phi)\frac{G}{\sigma}$$

### *Reference for the Phase Field in Fluids Interface*

---

1. P. Yue, C. Zhou, J.J. Feng, C.F. Ollivier-Gooch, and H.H. Hu, “Phase-field Simulations of Interfacial Dynamics in Viscoelastic Fluids Using Finite Elements with Adaptive Meshing,” *J. Comp. Phys.*, vol. 219, pp. 47–67, 2006.

# Theory for the Ternary Phase Field Interface

The Ternary Phase Field Interface theory is described in this section:

- [About the Phase Field Method](#)
- [The Equations of the Ternary Phase Field Method](#)
- [Reference for the Phase Field in Fluids Interface](#)

## *About the Phase Field Method*

---

The phase field method offers an attractive alternative to more established methods for solving multiphase flow problems. Instead of directly tracking the interface between two fluids, the separating interface is identified as the region across which the phase field variables of the two adjacent phases varies between its two limiting values. The surface tension force is added to the Navier–Stokes equations as a body force by multiplying the chemical potentials of the phases by the gradient of the corresponding phase field variable.

The evolution of each phase field variable is governed by the *Cahn–Hilliard equation*, which is a 4th-order PDE. The Ternary Phase Field interface decomposes each Cahn–Hilliard equation into two second-order PDEs.

For the level set method, the fluid interface is simply advected with the flow field. The Cahn–Hilliard equation, on the other hand, does not only convect the fluid interface, but it also ensures that the total energy of the system diminishes correctly. The phase field method thus includes more physics than the level set method.

The free energy of a system of immiscible fluids consists of mixing, bulk distortion, and anchoring energy. For simple three-phase flows, only the mixing energy is retained.

## *The Equations of the Ternary Phase Field Method*

---

The ternary phase field model implemented in COMSOL is based the work of Boyer and co-workers in [Ref. 2](#). The model is designed in order to study the evolution of the three immiscible phases, denoted phase A, phase B, and phase C, respectively. Each phase is represented by a phase field variable  $\phi$ , which takes values between 0 and 1. The phase field variables satisfies the constraint

$$\sum_{i=A,B,C} \phi_i = 0 \quad (6-19)$$

and density of each phase is assumed to be constant. This implies that the phase field variable corresponds directly to the volume fraction of the phase in question.

The free energy of the three phase system is defined as a function of the phase field variables in the manner of:

$$F = \sigma_{AB}\phi_A^2\phi_B^2 + \sigma_{AC}\phi_A^2\phi_C^2 + \sigma_{BC}\phi_B^2\phi_C^2 + \phi_A\phi_B\phi_C(\Sigma_A\phi_A + \Sigma_B\phi_B + \Sigma_C\phi_C) + \Lambda\phi_A^2\phi_B^2\phi_C^2 \quad (6-20)$$

Here the  $\sigma_{ij}$  denotes the surface tension coefficient of the interface separating phase  $i$  and  $j$ , and the capillary parameters  $\Sigma_i$  are defined as

$$\Sigma_i = \sigma_{ij} + \sigma_{ik} - \sigma_{jk} \quad (6-21)$$

and  $\Lambda$  is a function or parameter specifying the additional free bulk energy. By default  $\Lambda$  is zero. In this case it can be seen that the free energy in Equation 6-20 represents the mixing energy, since only interfaces between two phases (where two phase field variables varies between the limiting values) contributes to the free energy.

The Cahn–Hilliard equations to be solved for each phase  $p = A, B, C$  are

$$\begin{aligned} \frac{\partial \phi_i}{\partial t} + \frac{\partial u_j \phi_i}{\partial x_j} &= \frac{\partial}{\partial x_j} \left( \frac{M_0}{\Sigma_i} \frac{\partial \eta_i}{\partial x_j} \right) \\ \eta_i &= \frac{4\Sigma_T}{\varepsilon} \sum_{j \neq i} \left( \frac{1}{\Sigma_j} \left( \frac{\partial F}{\partial \phi_i} - \frac{\partial F}{\partial \phi_j} \right) \right) - \frac{3}{4} \varepsilon \Sigma_j \frac{\partial}{\partial x_j} \left( \frac{\partial \eta_i}{\partial x_j} \right) \end{aligned} \quad (6-22)$$

It can be noted that the Cahn–Hilliard equation is originally a 4th-order PDE. In COMSOL Multiphysics, that PDE is split up into two second-order PDEs by introducing an additional dependent variable, the generalized potential  $\eta$ , one for each phase. In order to satisfy Equation 6-20, two sets of the equations shown in Equation 6-22 are solved, those for phase A and phase B. The phase field variable, and correspondingly the mass fraction, for fluid C is computed from Equation 6-20.

In Equation 6-22  $\varepsilon$  (SI unit: 1/m) is a parameter controlling the interface thickness,  $M_0$  (SI unit:  $\text{m}^3/\text{s}$ ) is a molecular mobility parameter, and the parameter  $\Sigma_T$  is defined as

$$\frac{3}{\Sigma_T} = \frac{1}{\Sigma_A} + \frac{1}{\Sigma_B} + \frac{1}{\Sigma_C} \quad (6-23)$$


### *Reference for the Ternary Phase Field Interface*

---

2. F. Boyer, C. Lapuerta, S. Minjeaud, B. Piar, and M. Quintard, “Cahn–Hilliard/Navier–Stokes model for the simulation of three-phase flows”, *Transport in Porous Media*, vol. 28, pp. 463–484, 2010.
3. J. J. Jasper, “The Surface Tension of Pure Liquid Compounds”, *J. Phys. Chem. Ref. Data.*, vol. 1, no. 4, pp. 841-1009, 1972.
4. R. C. Weast, ed., “CRC Handbook of Chemistry and Physics”, ed. 69, MIT Press, 1988.
5. A. G. Gaonkar, “Interfacial Tensions of Vegetable Oil/Water Systems: Effect of Oil Purification”, *J. Am. Oil Chem. Soc.*, vol. 66, no. 8, pp. 1090-1092, 1989.





# Chemical Species Transport Interfaces

The physics interfaces under the **Chemical Species Transport** branch (  ) accommodate all types of material transport that can occur through diffusion and convection. When the material is transported by a flowing medium, the Reacting Flow multiphysics coupling automatically defines the dependencies of fluid properties on the species concentrations, and the convective velocity for the species transport.

In this chapter:

- [The Transport of Diluted Species Interface](#)
- [The Reacting Flow, Diluted Species Multiphysics Interface](#) (in the *COMSOL Multiphysics Reference Manual*)
- [Theory for the Transport of Diluted Species Interface](#)

# The Transport of Diluted Species Interface

The **Transport of Diluted Species (tds)** interface () found under the **Chemical Species Transport** branch () is used to calculate the concentration field of a dilute solute in a solvent. Transport and reactions of the species dissolved in a gas, liquid, or solid can be handled with this interface. The driving forces for transport can be diffusion by Fick's law, convection when coupled to a flow field, and migration, when coupled to an electric field.

The interface supports simulation of transport by convection and diffusion in 1D, 2D, and 3D as well as for axisymmetric components in 1D and 2D. The dependent variable is the molar concentration,  $c$ . Modeling multiple species transport is possible, whereby the physics interface solves for the molar concentration,  $c_i$ , of each species  $i$ .

## 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 `tds`.

## DOMAIN SELECTION

If any part of the model geometry should not partake in the mass transfer model, remove that part from the selection list.

## OUT-OF-PLANE THICKNESS

For 2D components, the **Thickness** field (default value: 1 m) defines a parameter for the thickness of the geometry perpendicular to the two-dimensional cross-section. Both constant and varying thicknesses are supported. The value of this parameter is used, among other things, to automatically calculate molar flow rates from the total molar flux.

### CROSS-SECTIONAL AREA

For 1D components, enter a **Cross-sectional area**  $A_c$  (SI unit:  $\text{m}^2$ ) to define a parameter for the area of the geometry perpendicular to the 1D component. Both constant and varying areas are supported. The value of this parameter is used, among other things, to automatically calculate molar flow rates from the total molar flux. The default is  $1 \text{ m}^2$ .

### TRANSPORT MECHANISMS

Mass transport due to diffusion is always included. Use the check boxes available under **Additional transport mechanisms** to control other transport mechanisms.

Note: Some of the additional transport mechanisms listed below are only available in certain products. For details see <https://www.comsol.com/products/specifications/>.

- By default, the **Convection** check box is selected. Clear the check box to disable convective transport.
- Select the **Migration in electric field** check box to activate transport of ionic species in an electric field. See further the theory section [Adding Transport Through Migration](#).



Transport of ionic species in an electric field is available in a limited set of add-on products. See <https://www.comsol.com/products/specifications/> for more details on availability.

#### *Mass Transport in Porous Media*

The **Mass transport in porous media** check box activates functionality specific to species transport in porous media:

- [Porous Medium](#)
- [Unsaturated Porous Medium](#)
- [Porous Electrode Coupling](#)

- Volatilization
- Species Source




Note: Mass transport in porous media is only available in a limited set of add-on products. See <https://www.comsol.com/products/specifications/> for more details on availability.

### CONSISTENT STABILIZATION


To display this sections, click the **Show** button (  ) and select **Stabilization**.

- When the **Crosswind diffusion** check box is selected, a weak term that reduces spurious oscillations is added to the transport equation. The resulting equation system is always nonlinear. There are two options for the **Crosswind diffusion type**:
  - **Do Carmo and Galeão** — the default option. This type of crosswind diffusion reduces undershoots and overshoots to a minimum but can in rare cases give equation systems that are difficult to fully converge.
  - **Codina** — this option is less diffusive compared to the Do Carmo and Galeão option but can result in more undershoots and overshoots. It is also less effective for anisotropic meshes. The Codina option activates a text field for the **Lower gradient limit**  $g_{lim}$ . Its defaults to  $0.1 [\text{mol}/\text{m}^3] / \text{tds.helem}$ , where  $\text{tds.helem}$  is the local element size.
- For both consistent stabilization methods, select an **Equation residual**. **Approximate residual** is the default and means that derivatives of the diffusion tensor components are neglected. This setting is usually accurate enough and is computationally faster. If required, select **Full residual** instead.

### INCONSISTENT STABILIZATION


To display this section, click the **Show** button (  ) and select **Stabilization**. By default, the **Isotropic diffusion** check box is not selected, because this type of stabilization adds artificial diffusion and affects the accuracy of the original problem. However, this option can be used to get a good initial guess for under resolved problems.

### ADVANCED SETTINGS

To display this section, click the **Show** button (  ) and select **Advanced Physics Options**. Normally these settings do not need to be changed. Select a **Convective term** —

**Nonconservative form** (the default) or **Conservative form**. The conservative formulation should be used for compressible flow. See [for more information](#).

## DISCRETIZATION

To display all settings available in this section, click the **Show** button (  ) and select **Advanced Physics Options**.

The **Compute boundary fluxes** check box is activated by default so that COMSOL Multiphysics computes predefined accurate boundary flux variables. When this option is selected, the solver computes variables storing accurate boundary fluxes from each boundary into the adjacent domain.

If the check box is cleared, the COMSOL Multiphysics software instead computes the flux variables from the dependent variables using extrapolation, which is less accurate in postprocessing results but does not create extra dependent variables on the boundaries for the fluxes.

The flux variables affected in the interface are:

- `ndflux_c` (where *c* is the dependent variable for the concentration). This is the normal diffusive flux and corresponds to the boundary flux when diffusion is the only contribution to the flux term.
- `ntflux_c` (where *c* is the dependent variable for the concentration). This is the normal total flux and corresponds to the boundary flux plus additional transport terms, for example, the convective flux when you use the nonconservative form.

Also the **Apply smoothing to boundary fluxes** check box is available if the previous check box is selected. The smoothing can provide a more well-behaved flux value close to singularities.



For details about the boundary fluxes settings, see [Computing Accurate Fluxes](#) in the *COMSOL Multiphysics Reference Manual*.

The **Value type when using splitting of complex variables** setting should in most pure mass transfer problems be set to **Real**, which is the default. It makes sure that the dependent variable does not get affected by small imaginary contributions, which can occur, for example, when combining a Time Dependent or Stationary study with a frequency-domain study. For more information, see [Splitting Complex-Valued Variables](#) in the *COMSOL Multiphysics Reference Manual*.



## DEPENDENT VARIABLES

The dependent variable name is the **Concentration**  $c$  by default. The names must be unique with respect to all other dependent variables in the component.



Add or remove species variables in the model and also change the names of the dependent variables that represent the species concentrations.

Enter the **Number of species**. Use the **Add concentration** (  ) and **Remove concentration** (  ) buttons as needed.

## FURTHER READING

	<ul style="list-style-type: none"><li>• <a href="#">Theory for Transport of Diluted Species</a> in the <i>COMSOL Multiphysics Reference Manual</i>.</li><li>• <a href="#">Numerical Stabilization</a> in the <i>COMSOL Multiphysics Reference Manual</i>.</li><li>• In the <i>COMSOL Multiphysics Reference Manual</i>, see <a href="#">Table 2-4</a> for links to common sections and <a href="#">Table 2-5</a> for common feature nodes. You can also search for information: press F1 to open the <b>Help</b> window or Ctrl+F1 to open the <b>Documentation</b> window.</li></ul>
	<ul style="list-style-type: none"><li>• <i>Effective Diffusivity in Porous Materials</i>: Application Library path <b>COMSOL_Multiphysics/Diffusion/effective_diffusivity</b></li><li>• <i>Micromixer</i>: Application Library path <b>COMSOL_Multiphysics/Fluid_Dynamics/micromixer</b></li></ul>

## *The Transport of Diluted Species in Porous Media Interface*

This interface (  ), found under the **Chemical Species Transport** branch (  ), is used to calculate the species concentration and transport in free and porous media. The interface is the same as the *Transport of Diluted Species* interface but it uses other defaults: The [Mass Transport in Porous Media](#) property is selected, and a [Porous Medium](#) node is added by default. The interface includes reaction rate expressions and solute sources for modeling of species transport and reaction in porous media.

This interface is dedicated to modeling transport in porous media, including immobile and mobile phases, where the chemical species may be subjected to diffusion, convection, migration, dispersion, adsorption, and volatilization in porous media. It

supports cases where either the solid phase substrate is exclusively immobile, or when a gas-filling medium is also assumed to be immobile.

It applies to one or more diluted species or solutes that move primarily within a fluid that fills (saturated) or partially fills (unsaturated) the voids in a solid porous medium. The pore space not filled with fluid contains an immobile gas phase. Models including a combination of porous media types can be studied.

The main feature nodes are the **Porous Medium** and **Unsaturated Porous Medium** nodes, which add the equations for the species concentrations and provide an interface for defining the properties of the porous media, as well as additional properties governing adsorption, volatilization, dispersion and diffusion, migration, and the velocity field to model convection.



The physics interface can be used for stationary and time-dependent analysis.

When this physics interface is added, these default nodes are also added to the **Model Builder** — **Porous Medium, No Flux** (the default boundary condition), and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions, reaction rate expressions, and species sources. You can also right-click **Transport of Diluted Species in Porous Media** to select physics features from the context menu.

**SETTINGS**

The rest of the settings are the same as .

**FURTHER READING**

	<ul style="list-style-type: none"><li>• <a href="#">Mass Balance Equation for Transport of Diluted Species in Porous Media</a> in the <i>COMSOL Multiphysics Reference Manual</i>.</li><li>• <a href="#">Domain, Boundary, and Pair Nodes for the Transport of Diluted Species Interface</a></li></ul>
	<ul style="list-style-type: none"><li>• <i>Variably Saturated Flow and Transport — Sorbing Solute</i>: Application Library path <b>Subsurface_Flow_Module/Solute_Transport/sorbing_solute</b> Web link: <a href="https://www.comsol.com/model/variably-saturated-flow-and-transport-8212-sorbing-solute-490">https://www.comsol.com/model/variably-saturated-flow-and-transport-8212-sorbing-solute-490</a></li></ul>

### *Domain, Boundary, and Pair Nodes for the Transport of Diluted Species Interface*

---

has the following domain, boundary, point, and pair nodes, listed in alphabetical order, available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or by right-clicking to access the context menu (all users).



- To add a node, go to the **Physics** toolbar, no matter what operating system you are using.
- Contextual subnodes (attributes) are available by clicking the parent node in the Model Builder, and then selecting the subnode from the **Attributes** menu.



- Adsorption
- Concentration
- Continuity on Interior Boundaries
- Dispersion
- Electrode Surface Coupling
- Equilibrium Reaction
- Fast Irreversible Surface Reaction
- Fluid
- Flux
- Flux Discontinuity
- Fracture
- Gas
- Inflow
- Initial Values
- Line Mass Source
- Liquid
- Mass-Based Concentrations
- No Flux
- Open Boundary
- Out-of-Plane Flux
- Outflow
- Partition Condition
- Periodic Condition
- Point Mass Source
- Porous Matrix
- Porous Medium
- Porous Electrode Coupling
- Reaction Coefficients
- Reactions
- Solid
- Species Properties
- Species Source
- Surface Reactions
- Surface Equilibrium Reaction
- Symmetry
- Thin Diffusion Barrier
- Thin Impermeable Barrier
- Transport Properties
- Unsaturated Porous Medium
- Volatilization

Some features require certain add-on modules. For details see <https://www.comsol.com/products/specifications/>



For axisymmetric components, COMSOL Multiphysics takes the axial symmetry boundaries (at  $r = 0$ ) into account and automatically adds an **Axial Symmetry** node that is valid on boundaries representing the symmetry axis.



In the *COMSOL Multiphysics Reference Manual*, see [Table 2-4](#) for links to common sections and [Table 2-5](#) for common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

---

### *Prescribing Conditions on Fluid-Solid Interfaces*

---

Defining conditions on interfaces between different phases, typically a wall in contact with a fluid, it is worthwhile to be careful with how variables in the conditions are evaluated. COMSOL Multiphysics supports that variables are defined differently on adjacent domains (or any other geometry dimension). Evaluating a variable with differing definitions, the mean value is returned on a boundary separating the different domains. The density, pressure, or temperature are examples of variables that may differ in a fluid and a solid. In order to specify on which side a variable is evaluated, the [up and down](#) operators (described in the *COMSOL Multiphysics Reference Manual*) can be used.

---

### *Species Properties*

---

Use this node to define parameters specific to each of the species. It is available when the **Migration in electric field** check box is selected in the [Transport Mechanisms](#) section of the interface.

The node will change its label depending on what inputs are available in the node. If only **Charge** is available, it will be labeled “Species charges”.

#### **CHARGE**

Enter the charge number  $z_c$  (dimensionless, but requires a plus or minus sign) for each species.

#### **ACTIVITY**

This section is available when the **Species Activity** combo box is set to **Debye-Hückel** or **User defined**.

If **Debye-Hückel** is selected, enter the ion size number,  $a_0$ , for each species. If **User defined** is selected, enter the activity coefficient,  $f$ , for each species.

## *Transport Properties*

---

The settings in this node are dependent on the check boxes selected under [Transport Mechanisms](#) in the **Settings** window for the Transport of Diluted Species interface. It includes only the sections required by the activated transport mechanisms. It has all the equations defining transport of diluted species as well as inputs for the material properties.

### **MODEL INPUTS**

The temperature model input is always available. Select the source of the **Temperature**. For **User defined**, enter a value or expression for the temperature in the input field. This input option is always available.

You can also select the temperature solved for by a Heat Transfer interface added to the model component. These physics interfaces are available for selection in the **Temperature** list.

### **CONVECTION**

If transport by convection is active, the velocity field of the solvent needs to be specified. Select the source of the **Velocity field**. For **User defined**, enter values or expressions for the velocity components in the input fields. This input option is always available.

You can also select the velocity field solved for by a Fluid Flow interface added to the model component. These physics interfaces are available for selection in the **Velocity field** list.

### **DIFFUSION**

Use this section to specify diffusion coefficients describing the diffusion of each species in the solvent fluid.

Use the **Source** list to select to pick up diffusion coefficients defined in a material or a Chemistry interface.

Select **Material**, and a **Fluid material** to use a diffusion coefficient in a material available in the model. User-defined property groups including a diffusion coefficient output property are available for selection in the **Fluid diffusion coefficient** list.

Select **Chemistry** to use a diffusion coefficient from a Chemistry interface added to model component. Diffusion coefficients are automatically defined when **Calculate transport properties** is selected on the interface level of a Chemistry interface. All

defined diffusion coefficients are available for selection in the **Fluid diffusion coefficient** list.

For **User defined**, enter a value or expression for the **Fluid diffusion coefficient**  $D_c$  of each species in the corresponding input field. This can be a scalar value for isotropic diffusion or a tensor describing anisotropic diffusion. Select the appropriate tensor type — **Isotropic**, **Diagonal**, **Symmetric**, or **Full** that describes the diffusion transport, and then enter the values in the corresponding element (one value for each species).

Note that multiple species, as well as Migration in Electric fields (described below) is only available for certain COMSOL Multiphysics add-on products. See details: <https://www.comsol.com/products/specifications/>.

### MIGRATION IN ELECTRIC FIELD

This section is available when the **Migration in electric field** check box is selected. From the **Electric potential** list, select the source of the electric field.

- Enter a value or expression for the **Electric potential**  $V$ , which is **User defined**; this input option is always available.
- Select the electric potential solved by an AC/DC-based interface that has also been added to the model.
- Select the electric potential defined or solved by Electrochemistry interface that has been added to the component.



By default the **Mobility** is set to be calculated based on the species diffusivity and the temperature using the **Nernst-Einstein relation**. For **User defined**, and under **Mobility**, select the appropriate scalar or tensor type — **Isotropic**, **Diagonal**, **Symmetric**, or **Full** — and type in the value of expression of the mobility  $u_{m,c}$ .

Enter the **Charge number**  $z_c$  (dimensionless, but requires a plus or minus sign) for each species.

The temperature (if you are using mobilities based on the Nernst–Einstein relation) is taken from **Model Inputs** section.

Note that the migration in electric fields feature is only available in some COMSOL products. See details: <https://www.comsol.com/products/specifications/>.

## EXAMPLE MODELS

	<ul style="list-style-type: none"><li>• <i>Separation Through Dialysis</i>: Application Library path <b>Chemical Reaction_Engineering_Module/Mixing_and_Separation/dialysis</b> Web link: <a href="https://www.comsol.com/model/separation-through-dialysis-258">https://www.comsol.com/model/separation-through-dialysis-258</a></li></ul>
	<ul style="list-style-type: none"><li>• <i>Transport in an Electrokinetic Valve</i>: Application Library path <b>Microfluidics_Module/Fluid_Flow/electrokinetic_valve</b> Web link: <a href="https://www.comsol.com/model/transport-in-an-electrokinetic-valve-603">https://www.comsol.com/model/transport-in-an-electrokinetic-valve-603</a></li></ul>

## *Solid*

Use this node to describe diffusion in a solid phase domain. In this feature, the activities of the species are assumed to be ideal.

### DIFFUSION

Specify the diffusion coefficients describing the diffusion of each species in the solid.


Enter a value or expression for the **Diffusion coefficient**  $D_c$  of each species in the corresponding input field. This can be a scalar value for isotropic diffusion or a tensor describing anisotropic diffusion. Select the appropriate tensor type — **Isotropic**, **Diagonal**, or **Symmetric** that describes the diffusion transport, and then enter the values in the corresponding element (one value for each species).

### ADVANCED

This section is only available with the **Advanced Physics Options** selected and when the model contains a split between the material frame and the spatial frame. A split of these frames occurs when using a **Moving Mesh** node or a Structural Mechanics interface like **Solid Mechanics**. In such a model, the reference geometry (the material frame) differs from the deformed geometry (the spatial frame). The diffusive transport of species is solved for on the material frame, but the equation is adjusted for geometry deformations so that species conservation is achieved on the spatial frame. Use the **Diffusion model** to define the frame of reference for the diffusion tensor.

Select **Reference geometry based (Lagrangian)** to specify the diffusion tensor in the undeformed geometry, implying that the diffusion is defined in the material frame (indicated by coordinates  $X, Y, Z$ ). This is the preferred setting for an anisotropic diffusion in combination with solid body translation or rotation. Also in general when the diffusion tensor components have been measured in an undeformed geometry.

Select **Deformed geometry based (Eulerian)** to specify the diffusion tensor in the deformed geometry. This means that the diffusion is defined in the spatial frame (indicated by coordinates  $x, y, z$ ). Use this setting when the diffusion tensor components have been measured in a deformed geometry.

The **Advanced Physics Options** is found in the **Show More Options** dialog box, accessed by clicking the **Show More Options** button (  ).

Note that multiple species are only available for certain COMSOL Multiphysics add-on products. See details: <https://www.comsol.com/products/specifications/>.

### *Initial Values*

---

The **Initial Values** node specifies the initial values for the concentration of each species. These serve as an initial guess for a stationary solver or as initial conditions for a transient simulation.

#### **DOMAIN SELECTION**

If there are several types of domains with different initial values defined, it might be necessary to remove some domains from the selection. These are then defined in an additional **Initial Values** node.

#### **INITIAL VALUES**

Enter a value or expression for the initial value of the **Concentration** or concentrations,  $c_i$ . This also serves as a starting guess for stationary problems.

### *Mass-Based Concentrations*

---

Use the **Mass-Based Concentrations** node to add postprocessing variables for mass-based concentrations (SI unit:  $\text{kg}/\text{m}^3$ ) and mass fractions (dimensionless) for all species.

## MIXTURE PROPERTIES

The default **Solvent density**  $\rho_{\text{solvent}}$  is taken **From material**. For **User defined**, enter a value or expression manually. Define the **Molar mass** of each species, which is needed to calculate the mass-based concentration.

## Reactions

---

Use the **Reactions** node to account for the consumption or production of species through chemical reactions. Define the rate expressions as required.

## DOMAIN SELECTION

From the **Selection** list, choose the domains on which to define rate expression or expressions that govern the source term in the transport equations.

Several reaction nodes can be used to account for different reactions in different parts for the modeling geometry.

## REACTION RATES

Add a rate expression  $R_i$  (SI unit:  $\text{mol}/(\text{m}^3 \cdot \text{s})$ ) for species  $i$ . Enter a value or expression in the field. Note that if you have the *Chemistry* interface available, provided with the *Chemical Reaction Engineering Module*, the reaction rate expressions can be automatically generated and picked up using the drop-down list. For an example, see the application *Fine Chemical Production in a Plate Reactor* as linked below.

## REACTING VOLUME

This section is only available when the **Mass Transport in Porous Media** property is available and selected. See <https://www.comsol.com/products/specifications/> for more details on availability.

When specifying reaction rates for a species in porous media, the specified reaction rate may have the basis of the total volume, the pore volume, or in the case of using the volume of a particular phase (applicable when modeling an unsaturated porous medium).

- For **Total volume**, the reaction expressions, in  $\text{mol}/(\text{m}^3 \cdot \text{s})$ , are specified per unit volume of the model domain (multiplied by unity).
- For **Pore volume**, the reaction expressions, in  $\text{mol}/(\text{m}^3 \cdot \text{s})$ , are specified per unit volume of total pore space. The reaction expressions will be multiplied by the domain porosity,  $\varepsilon_p$ . ( $\varepsilon_p$  equals unity for nonporous domains).

- For **Liquid phase**, the reaction expressions, in  $\text{mol}/(\text{m}^3 \cdot \text{s})$ , are specified per unit volume of liquid in the pore space. The expressions will be multiplied by the liquid volume fraction  $\theta_l$ . ( $\theta_l$  equals the porosity ( $\varepsilon_p$ ) on Porous Medium domains).
- For **Gas phase**, the reaction expressions, in  $\text{mol}/(\text{m}^3 \cdot \text{s})$ , are multiplied by the gas volume fraction  $\theta_g = \varepsilon_p - \theta_l$ .  $\theta_g$  equals 0 for Porous Medium domains.

#### FURTHER READING

See the theory chapter on chemical species transport, starting with the section *Mass Balance Equation*.



- *Fine Chemical Production in a Plate Reactor*: Application Library path  
**Chemical\_Reaction\_Engineering\_Module/Reactors\_with\_Mass\_and\_Heat\_Transfer/plate\_reactor**  
Web link:  
<https://www.comsol.com/model/fine-chemical-production-in-a-plate-reactor-8589>

#### No Flux

This node is the default boundary condition on exterior boundaries. It should be used on boundaries across which there is no mass flux, typically solid walls where no surface reactions occur. The condition applied for each species corresponds to

$$-\mathbf{n} \cdot (-D \nabla c) = 0$$

where  $\mathbf{n}$  denotes the outward pointing normal of the boundary. When the mass transport includes migration of ionic species, the no flux condition is:

$$-\mathbf{n} \cdot (-D \nabla c - z u_m F c \nabla \phi) = 0$$

#### CONVECTION

By default, the feature prescribes a vanishing flux due to diffusion and migration in an electric field. This is the appropriate no flux condition when the relative convective velocity at the boundary is zero. When the fluid velocity at the boundary is not equal to that of the boundary, it is often convenient to prescribe the total flux including the convection. To do this select **Include** in the **Convection** section.

When including the convection, the no flux condition prescribed is



$$-\mathbf{n} \cdot (-D\nabla c + \mathbf{u}c) = 0$$

or

$$-\mathbf{n} \cdot (-D\nabla c - zu_m Fc\nabla\phi + \mathbf{u}c) = 0$$

when migration of ionic species is included.

### *Inflow*

---

Use this node to specify all species concentrations at an inlet boundary.

If you want to specify the concentration of a subset of the partaking species, this can be done by using the [Concentration](#) node instead.

For the **Electroanalysis** interface, this node is available when you select the **Convection** check box on the physics interface **Settings** window.

#### **CONCENTRATION**


For the concentration of each species  $c_{0,c}$  (SI unit: mol/m<sup>3</sup>), enter a value or expression.

#### **BOUNDARY CONDITION TYPE**

The option **Concentration constraint** constrains the concentration values on the boundary by the use of pointwise constraints.

The other option, **Flux (Danckwerts)** can be used when the concentration at the boundary is not known, or when it varies in a nontrivial manner. This may, for example, be useful when reactions with high reaction rates occur in the vicinity of the inlet. In this case the concentration far upstream of the boundary is instead prescribed. The **Flux (Danckwerts)** condition prescribes the total flux defined by the upstream concentration and the fluid velocity at the boundary.

#### **CONSTRAINT SETTINGS**

To display this section, click the **Show More Options** button (  ) and select **Advanced Physics Options**. You can find details about the different constraint settings in the section [Constraint Reaction Terms](#) in the *COMSOL Multiphysics Reference Manual*.

#### **FURTHER READING**

See the theory chapter in the section .

## Outflow

---

Apply this condition at outlets boundaries where species should be transported out of the model domain by fluid motion or by an electric field (in the case of ions). It is assumed that convection and migration (in an electric field) are the dominating transport mechanisms across the boundary, and therefore that the diffusive transport can be ignored, that is:

$$\mathbf{n} \cdot (-D\nabla c) = 0$$

Note that the **Convection** or the **Migration in electric field** transport mechanisms needs to be included for this node to be available.

## Concentration

---


This condition node adds a boundary condition for the species concentration. For example, a  $c = c_0$  condition specifies the concentration of species  $c$ .

### CONCENTRATION

Individually specify the concentration for each species. Select the check box for the **Species** to specify the concentration, and then enter a value or expression in the corresponding field. To use another boundary condition for a specific species, click to clear the check box for the concentration of that species.

When defining conditions on boundaries between different phases, see the section [Prescribing Conditions on Fluid-Solid Interfaces](#).

### CONSTRAINT SETTINGS

To display this section, click the **Show** button (  ) and select **Advanced Physics Options**. You can find details about the different constraint settings in the section [Constraint Reaction Terms](#) in the *COMSOL Multiphysics Reference Manual*.

## Flux

---

This node can be used to specify the species molar flux across a boundary. The flux can for example occur due to chemical reactions or a phase change at the boundary. The flux can also represent the transport to or from a surrounding environment currently not included model.

The prescribed flux of a species  $c$  is by default defined as

$$-\mathbf{n} \cdot (-D\nabla c) = J_0$$

where  $\mathbf{n}$  denotes the outward pointing normal of the boundary. When the mass transport includes migration of ionic species, the flux is defined as:

$$-\mathbf{n} \cdot (-D\nabla c - zu_m Fc \nabla \phi) = J_0$$

The flux prescribed,  $J_0$ , can include any arbitrary user-specified expression. It can be constant or a function of a dependent variable or independent variable. Common examples are a flux dependent of the concentration, temperature, pressure, or the electric potential  $\phi$ .

When defining conditions on boundaries between different phases, see the section [Prescribing Conditions on Fluid-Solid Interfaces](#).

### CONVECTION

By default, the flux due to diffusion and migration in an electric field is prescribed. This is the appropriate flux condition when the relative velocity at the boundary is zero. When the fluid velocity is not equal to that of the boundary, it is often convenient to prescribe the total flux, including the convection. To do this select **Include** in the **Convection** section.

When including the convection, the prescribed flux is defined as:

$$-\mathbf{n} \cdot (-D\nabla c + \mathbf{u}c) = J_0$$

or

$$-\mathbf{n} \cdot (-D\nabla c - zu_m Fc \nabla \phi + \mathbf{u}c) = J_0$$

when migration of ionic species is included.

### INWARD FLUX

Select the Species check box for the species for which to specify the flux, and enter a value or expression for the inward flux in the corresponding field. Use a minus sign when specifying a flux directed out of the system. To use another boundary condition for a specific species, click to clear the check box for that species.

#### *External convection*

Set **Flux type** to **External convection** to prescribe a flux to or from an exterior domain (not modeled) assumed to include convection. The exterior can for example include a

forced convection to control the temperature or to increase the mass transport. In this case the prescribed mass flux corresponds to

$$J_0 = k_c(c_b - c)$$

where  $k_c$  is a mass transfer coefficient and  $c_b$  is the bulk concentration, the typical concentration far into the surrounding exterior domain.

### *Symmetry*

---

The **Symmetry** node can be used to represent boundaries where the species concentration is symmetric, that is, where there is no mass flux across the boundary.

This boundary condition is identical to that of the **No Flux** node.

### *Flux Discontinuity*

---

This node represents a discontinuity in the mass flux across an interior boundary:

where the value  $N_0$  (SI unit:  $\text{mol}/(\text{m}^2 \cdot \text{s})$ ) specifies the jump in total flux at the boundary. This can be used to model a boundary source, for example a surface reaction, adsorption, or desorption.

#### **FLUX DISCONTINUITY**

In this section the jump in species flux (or surface source) is specified.

Select the Species check box for the species to specify and enter a value or expression for the material flux jump in the corresponding field. To use a different boundary condition for a specific species, click to clear the check box for the flux discontinuity of that species.

### *Partition Condition*

---

The **Partition Condition** node can be used to prescribe the relation between the concentration of a solute species in two adjoining immiscible phases. It can for example be used on interior boundaries separating two liquid phases, a gas-liquid interface, or on a boundary separating a liquid phase and a solid or porous media.

Select **Partition coefficient** from the **Relation** list to prescribe a linear relationship between the concentration on either side of a boundary. For a species concentration  $c_i$ , the ratio between the concentration on the upside and on the downside of the

boundary ( $c_{i,u}$  and  $c_{i,d}$  respectively) is then defined in terms of a partition coefficient  $K_i$  in the manner of

$$K_i = \frac{c_{i,u}}{c_{i,d}}$$

Use the associated input field to prescribe the partition coefficient for each species.

Select **User defined** from the **Relation** list to set up a nonlinear relation between the up and downside concentrations. For such relations  $f(c_i)_u = f(c_i)_d$ , where  $f(c_i)_u$  is the expression enforced on the upside. Use the associated input fields to enter the upside and downside expressions for each species.

The up and downside of the selected boundary is indicated with a red arrow in the Graphics window. The arrow points from the downside into the upside. Select the **Reverse direction** check box to reverse the direction of the arrow on the selected boundary, and update the definition of the up and downside concentrations accordingly.



This feature is only available in a limited set of add-on products. See <https://www.comsol.com/products/specifications/> for more details on availability.

#### FURTHER READING

For an example of using a partition condition, see this application example:



*Separation Through Dialysis*: Application Library path  
**Chemical\_Reaction\_Engineering\_Module/Mixing\_and\_Separation/dialysis**

#### *Periodic Condition*

The **Periodic Condition** node can be used to define periodicity for the mass transport between two sets of boundaries. The node prescribes continuity in the concentration and the mass flux between the “source” and the “destination” side, respectively. Note that these names are arbitrary and does not influence the direction in which mass is transported. It is dictated by mass transfer equations in the adjacent domains.


## BOUNDARY SELECTION

The node can be activated on more than two boundaries, in which case the feature tries to identify two separate surfaces that each consist of one or several connected boundaries.


## DESTINATION SELECTION

For more complex geometries, it might be necessary to specify the destination selection manually. To do so, right-click the **Periodic Condition** node and choose **Manual Destination Selection**. You can then specify the boundaries that constitute the destination surfaces in the **Destination Selection** section.

## CONSTRAINT SETTINGS

To display this section, click the **Show More Options** button (  ) and select **Advanced Physics Options** in the **Show More Options** dialog box. This section contains settings for specifying the type of constraint and whether to use a pointwise or weak constraint. See [Constraint Settings](#) in the *COMSOL Multiphysics Reference Manual* for more information.


## ORIENTATION OF SOURCE

To display this section, click the **Show More Options** button (  ) and select **Advanced Physics Options** in the **Show More Options** dialog box. For information about the **Orientation of Source** section, see [Orientation of Source and Destination](#) in the *COMSOL Multiphysics Reference Manual*.

## ORIENTATION OF DESTINATION

This section appears if the setting for **Transform to intermediate map** in the **Orientation of Source** section is changed from the default value, **Automatic**, and **Advanced Physics Options** is selected in the **Show More Options** dialog box. For information about the **Orientation of Destination** section, see [Orientation of Source and Destination](#) in the *COMSOL Multiphysics Reference Manual*.

## MAPPING BETWEEN SOURCE AND DESTINATION

To display this section, click the **Show More Options** button (  ) and select **Advanced Physics Options** in the **Show More Options** dialog box. For information about the **Mapping Between Source and Destination** section, see [Mapping Between Source and Destination](#) in the *COMSOL Multiphysics Reference Manual*.

## FURTHER READING

For an example of using a periodic condition, see this application example:



*The KdV Equation and Solitons*: Application Library path  
**COMSOL\_Multiphysics/Equation\_Based/kdv\_equation**

## Line Mass Source

The **Line Mass Source** feature models mass flow originating from a tube or line region with an infinitely small radius.



This feature is only available in a limited set of add-on products. See <https://www.comsol.com/products/specifications/> for more details on availability.

## SELECTION

The **Line Mass Source** feature is available for all dimensions, but the applicable selection differs between the dimensions.

MODEL DIMENSION	APPLICABLE GEOMETRICAL ENTITY
2D	Points
2D Axisymmetry	Points not on the symmetry axis and the symmetry axis
3D	Edges

## SPECIES SOURCE

Enter the source strength,  $\dot{q}_{l,c}$ , for each species (SI unit: mol/(m·s)). A positive value results in species injection from the line into the computational domain, and a negative value means that the species is removed from the computational domain.

Line sources located on a boundary affect the adjacent computational domains. This effect makes the physical strength of a line source located in a symmetry plane twice the given strength.

## Point Mass Source

---

The **Point Mass Source** feature models mass flow originating from an infinitely small domain around a point.



This feature is only available in a limited set of add-on products. See <https://www.comsol.com/products/specifications/> for more details on availability.

### SPECIES SOURCE

Enter the source strength,  $\dot{q}_{p,c}$ , for each species (SI unit: mol/s). A positive value results in species injection from the point into the computational domain, and a negative value means that the species is removed from the computational domain.

Point sources located on a boundary or on an edge affect the adjacent computational domains. This has the effect, for example, that the physical strength of a point source located in a symmetry plane is twice the given strength.

## Open Boundary

---

Use this node to set up mass transport across boundaries where both convective inflow and outflow can occur. On the parts of the boundary where fluid flows into the domain, an exterior species concentration is prescribed. On the remaining parts, where fluid flows out of the domain, a condition equivalent to the **Outflow** node is instead prescribed.

The direction of the flow across the boundary is typically calculated by a fluid flow interface and is provided as a model input to the Transport of Diluted Species interface.



This feature is only available in a limited set of add-on products. See <https://www.comsol.com/products/specifications/> for more details on availability.

### EXTERIOR CONCENTRATION

Enter a value or expression for the **Exterior concentration**.



### *Thin Diffusion Barrier*

---

Use this boundary condition to model a thin layer through which mass is transported by diffusion only. The node is applicable on interior boundaries and can be used to avoid meshing thin structures.

#### **THIN DIFFUSION BARRIER**

Specify the **Layer thickness**,  $d_s$ , and input a **Diffusion coefficient**,  $D_{s,c}$ , for each of the species included.

### *Thin Impermeable Barrier*

---

This feature models a thin mass transfer barrier. It is available on interior boundaries and introduces a discontinuity in the concentration across the boundary. On each side, a no-flux condition is prescribed for the mass transport implying that it acts as a barrier. The feature can be used to avoid meshing thin structures.

Solving a model involving coupled fluid flow and mass transfer, the Thin Impermeable Barrier feature can be combined with an Interior Wall feature in order to model a thin solid wall.

### *Equilibrium Reaction*

---

Use this node to model a reaction where the kinetics is so fast that the equilibrium condition is fulfilled at all times. The node solves for an additional degree of freedom (the reaction rate  $R_{eq}$ ) to fulfill the equilibrium condition at all times in all space coordinates.

If the **Apply equilibrium condition on inflow boundaries** check box is selected, the specified inflow concentration values in all active **Inflow** boundary nodes for the physics interface are modified to comply with the equilibrium condition.



- A necessary requirement for this feature to be available is that two or more species are solved for by the interface.
  - This feature is only available in a limited set of add-on products. See <https://www.comsol.com/products/specifications/> for more details on availability.
-

## EQUILIBRIUM CONDITION

The list defaults to **Equilibrium constant** or select **User defined**. For either option, the **Apply equilibrium condition on inflow boundaries** check box is selected by default.

For **Equilibrium constant**, enter an **Equilibrium constant**  $K_{\text{eq}}$  (dimensionless). Also enter a value or expression for the **Unit activity concentration**  $C_{a0}$  (SI unit:  $\text{mol}/\text{m}^3$ ).

Selecting **Equilibrium constant** defines an equilibrium condition based on the stoichiometric coefficients, the species activities, and the law of mass action.

For **User defined**, enter an **Equilibrium expression**  $E_{\text{eq}}$  (dimensionless).

## STOICHIOMETRIC COEFFICIENTS

Enter a value for the stoichiometric coefficient  $\nu_c$  (dimensionless). The default is 0. Use negative values for reactants and positive values for products in the modeled reaction.

Species with a stoichiometric coefficient value of 0 are not affected by the **Equilibrium Reaction** node.

## Surface Reactions

---

The **Surface Reactions** node can be used to account for the species boundary flux due to chemical reactions occurring on a surface (heterogeneous reactions). For a domain species participating in a surface reaction, the boundary flux corresponds to the reaction rate at the surface.

## SURFACE REACTION RATE

Specify the surface reaction rate  $J_0$  of each species resulting from the reactions. Note that if you have the Chemistry interface available, provided with the Chemical Reaction Engineering Module, the reaction rate expressions can be automatically generated and picked up using the drop-down list.

When defining conditions on boundaries between different phases, see the section [Prescribing Conditions on Fluid-Solid Interfaces](#).

## FURTHER READING

For an example of using the Surface Reactions node, see this application example:



*Chemical Vapor Deposition of GaAs*: Application Library path  
**Chemical\_Reaction\_Engineering\_Module/Reactors\_with\_Mass\_and\_Heat\_Transfer/gaas\_cvd**

---

## *Surface Equilibrium Reaction*

---

Use this node to model an equilibrium reaction on a boundary (surface). The settings for this node are similar to [Equilibrium Reaction](#). Note that a necessary requirement for this is feature to be available is that two or more species are solved for by the interface.

When defining conditions on boundaries between different phases, see the section [Prescribing Conditions on Fluid-Solid Interfaces](#).



This feature is only available in a limited set of add-on products. See <https://www.comsol.com/products/specifications/> for more details on availability.

## *Fast Irreversible Surface Reaction*

---

This boundary node defines an irreversible reaction where the kinetics is so fast that the only factor limiting the reaction rate is the transport of a species to the reacting surface.

The node will set the **Rate limiting species concentration to zero** at the boundary, and balance the fluxes of the species participating in the reaction and the current densities according to the Stoichiometric Coefficients settings.

When defining conditions on boundaries between different phases, see the section [Prescribing Conditions on Fluid-Solid Interfaces](#).



This feature is only available in a limited set of add-on products. See <https://www.comsol.com/products/specifications/> for more details on availability.

## *Porous Electrode Coupling*

---

Use this node to add a molar source in a domain that is coupled to one or multiple nodes of an Electrochemistry Interface.

The molar source is calculated from the number of electrons, stoichiometric coefficients, and volumetric current densities of the coupled porous electrode reactions specified in the **Reaction Coefficients** subnodes.

In the Transport of Concentrated Species interface, the molar sources (or sinks) are multiplied by the species molar masses to obtain the corresponding mass sources.

Additional [Reaction Coefficients](#) subnodes are available from the context menu (right-click the parent node) as well as from the **Physics** toolbar, **Attributes** menu.

Note that if you are also modeling the momentum transport and expect a nonnegligible total mass source or sink, which is often the case in gas diffusion electrodes, you need to also add a corresponding Porous Electrode Coupling node in the Fluid Flow interface.



This feature is only available in a limited set of add-on products. See <https://www.comsol.com/products/specifications/> for more details on availability.

---

### *Reaction Coefficients*

---

Add this node to the [Electrode Surface Coupling](#) and [Porous Electrode Coupling](#) features to define molar fluxes and sources based on electrode current densities in an **Electrochemistry** interface.

The molar flux or source is proportional to the stoichiometric coefficients and the current density according to Faraday's law.

Current densities from **Electrode Reaction** ( $i_{loc}$ , SI unit:  $A/m^2$ ) or **Porous Electrode Reaction** nodes ( $i_v$ , SI unit:  $A/m^3$ ) of any **Electrochemistry** interface in the model are available for selection as the **Coupled reaction**, and user-defined expressions are also supported.

Enter the **Number of participating electrons**  $n_m$  (dimensionless) and the **Stoichiometric coefficient**  $v_c$  (dimensionless) as explained in the theory section linked below.

Use multiple subnodes to couple to multiple reactions.

---

### *Electrode Surface Coupling*

---

Use this node to define a flux boundary condition based on current densities of one or multiple nodes in an Electrochemistry interface.

The flux is proportional to the current densities and the stoichiometric coefficients according to Faraday's law as defined by summation over the [Reaction Coefficients](#) subnodes.

Note that if you are also modeling the momentum transport and expect a nonnegligible total mass flux over the boundary, which is often the case for gas diffusion electrodes, you need to also add a corresponding Electrode Surface Coupling node in the Fluid Flow interface.



This feature is only available in a limited set of add-on products. See <https://www.comsol.com/products/specifications/> for more details on availability.

---

### *Porous Medium*

---

Use this node to model the concentration of diluted species transported by a solvent (mobile fluid phase) through interstices in a solid porous medium. In addition to transport due to convection and diffusion, the node contains functionality to include species evolution through adsorption and dispersion.



This feature is only available in a limited set of add-on products. See <https://www.comsol.com/products/specifications/> for more details on availability.

Species transport through a porous medium is affected both by properties of the fluid phase, and as well as properties of the solid matrix. These properties are defined in the [Fluid](#) and [Porous Matrix](#) subnodes respectively. The Porous Medium node supports material properties using a [Porous Material](#) node (described in the *COMSOL Multiphysics Reference Manual*), where properties are defined per phase in a similar manner.

Add an [Adsorption](#) or a [Dispersion](#) subfeature to the **Porous Medium** to account for the corresponding transport mechanism.

---

### *Fluid*

---

Use this node to specify the mass transfer in the mobile fluid solvent filling the pores of a porous medium. It is used as a subnode to [Porous Medium](#).

#### **MODEL INPUTS**

The temperature model input is always available. By default, the **Temperature** model input is set to **Common model input**, and the temperature is controlled from [Default](#)

**Model Inputs** under **Global Definitions** or by a locally defined **Model Input**. If a Heat Transfer interface is included in the component, it controls the temperature **Common model input**. Alternatively, the temperature field can be selected from another physics interface. All physics interfaces have their own tags (**Name**). For example, if a Heat Transfer in Fluids interface is included in the component, the **Temperature (ht)** option is available for  $T$ .

You can also select **User defined** from the **Temperature** model input in order to manually prescribe  $T$ .

## CONVECTION

If transport by convection is active, the velocity field of the solvent needs to be specified. For **User defined**, enter values or expressions for the velocity components in the input fields. This input option is always available. You can also select the velocity field solved for by a Fluid Flow interface added to the model component. These physics interfaces are available for selection in the **Velocity field** list.

## DIFFUSION

Use this section to specify diffusion coefficients describing the diffusion of each species in the solvent fluid.

Use the **Source** list to select to pick up diffusion coefficients defined in a material or a Chemistry interface.

Select **Material**, and a **Fluid material** to use a diffusion coefficient in a material available in the model. User-defined property groups including a diffusion coefficient output property are available for selection in the **Fluid diffusion coefficient** list.

Select **Chemistry** to use a diffusion coefficient from a Chemistry interface added to model component. Diffusion coefficients are automatically defined when **Calculate transport properties** is selected on the interface level of a Chemistry interface. All defined diffusion coefficients are available for selection in the **Fluid diffusion coefficient** list.

For **User defined**, enter a value or expression for the **Fluid diffusion coefficient**  $D_{F,i}$  of each species in the corresponding input field.

In a porous medium the diffusivity is reduced due to the fact that the solid grains impede Brownian motion. Select an **Effective diffusivity model** to account for the reduced diffusivity. The available models are **Millington and Quirk model** (the default), **Bruggeman model**, **Tortuosity model**, or **No correction**. For **Tortuosity model**, enter a value for the tortuosity  $\tau_{F,i}$  (dimensionless).

Note that multiple species, as well as Migration in Electric fields (described below) is only available for certain COMSOL Multiphysics add-on products. See details: <https://www.comsol.com/products/specifications/>.

## MIGRATION IN ELECTRIC FIELD

This section is available when the **Migration in electric field** check box is selected. From the **Electric potential** list, select the source of the electric field.

- For **User defined**, enter a value or expression for the **Electric potential**  $V$ . This input option is always available.
- Select the electric potential solved by an AC/DC-based interface that has added to the component.
- Select the electric potential defined or solved for by an Electrochemistry interface added to the component.

By default the **Mobility** is set to be calculated based on the species effective diffusivity and the temperature using the **Nernst-Einstein relation**. For **User defined**, select the appropriate scalar or tensor type — **Isotropic**, **Diagonal**, **Symmetric**, or **Full** — and type in the value or expression of the effective mobility  $u_{me,i}$ .

Enter the **Charge number**  $z_c$  for each species.

## *Porous Matrix*

---

This node sets the porosity when modeling transport of diluted species in a **Porous Medium** or a **Unsaturated Porous Medium**.

The default **Porosity**  $\epsilon_p$  of the solid matrix is taken **From material**. The Porous Matrix node supports the use of a **Porous Material** node, where the porosity is defined in the manner of

$$\epsilon_p = 1 - \sum_i \theta_{s,i} - \sum_i \theta_{imf,i}$$

where  $\theta_{s,i}$  and  $\theta_{imf,i}$  are the porosities of the **Solid** and **Immobile Fluids** subnodes under the **Porous Material** node.

Select **From pellet bed densities** to compute the porosity using the (dry bulk) **Bed density**  $\rho_b$  and the (single phase) **Pellet density**  $\rho_{pe}$ . The porosity is then defined from

$$\varepsilon_p = 1 - \frac{\rho_b}{\rho_{pe}}$$

For **User defined**, enter a value or expression for the porosity.

### *Dispersion*

---

Local variations in the velocity as the fluid flows around solid particles lead to mechanical mixing, referred to as dispersion. Use this feature to account for dispersion in a [Porous Medium](#) or a [Unsaturated Porous Medium](#)

This subfeature is available when both the **Mass transfer in porous media mass transfer** check box and the **Convection** check box are selected in the **Settings** window for the physics interface.

Select the **Specify dispersion for each species individually** check box to specify the dispersion tensor  $D_D$  (SI unit:  $\text{m}^2/\text{s}$ ) for each species separately. When not selected the same dispersion tensor  $D_D$  is used for all species.

Select an option from the **Dispersion tensor** list — **Dispersivity** or **User defined**.

Select **Dispersivity** to specify the dispersion in terms of dispersivities (SI unit: m). Select an option from the **Dispersivity model** list: **Isotropic** (the default) or **Transverse isotropic** based on the properties of the porous media. For isotropic porous media, specify the longitudinal and transverse dispersivities. For transverse isotropic porous media, specify the longitudinal, horizontal transverse, and vertical transverse dispersivities.

For **User defined**, specify the dispersion components in terms of constants or expressions. Select **Isotropic**, **Diagonal**, **Symmetric**, or **Full** to enable the appropriate tensor components.

### *Unsaturated Porous Medium*

---

Use this node to model the concentration of diluted species transported by a solvent (mobile fluid phase) through interstices in a solid porous medium. In addition to transport due to convection and diffusion, the node contains functionality to include species evolution through adsorption and dispersion.

Use this node to model the concentration of diluted species transported by a liquid (mobile fluid phase) in a partially filled solid porous medium. The interstices of the porous medium contains the liquid carrier phase and gas pockets. Apart from



convection and diffusion, the node contains functionality to include species evolution through adsorption, dispersion, and volatilization.



This feature is only available in a limited set of add-on products. See <https://www.comsol.com/products/specifications/> for more details on availability.

The properties of each phase present are defined using the [Liquid](#), the [Gas](#), and the [Porous Matrix](#) subnodes respectively. The Unsaturated Porous Medium node supports material properties using a [Porous Material](#) node (described in the *COMSOL Multiphysics Reference Manual*), where properties are defined per phase in a similar manner.

Add an [Adsorption](#) or a [Dispersion](#) subfeature to the **Unsaturated Porous Medium** to account for the corresponding transport mechanism.

## *Liquid*

Use this node to specify the mass transfer in the mobile liquid solvent present in the pores of the [Unsaturated Porous Medium](#).

### MODEL INPUTS

The temperature model input is always available. By default, the **Temperature** model input is set to **Common model input**, and the temperature is controlled from [Default Model Inputs](#) under **Global Definitions** or by a locally defined [Model Input](#). If a Heat Transfer interface is included in the component, it controls the temperature **Common model input**. Alternatively, the temperature field can be selected from another physics interface. All physics interfaces have their own tags (**Name**). For example, if a Heat Transfer in Fluids interface is included in the component, the **Temperature (ht)** option is available for  $T$ .

You can also select **User defined** from the **Temperature** model input in order to manually prescribe  $T$ .

### SATURATION

Select **Saturation** or **Liquid volume fraction** from the list.

For **Saturation**, enter a value for  $s$  (dimensionless) between 0 and 1. The liquid volume fraction is then computed from the saturation and porosity as  $\theta_l = \varepsilon_p s$ .

For **Liquid volume fraction**, enter a value for  $\theta_l$  (dimensionless) between 0 and the value of the porosity.

Select a **Liquid fraction time change**: **Liquid fraction constant in time** (the default), **Time change in liquid fraction**, or **Time change in pressure head**.

- For **Time change in fluid fraction**, enter  $d\theta/dt$  (SI unit: 1/s).
- For **Time change in pressure head**, enter  $dH_p/dt$  (SI unit: m/s) and a **Specific moisture capacity**  $C_m$  (SI unit: 1/m). If a Darcy's Law interface is included in the component, the time change in pressure head solved for can be selected.

## CONVECTION

If transport by convection is active, the velocity field of the solvent needs to be specified. For **User defined**, enter values or expressions for the velocity components in the input fields. This input option is always available. You can also select the velocity field solved for by a Fluid Flow interface added to the model component. These physics interfaces are available for selection in the **Velocity field** list.

## DIFFUSION

Use this section to specify diffusion coefficients describing the diffusion of each species in the solvent liquid.

Use the **Source** list to select to pick up diffusion coefficients defined in a material or a Chemistry interface.

Select **Material**, and a **Liquid material** to use a diffusion coefficient in a material available in the model. User-defined property groups including a diffusion coefficient output property are available for selection in the **Liquid diffusion coefficient** list.

Select **Chemistry** to use a diffusion coefficient from a Chemistry interface added to model component. Diffusion coefficients are automatically defined when **Calculate transport properties** is selected on the interface level of a Chemistry interface. All defined diffusion coefficients are available for selection in the **Liquid diffusion coefficient** list.

For **User defined**, enter a value or expression for the **Liquid diffusion coefficient**  $D_{L,i}$  of each species in the corresponding input field.

In a porous medium the diffusivity is reduced due to the fact that the solid grains impede Brownian motion. Select an **Effective diffusivity model, liquid** to account for the reduced diffusivity in the liquid. The available models are **Millington and Quirk model**

(the default), **Bruggeman model**, **Tortuosity model**, or **No correction**. For **Tortuosity model**, enter a value for the tortuosity  $\tau_{L,i}$  (dimensionless).

Note that multiple species, as well as Migration in Electric fields (described below) is only available for certain COMSOL Multiphysics add-on products. For details see: <https://www.comsol.com/products/specifications/>.

#### MIGRATION IN ELECTRIC FIELD

This section is available when the **Migration in electric field** check box is selected. From the **Electric potential** list, select the source of the electric field.

- For **User defined**, enter a value or expression for the **Electric potential**  $V$ . This input option is always available.
- Select the electric potential solved by an AC/DC-based interface that has added to the component.
- Select the electric potential defined or solved for by an Electrochemistry interface added to the component.

By default the **Mobility** is set to be calculated based on the species effective diffusivity and the temperature using the **Nernst-Einstein relation**. For **User defined**, select the appropriate scalar or tensor type — **Isotropic**, **Diagonal**, **Symmetric**, or **Full** — and type in the value or expression of the effective mobility  $u_{me,i}$ .

Enter the **Charge number**  $z_c$  for each species.

#### *Gas*

---

Use this node to specify the mass transfer in the gas phase present in the pores of the [Unsaturated Porous Medium](#).

#### DIFFUSION

The settings for the diffusion of each species in the gas phase are identical to those in the [Liquid](#) subnode.

#### VOLATILIZATION

Enter a value for the volatilization isotherm  $k_{G,c}$  (dimensionless) for each species.

## Adsorption

Use this node to model adsorption of the (fluid phase) solute species onto the surface of the porous matrix. It is available as a subnode to the **Porous Medium** and the **Unsaturated Porous Medium** nodes.



This feature is only available in a limited set of add-on products. See <https://www.comsol.com/products/specifications/> for more details on availability.

### MATRIX PROPERTIES

The density of the porous media is needed when modeling adsorption to the surface of the porous matrix. Choose to input either the **Dry bulk density**  $\rho$ , or the **Solid phase density**  $\rho_s$ . The former is the density of the porous matrix including empty pores, while the latter corresponds to the density of the pure solid phase. The density can be defined from the domain material by selecting **From material**, or from a user defined expression. When a **Porous Material** is used on the selection, the density will be requested from a **Solid** subfeature. Several **Solid** features can be used to model a homogeneous mixture of several solid components.

### ADSORPTION

Select an **Adsorption isotherm** — **Langmuir**, **Freundlich**, **Toth**, **BET**, or **User defined** to specify how to compute  $c_p$ , the amount of species sorbed to the solid phase (moles per unit dry weight of the solid):

- For **Langmuir**:

$$c_p = c_{pmax} \frac{K_L c}{1 + K_L c} \quad , \quad K_P = \frac{\partial c_p}{\partial c} = \frac{K_L c_{pmax}}{(1 + K_L c)^2}$$

Enter a **Langmuir constant**  $k_{L,c}$  (SI unit:  $\text{m}^3/\text{mol}$ ) and an **Adsorption maximum**  $c_{p,max,c}$  (SI unit:  $\text{mol}/\text{kg}$ ).

- For **Freundlich**:

$$c_P = K_F \left( \frac{c}{c_{\text{ref}}} \right)^N, \quad K_P = \frac{\partial c_P}{\partial c} = N \frac{c_P}{c}$$

Enter a **Freundlich constant**  $k_{F,c}$  (SI unit: mol/kg), a **Freundlich exponent**  $N_{F,c}$  (dimensionless), and a **Reference concentration**  $c_{\text{ref},c}$  (SI unit: mol/m<sup>3</sup>).

- For **Toth**:

$$c_P = c_{P\text{max}} \frac{b_T c}{(1 + (b_T c)^{N_T})^{1/N_T}}$$

and

$$K_P = \frac{\partial c_P}{\partial c} = c_{P\text{max}} b_T (1 + (b_T c)^{N_T})^{-\left(1 + \frac{1}{N_T}\right)}$$

Enter a **Toth constant**  $b_{T,c}$  (SI unit: m<sup>3</sup>/mol), a **Toth exponent**  $N_{T,c}$  (dimensionless), and an **Adsorption maximum**  $c_{P,\text{max},c}$  (SI unit: mol/kg).

- For **BET (Brunauer-Emmett-Teller)**:

$$c_P = \frac{K_B c_0 c}{(c_S - c) \left( 1 + (K_B - 1) \frac{c}{c_S} \right)}$$

and

$$K_P = \frac{\partial c_P}{\partial c} = \frac{K_B c_0 c_S ((K_B - 1) c^2 + c_S^2)}{(c - c_S)^2 (c_S + (K_B - 1) c)^2}$$

Enter a **BET constant**  $K_{B,c}$  (dimensionless), a **Monolayer adsorption capacity**  $c_{0,c}$  (SI unit: mol/kg), and an **Saturation concentration**  $c_{S,c}$  (SI unit: mol/m<sup>3</sup>).

- For **User defined** enter an **Adsorption isotherm**  $c_{P,c}$  (SI unit: mol/kg):

$$c_P = f(c)$$

For more information, see in the theory section.

## FURTHER READING

See the theory chapter in the section .

## Volatilization

---

This feature is available when the **Mass transfer in porous media** check box is selected in the **Settings** window for the physics interface.

Use this feature to model mass transfer at the boundary due to volatilization. The species dissolved in the liquid are assumed to be vaporized at the boundary, and transported into the surrounding bulk region due to convection and diffusion. The mass transfer at the boundary is defined as

$$-\mathbf{n} \cdot \mathbf{J}_c = -h_c(k_{G,c}c - c_{\text{Gatm},c})$$

where  $h_c$  is the mass transfer coefficient,  $k_{G,c}$  the volatilization coefficient, and  $c_{\text{Gatm},c}$  the concentration in the surrounding atmosphere.



This feature is only available in a limited set of add-on products. See <https://www.comsol.com/products/specifications/> for more details on availability.

---

### VOLATILIZATION

Enter a **Mass transfer coefficient**  $h_c$  defining the transfer into the surrounding media. This can be given by boundary layer theory. When assuming that no convective flow is present in the surrounding, the mass transfer coefficient can be defined from the gas diffusion coefficient  $D_{Gc}$  and the thickness of the diffusion layer  $d_s$  in the manner of

$$h_c = \frac{D_{Gc}}{d_s}$$

Also give the atmospheric concentration for each species,  $c_{\text{Gatm},c}$ . The **Volatilization** coefficient  $k_{G,c}$  for each species are taken from the adjacent **Unsaturated Porous Medium** domain.

## Species Source

---

In order to account for consumption or production of species in porous domains, the **Species Source** node adds source terms expressions  $S_i$  to the right-hand side of the species transport equations.

## DOMAIN SELECTION

From the **Selection** list, choose the domains on which to define rate expression or expressions that govern the source term in the transport equations.


If there are several types of domains, with subsequent and different reactions occurring within them, it might be necessary to remove some domains from the selection. These are then defined in an additional **Species Source** node.

## SPECIES SOURCE

Add a source term  $S_i$  (SI unit:  $\text{mol}/(\text{m}^3 \cdot \text{s})$ ) for each of the species solved for. Enter a value or expression in the field of the corresponding species.

## *Hygroscopic Swelling*

---

The **Hygroscopic Swelling** multiphysics coupling node (  ) is used for moisture concentration coupling between the Solid Mechanics interface and either the Transport of Diluted Species or Transport of Diluted Species in Porous Media interfaces.

Hygroscopic swelling is an effect of internal strain caused by changes in moisture content. This volumetric strain can be written as

$$\varepsilon_{\text{hs}} = \beta_{\text{h}} M_{\text{m}} (c_{\text{mo}} - c_{\text{mo,ref}})$$

where  $\beta_{\text{h}}$  is the coefficient of hygroscopic swelling,  $M_{\text{m}}$  is the molar mass,  $c_{\text{mo}}$  is the moisture concentration, and  $c_{\text{mo,ref}}$  is the strain-free reference concentration.

This feature requires a license of either the MEMS Module or the Structural Mechanics Module. The multiphysics feature will appear automatically if both the Transport of Diluted Species and the Solid Mechanics interfaces are added to the same component. For the most current information about licensing, please see <https://www.comsol.com/products/specifications/>.

## FURTHER READING

More information about how to use hygroscopic swelling can be found in [Hygroscopic Swelling Coupling](#) section in the *Structural Mechanics Module User's Guide*.

More information about multiphysics coupling nodes can be found in the section [The Multiphysics Branch](#).

## Out-of-Plane Flux

The out-of-plane flux node is used to prescribe a flux in an out-of-plane domain, and it can be used to reduce a model geometry to 2D, or even 1D, when the concentration variation is small in one or more directions. This could be the case for example when the object to model is thin or slender. Figure 7-1 shows examples of possible situations in which this type of geometry reduction can be applied.

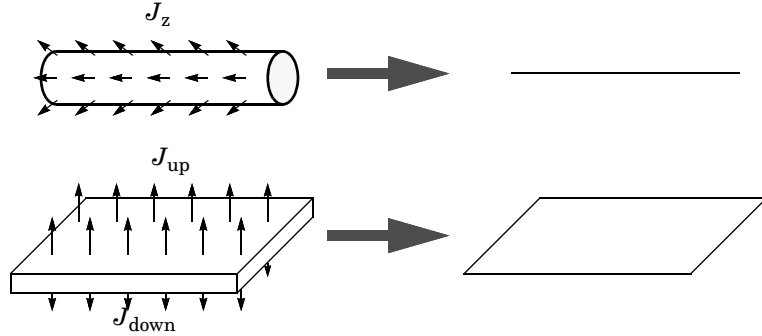


Figure 7-1: Geometry reduction from 3D to 1D (top) and from 3D to 2D (bottom).

For a 1D component this node adds a single out-of-plane molar flux  $J_{0,z,i}$  for species  $i$ . For a 2D component two fluxes can be prescribed for each species;  $J_{0,u,i}$  for the upside of the domain, and  $J_{0,d,i}$  for the downside of the domain.

The fluxes are added as a source contribution to the right-hand side of the transport equation. For example, in 2D components with diffusion, convection, and out-of-plane flux present, the mass transfer of species  $i$  is defined by the mass conservation equation

$$\frac{\partial c_i}{\partial t} + \nabla \cdot \mathbf{J}_i + \mathbf{u} \cdot \nabla c_i = R_i + S_{\text{opf},i}, \quad (7-1)$$

where  $S_{\text{opf},i}$  is the out-of-plane source for species  $i$

$$S_{\text{opf},i} = \frac{J_{0,i}}{d_z} = \frac{J_{0,u,i} + J_{0,d,i}}{d_z}.$$

For external convection on the upside and the downside of the domain, the out-of-plane flux is



$$J_{0,i} = k_{c,u,i}(c_{b,u,i} - c_i) + k_{c,d,i}(c_{b,d,i} - c_i)$$

where  $k_c$  is a mass transfer coefficient and  $c_b$  is the bulk concentration, the typical concentration far into the surrounding exterior domain. The prescribed flux,  $J_0$ , can include any arbitrary user-specified expressions. It can be a constant or a function of a dependent variable or independent variables.

#### UPSIDE INWARD FLUX

For 1D components, enter the **cross-sectional perimeter**  $P_c$  to get the out-of-plane flux

$$J_{0,i} = P_c J_{0,z,i}.$$

The default value of  $P_c$  is the circumference. Either keep the default value, for a circular cross-section shape, or edit the value to get a user-defined shape of the out-of-plane cross-section.

The available flux type options are **General inward flux** and **External convection**. Select the **Species** check box for the species for which to specify the flux, and enter a value or expression for the inward flux in the corresponding field. Use a minus sign when specifying a flux directed out of the system. To use another boundary condition for a specific species, click to clear the check box for that species.

Set **Flux type** to **External convection** to prescribe a flux to or from an exterior domain (not modeled) assumed to include convection. The exterior can for example include a forced convection to control the temperature or to increase the mass transport. In this case the prescribed mass flux corresponds to

$$J_0 = k_c(c_b - c)$$

where  $k_c$  is a mass transfer coefficient and  $c_b$  is the bulk concentration, the typical concentration far into the surrounding exterior domain.

#### DOWNSIDE INWARD FLUX

The available options are **General inward flux** and **External convection**. The settings are the same as for the [Upside Inward Flux](#) section.

## FURTHER READING

For an example of using the Out-of-Plane Flux node, see this application example:



*Mass Transfer from a Thin Domain:* Application Library path  
**Chemical\_Reaction\_Engineering\_Module/Tutorials/thin\_domain**

## Fracture

Use this node to model mass transport along thin fractures in porous media. The node assumes that the transport in the tangential direction along the fracture is dominant as a result of lower flow resistance.

The Fracture feature has two default subfeatures — the **Fluid (Fracture)** subfeature where convection and diffusion are specified and the **Fracture Material** subfeature where the porosity  $\varepsilon_p$  is defined.



This feature is only available in a limited set of add-on products. See <https://www.comsol.com/products/specifications/> for more details on availability.

## FRACTURE PROPERTIES

Specify a value for the **Fracture thickness**  $d_{\text{fr}}$ .

For the **Fracture** boundary condition, there are a number of **Attributes** available which can be accessed either via the **Physics** Ribbon toolbar (for Windows) or by right-clicking on the **Fracture** node. These attributes are identical to the **Boundary, Edge, Point, and Pair Nodes for the Transport of Diluted Species in Fractures Interface** described, for example, in the *Chemical Reaction Engineering Module User's Guide*.

## Fluid (Fracture)

Use this node to specify the mass transfer in the mobile fluid solvent filling the pores of a fracture.

It uses the same input options as the **Fluid** subfeature under the **Porous Medium** feature. There are further details about the settings.

## MODEL INPUTS

The temperature model input is always available. By default, the **Temperature** model input is set to **Common model input**, and the temperature is controlled from **Default Model Inputs** under **Global Definitions** or by a locally defined **Model Input**. If a Heat Transfer interface is included in the component, it controls the temperature **Common model input**. Alternatively, the temperature field can be selected from another physics interface. All physics interfaces have their own tags (**Name**). For example, if a Heat Transfer in Fluids interface is included in the component, the **Temperature (ht)** option is available for T.

## CONVECTION

If transport by convection is active, the velocity field of the solvent needs to be specified. Select an option from the **Velocity field** list to specify the convective velocity along the fracture. For a consistent model, use a Fracture feature in a Darcy's Law interface to compute the fluid flow velocity in the fracture.

For **User defined**, enter values or expressions for the velocity components in the table shown

## DIFFUSION

Use this section to specify diffusion coefficients describing the diffusion of each species in the solvent fluid.



The **Diffusion** settings are similar to those for the diffusion coefficients in the **Fluid** node under **Porous Medium**.

## Fracture Material

Specify the **Porosity**,  $\varepsilon_p$  (dimensionless) of the porous matrix. This is by default taken **From material**. Select **From pellet bed densities** to compute the porosity from the (dry bulk) **Bed density**  $\rho_b$  and the (single phase) **Pellet density**  $\rho_{pe}$  through the relation

$$\varepsilon_p = 1 - \frac{\rho_b}{\rho_{pe}}$$

For **User defined**, enter a value or expression for the porosity.

# Theory for the Transport of Diluted Species Interface

The [Transport of Diluted Species Interface](#) provides a predefined modeling environment for studying the evolution of chemical species transported by diffusion and convection. The physics interface assumes that all species present are dilute; that is, that their concentration is low compared to a solvent fluid or solid. As a rule of thumb, a mixture containing several species can be considered dilute when the concentration of the solvent is more than 90 mol%. Due to the dilution, mixture properties such as density and viscosity can be assumed to correspond to those of the solvent.

When studying mixtures that are not dilute, the mixture and transport properties depend on the composition, and a different physics interface is recommended. See [The Transport of Concentrated Species Interface](#) for more information.

*Fick's law* governs the diffusion of the solutes, dilute mixtures, or solutions, while the phenomenon of ionic migration is sometimes referred to as *electrokinetic flow*. The Transport of Diluted Species interface supports the simulations of chemical species transport by convection, migration, and diffusion in 1D, 2D, and 3D as well as for axisymmetric components in 1D and 2D.

In this section:

- [Adding Transport Through Migration](#)
- [Convective Term Formulation](#)
- [Crosswind Diffusion](#)
- [Danckwerts Inflow Boundary Condition](#)
- [Equilibrium Reaction Theory](#)
- [Mass Balance Equation](#)
- [Mass Sources for Species Transport](#)
- [Solving a Diffusion Equation Only](#)
- [Supporting Electrolytes](#)
- [References](#)

Note: Some features explained in this section require certain add-on modules. For details see <https://www.comsol.com/products/specifications/>

The section also includes the theory for :

- Adsorption
- Convection in Porous Media
- Diffusion in Porous Media
- Dispersion
- Mass Balance Equation for Transport of Diluted Species in Porous Media
- Mass Transport in Fractures
- Reactions

### *Mass Balance Equation*

---

The default node attributed to the Transport of Diluted Species interface models chemical species transport through diffusion and convection and solves the mass conservation equation for one or more chemical species  $i$ :

$$\frac{\partial c_i}{\partial t} + \nabla \cdot \mathbf{J}_i + \mathbf{u} \cdot \nabla c_i = R_i \quad (7-2)$$

Equation 7-2 in its form above includes the transport mechanisms diffusion and convection. If *Migration in Electric Field* is activated (only available in some add-on products), the migration mechanism will be added to the equation as well. See more details in the section [Adding Transport Through Migration](#).

- $c_i$  is the concentration of the species (SI unit: mol/m<sup>3</sup>)
- $D_i$  denotes the diffusion coefficient (SI unit: m<sup>2</sup>/s)
- $R_i$  is a reaction rate expression for the species (SI unit: mol/(m<sup>3</sup>·s))
- $\mathbf{u}$  is the mass averaged velocity vector (SI unit: m/s)
- $\mathbf{J}_i$  is the mass flux diffusive flux vector (SI unit: mol/(m<sup>2</sup>·s))

The mass flux relative to the mass averaged velocity,  $\mathbf{J}_i$  (SI unit: mol/(m<sup>2</sup>·s)), is associated with the mass balance equation above and used in boundary conditions and flux computations. The Transport of Diluted Species interface always includes mass transport due to molecular diffusion. In this case the mass flux  $\mathbf{J}_i$  defines the diffusive flux vector

$$\mathbf{J}_i = -D \nabla c_i \quad (7-3)$$

An input field for the diffusion coefficient is available.

When *Migration in Electric Fields* is activated, the migration term is also added to the diffusive flux vector as shown in the section [Adding Transport Through Migration](#).

The third term on the left side of [Equation 7-2](#) describes the convective transport due to a velocity field  $\mathbf{u}$ . This field can be expressed analytically or obtained from coupling the physics interface to one that solves for fluid flow, such as *Laminar Flow*. Note that all fluid flow interfaces solve for the mass averaged velocity.

On the right-hand side of the mass balance equation ([Equation 7-2](#)),  $R_i$  represents a source or sink term, typically due to a chemical reaction or desorption on a porous matrix. To specify  $R_i$ , another node must be added to the Transport of Diluted Species interface — the **Reaction** node for example, which includes an input field for specifying a reaction expression using the variable names of all participating species.

### *Equilibrium Reaction Theory*

---

The feature Equilibrium Reaction is described in this section. A chemical equilibrium reaction system is defined by the stoichiometry of the reaction and the relation between the chemical activities of the chemical species participating in the reaction (the equilibrium condition).

The kinetics of the reaction is so fast that the equilibrium condition is fulfilled at all times in all space coordinates.

The equilibrium condition is commonly based on the stoichiometric coefficients,  $\nu_i$  (dimensionless), of the reaction; the species activities of the reacting species  $a_i$  (dimensionless); and an equilibrium constant,  $K_{\text{eq}}$  (1) according to:

$$K_{\text{eq}} = \frac{\prod_{i \in \text{products}} a_i^{\nu_i}}{\prod_{i \in \text{reactants}} a_i^{-\nu_i}}$$

where the species activities are defined as

$$a_i = \gamma_{c,i} \frac{c_i}{c_{a0}}$$

where  $c_{a0}$  (SI unit: mol/m<sup>3</sup>) is the standard molarity, and  $\gamma_{c,i}$  (dimensionless) an activity coefficient.

Defining the stoichiometric coefficients positive for products and negative for reactants, the above equilibrium condition can also be written:

$$K_{\text{eq}} = \prod_i a_i^{v_i}$$

The **Equilibrium Reaction** node solves for a reaction rate so that the equilibrium condition is always fulfilled in the domain. It is available for the Chemical Reaction Engineering Module; Battery Design Module; Corrosion Module; Electrochemistry Module; Electrodeposition Module; and Fuel Cell & Electrolyzer Module.



$\gamma_{e,i}$  is set to unity when the Equilibrium constant is selected in the Settings window. For nonunity activity coefficients, a user defined equilibrium condition can be used.

#### EQUILIBRIUM REACTIONS AND INFLOW BOUNDARY CONDITIONS

Contradictory constraints arise if the boundary conditions for concentrations or activities are set so that the domain equilibrium condition is not fulfilled. Special treatment is therefore needed at Inflow boundaries, where the concentrations are set for all species in the mass transport interfaces.

One way of avoiding competing constraints on an inflow boundary is to add an additional reaction coordinate degree of freedom, solved for to create a set of modified inflow concentrations that fulfill the domain equilibrium condition. The reaction coordinate gives rise to a concentration shift, which is the offset to the inflow concentrations provided by the user. The shift for each species obeys the stoichiometry of the reaction and the equilibrium expression. The modified inflow concentrations are then used in the boundary conditions for the domain mass transport equations. The resulting modified inflow concentrations can be seen as the stationary solution for a batch reactor with the user inflow concentrations as initial concentrations. In addition, the domain reaction rate degree of freedom of the equilibrium reaction is constrained to zero on all Inflow boundaries.

#### EQUILIBRIUM REACTIONS AND CONCENTRATION BOUNDARY CONDITIONS

No special treatment is made with regards to input concentration values of the Concentration boundary node. Using this feature, you can explicitly set one or a set of concentrations, and the equilibrium condition acts on the rest of the concentrations.

However, there is no solution to the problem if more concentrations than the number of species minus the number of equilibrium reactions are set using this feature.

### EQUILIBRIUM REACTIONS AND TIME-DEPENDENT SIMULATIONS

Spurious oscillations may occur in a time-dependent problem if the initial conditions do not fulfill the equilibrium condition. Since equilibrium reactions are assumed to be infinitely fast, the solution is to initialize the problem using an additional study step, solving for a stationary problem with all nonequilibrium reaction rates set to zero. Manual scaling of the reaction rate dependent variables is needed in this study step.

### *Convective Term Formulation*

---


The default node attributed to `assumes` chemical species transport through diffusion and convection (depending on the modules licensed, a check box to activate migration is available) and implements the mass balance equation in [Equation 7-2](#).

There are two ways to present a mass balance where chemical species transport occurs through diffusion and convection. These are the nonconservative and conservative formulations of the convective term:

$$\text{nonconservative: } \frac{\partial c}{\partial t} + \mathbf{u} \cdot \nabla c = \nabla \cdot \mathbf{J}_i + R \quad (7-4)$$

$$\text{conservative: } \frac{\partial c}{\partial t} + \nabla \cdot (c\mathbf{u}) = \nabla \cdot \mathbf{J}_i + R \quad (7-5)$$

and each is treated slightly differently by the solver algorithms. In these equations  $\mathbf{J}_i$  (SI unit:  $\text{mol}/(\text{m}^2 \cdot \text{s})$ ) is the diffusive flux vector,  $R$  (SI unit:  $\text{mol}/(\text{m}^3 \cdot \text{s})$ ) is a production or consumption rate expression, and  $\mathbf{u}$  (SI unit:  $\text{m}/\text{s}$ ) is the solvent velocity field. The diffusion process can be anisotropic, in which case  $D$  is a tensor.

If the conservative formulation is expanded using the chain rule, then one of the terms from the convection part,  $c\nabla \cdot \mathbf{u}$ , would equal zero for an incompressible fluid and would result in the nonconservative formulation above. This is in fact the default formulation in this physics interface. To switch between the two formulations, click the **Show** button (  ) and select **Advanced Physics Options**.



## *Solving a Diffusion Equation Only*

---

Remove the convection term from [Equation 7-4](#) and [Equation 7-5](#) by clearing the Convection check box in the Transport Mechanisms section for . The equation then becomes

$$\frac{\partial c}{\partial t} = \nabla \cdot \mathbf{J}_i + R$$

## *Mass Sources for Species Transport*

---

**Note:** The features below are only available in a limited set of add-on products. For a detailed overview of which features are available in each product, visit <https://www.comsol.com/products/specifications/>

---

There are two types of mass sources in the Transport of Diluted Species interface: point sources and line sources.

### **POINT SOURCE**

A point source is theoretically formed by assuming a mass injection/ejection,  $Q_c$  (SI unit:  $\text{mol}/(\text{m}^3 \cdot \text{s})$ ), in a small volume  $\delta V$  and then letting the size of the volume tend to zero while keeping the total mass flux constant. Given a point source strength,  $\dot{q}_{p,c}$  (SI unit:  $\text{mol}/\text{s}$ ), this can be expressed as

$$\lim_{\delta V \rightarrow 0} \int_{\delta V} Q_c = \dot{q}_{p,c} \quad (7-6)$$

An alternative way to form a point source is to assume that mass is injected/extracted through the surface of a small object. Letting the object surface area tend to zero while keeping the mass flux constant results in the same point source. For this alternative approach, effects resulting from the physical object's volume need to be neglected.

The weak contribution

$$\dot{q}_{p,c} \text{test}(c)$$

is added at a point in the geometry. As can be seen from [Equation 7-6](#),  $Q_c$  must tend to plus or minus infinity as  $\delta V$  tends to zero. This means that in theory the concentration also tends to plus or minus infinity.

Observe that “point” refers to the physical representation of the source. A point source can therefore only be added to points in 3D components and to points on the symmetry axis in 2D axisymmetry components. Other geometrical points in 2D components represent physical lines.

The finite element representation of Equation 7-6 corresponds to a finite concentration at a point with the effect of the point source spread out over a region around the point. The size of the region depends on the mesh and on the strength of the source. A finer mesh gives a smaller affected region but also a more extreme concentration value. It is important not to mesh too finely around a point source since this can result in unphysical concentration values. It can also have a negative effect on the condition number for the equation system.

### LINE SOURCE

A line source can theoretically be formed by assuming a source of strength  $Q_{l,c}$  (SI unit:  $\text{mol}/(\text{m}^3 \cdot \text{s})$ ), located within a tube with cross section  $\delta S$  and then letting  $\delta S$  tend to zero while keeping the total mass flux per unit length constant. Given a line source strength,  $\dot{q}_{l,c}$  (SI unit:  $\text{mol}/(\text{m} \cdot \text{s})$ ), this can be expressed as

$$\lim_{\delta S \rightarrow 0} \int_{\delta S} Q_{l,c} = q_{l,c} \quad (7-7)$$

As in the point source case, an alternative approach is to assume that mass is injected/extracted through the surface of a small object. This results in the same mass source, but requires that effects resulting from the physical object’s volume are neglected.

The weak contribution

$$q_{l,c} \text{test}(c)$$

is added on lines in 3D or at points in 2D (which represent cut-through views of lines). Line sources can also be added on the axisymmetry line in 2D axisymmetry components. It cannot, however, be added on geometrical lines in 2D since those represent physical planes.

As with a point source, it is important not to mesh too finely around the line source.



For feature node information, see [Line Mass Source](#) and [Point Mass Source](#).

**Note:** Migration is only available in a limited set of add-on products. For a detailed overview of which features are available in each product, visit <https://www.comsol.com/products/specifications/>

---

In addition to transport due to convection and diffusion, the Transport of Diluted Species interface supports ionic species transport by *migration*. This is done by selecting the Migration in Electric Field check box under the Transport Mechanisms section for the physics interface. The mass balance then becomes:

$$\frac{\partial c_i}{\partial t} + \nabla \cdot (-D_i \nabla c_i - z_i u_{m,i} F c_i \nabla V + c_i \mathbf{u}) = R_i \quad (7-8)$$

where

- $c_i$  (SI unit: mol/m<sup>3</sup>) denotes the concentration of species  $i$
- $D_i$  (SI unit: m<sup>2</sup>/s) is the diffusion coefficient of species  $i$
- $\mathbf{u}$  (SI unit: m/s) is the fluid velocity
- $F$  (SI unit: A·s/mol) refers to Faraday's constant
- $V$  (SI unit: V) denotes the electric potential
- $z_i$  (dimensionless) is the charge number of the ionic species, and
- $u_{m,i}$  (SI unit: mol·s/kg) is its ionic mobility

In this case the diffusive flux vector is

$$\mathbf{J}_i = -D_i \nabla c_i - z_i u_{m,i} F c_i \nabla V$$

The velocity,  $\mathbf{u}$ , can be a computed fluid velocity field from a Fluid Flow interface or a specified function of the spatial variables  $x$ ,  $y$ , and  $z$ . The potential can be provided by an expression or by coupling the system of equations to a current balance, such as the Electrostatics interface. Sometimes it is assumed to be a supporting electrolyte present, which simplifies the transport equations. In that case, the modeled charged species concentration is very low compared to other ions dissolved in the solution. Thus, the species concentration does not influence the solution's conductivity and the net charge within the fluid.

The Nernst–Einstein relation can in many cases be used for relating the species mobility to the species diffusivity according to

$$u_{m,i} = \frac{D_i}{RT}$$

where  $R$  (SI unit: J/(mol·K)) is the molar gas constant and  $T$  (SI unit: K) is the temperature.

---

**Note:** In the *Nernst–Planck Equations* interface, the ionic species contribute to the charge transfer in the solution. It includes an *electroneutrality condition* and also computes the electric potential field in the electrolyte. For more information, see [Theory for the Nernst–Planck Equations Interface](#). This interface is included in the Chemical Reaction Engineering Module.

---

### *Supporting Electrolytes*

---

In *electrolyte* solutions, a salt can be added to provide a high electrolyte conductivity and decrease the ohmic losses in a cell. These solutions are often called *supporting electrolytes*, buffer solutions, or carrier electrolytes. The added species, a negative and a positive ion pair, predominates over all other species. Therefore, the supporting electrolyte species can be assumed to dominate the current transport in the solution. In addition, the predominant supporting ions are usually selected so that they do not react at the electrode surfaces since the high conductivity should be kept through the process, that is, they should not be electroactive species. This also means that the concentration gradients of the predominant species in a supporting electrolyte are usually negligible.

Modeling and solving for a supporting electrolyte in the *Electrostatics* or *Secondary Current Distribution* interfaces will give a potential distribution that drives the migration in the Transport of Diluted Species Interface.

The current density vector is proportional to the sum of all species fluxes as expressed by Faraday’s law:

$$\mathbf{i} = F \sum_i z_i \mathbf{N}_i$$

The electroneutrality condition ensures that there is always a zero net charge at any position in a dilute solution. Intuitively, this means that it is impossible to create a current by manually pumping positive ions in one direction and negative ions in the other. Therefore, the convective term is canceled out to yield the following expression for the electrolyte current density, where  $j$  denotes the supporting species:

$$\mathbf{i} = F \sum_j -z_j^2 u_{m,j} F c_j \nabla \phi \quad (7-9)$$

Equation 7-9 is simply Ohm's law for ionic current transport and can be simplified to

$$\mathbf{i} = -\kappa \nabla \phi \quad (7-10)$$

where  $\kappa$  is the conductivity of the supporting electrolyte. A current balance gives the current and potential density in the cell

$$\nabla \cdot \mathbf{i} = 0$$

which, in combination with Equation 7-10, yields:

$$\nabla \cdot (-\kappa \nabla \phi) = 0 \quad (7-11)$$

Equation 7-11 can be easily solved using the Electrostatics or Secondary Current Distribution interface and, when coupled to the Transport in Diluted Species interface, the potential distribution shows up in the migration term.

### *Crosswind Diffusion*

---

Transport of diluted species applications can often result in models with a very high cell Péclet number — that is, systems where convection or migration dominates over diffusion. Streamline diffusion and crosswind diffusion are of paramount importance to obtain physically reasonable results. The Transport of Diluted Species interface provides two crosswind diffusion options using different formulations. Observe that crosswind diffusion makes the equation system nonlinear even if the transport equation is linear.

#### **DO CARMO AND GALEÃO**

This is the formulation described in [Numerical Stabilization](#) in the *COMSOL Multiphysics Reference Manual*. The method reduces over- and undershoots to a minimum, even for anisotropic meshes.

In some cases, the resulting nonlinear equation system can be difficult to converge. This can happen when the cell Péclet number is very high and the model contains many thin layers, such as contact discontinuities. You then have three options:

- Refine the mesh, especially in regions with thin layers.
- Use a nonlinear solver with a constant damping factor less than one.
- Switch to the Codina crosswind formulation.

#### **CODINA**

The Codina formulation is described in [Ref. 1](#). It adds diffusion strictly in the direction orthogonal to the streamline direction. Compared to the do Carmo and Galeão formulation, the Codina formulation adds less diffusion but is not as efficient at reducing over- and undershoots. It also does not work as well for anisotropic meshes. The advantage is that the resulting nonlinear system is easier to converge and that underresolved gradients are less smeared out.

#### *Danckwerts Inflow Boundary Condition*

---

Constraining the composition to fixed values at an inlet to a reactor may sometimes result in issues with unreasonably high reaction rates or singularities at the inlet boundary. These problems may many times be mitigated by using a flux boundary condition instead, based on the upstream concentrations and the fluid velocity at the boundary. In chemical engineering, this type of flux boundary condition is also known as a Danckwerts condition.

Use the Danckwerts condition to specify inlet conditions for domains where high reaction rates are anticipated in the vicinity to the inlet ([Ref. 2](#)).

Given an upstream concentration  $c_{i,0}$ , the Danckwerts inflow boundary condition prescribed the total flux as

$$\mathbf{n} \cdot (\mathbf{J}_i + \mathbf{u}c_i) = \mathbf{n} \cdot (\mathbf{u}c_{i,0}) \quad (7-12)$$

### VARIABLY SATURATED POROUS MEDIA

The following equations for the molar concentrations,  $c_i$ , describe the transport of solutes in a variably saturated porous medium for the most general case, when the pore space is primarily filled with liquid but also contain pockets or immobile gas:

$$\frac{\partial}{\partial t}(\theta_l c_i) + \frac{\partial}{\partial t}(\rho c_{P,i}) + \frac{\partial}{\partial t}(\theta_g c_{G,i}) + \mathbf{u} \cdot \nabla c_i = \nabla \cdot [(D_{D,i} + D_{e,i}) \nabla c_i] + R_i + S_i \quad (7-13)$$

On the left-hand side of Equation 7-13, the first three terms correspond to the accumulation of species within the liquid, solid, and gas phases, while the last term describes the convection due to the velocity field  $\mathbf{u}$  (SI unit: m/s).

In Equation 7-13  $c_i$  denotes the concentration of species  $i$  in the liquid (SI unit: mol/m<sup>3</sup>),  $c_{P,i}$  the amount adsorbed to solid particles (moles per unit dry weight of the solid), and  $c_{G,i}$  the concentration of species  $i$  in the gas phase.

The equation balances the mass transport throughout the porous medium using the porosity  $\varepsilon_p$ , the liquid volume fraction  $\theta_l$ ; the dry bulk density,  $\rho = (1 - \varepsilon_g)\rho_s$ , and the solid phase density  $\rho_s$ .

For saturated porous media, the liquid volume fraction  $\theta_l$  is equal to the porosity  $\varepsilon_p$ , but for unsaturated porous media, they are related by the saturation  $s$  as  $\theta_l = \varepsilon_p s$ . The resulting gas volume fraction in the case of an unsaturated porous medium is

$$\theta_g = \varepsilon_p - \theta_l = (1 - s)\varepsilon_p$$

On the right-hand side of Equation 7-13, the first term introduces the spreading of species due to mechanical mixing resulting from the porous media (dispersion), as well as from diffusion and volatilization to the gas phase. The dispersion tensor is denoted  $D_D$  (SI unit: m<sup>2</sup>/s) and the effective diffusion by  $D_e$  (SI unit: m<sup>2</sup>/s).

The last two terms on the right-hand side of Equation 7-13 describe production or consumption of the species;  $R_i$  is a reaction rate expression which can account for reactions in the liquid, solid, or gas phase, and  $S_i$  is an arbitrary source term, for example due to a fluid flow source or sink.

### Adsorption

The time evolution of the adsorption, the solute transport to or from the solid phase, is defined by assuming that the amount of solute adsorbed to the solid,  $c_{P,i}$ , is a function of the concentration in the fluid  $c_i$ . This implies that the solute concentration in the liquid and solid phase are in instant equilibrium. The adsorption term can be expanded to give

$$\frac{\partial}{\partial t}(\rho c_{P,i}) = \rho \frac{\partial c_{P,i}}{\partial c_i} \frac{\partial c_i}{\partial t} - c_{P,i} \rho_s \frac{\partial \varepsilon_P}{\partial t} = \rho K_{P,i} \frac{\partial c_i}{\partial t} - c_{P,i} \rho_s \frac{\partial \varepsilon_P}{\partial t} \quad (7-14)$$

where  $K_{P,i} = \partial c_{P,i} / \partial c_i$  is the adsorption isotherm.

### Volatilization

Volatilization is the process where a solute species in the liquid is transported to the gas phase due to vaporization. Assuming that the amount of solute in the gas phase,  $c_{G,i}$ , is a linear function of the liquid phase concentration, the volatilization term is defined as

$$\frac{\partial}{\partial t} \theta_g c_{G,i} = \theta_g \frac{\partial c_{G,i}}{\partial c_i} \frac{\partial c_i}{\partial t} + k_{G,i} c_i \frac{\partial \theta_g}{\partial t} = \theta_g k_{G,i} \frac{\partial c_i}{\partial t} + k_{G,i} c_i \frac{\partial \theta_g}{\partial t} \quad (7-15)$$

where  $k_{G,i} = \partial c_{G,i} / \partial c_i$  is the linear volatilization.

## SATURATED POROUS MEDIA

In the case of transport in a saturated porous medium,  $\theta_l = \varepsilon_P$ , and the governing equations are

$$\frac{\partial}{\partial t}(\varepsilon_P c_i) + \frac{\partial}{\partial t}(\rho c_{P,i}) + \mathbf{u} \cdot \nabla c_i = \nabla \cdot [(D_{D,i} + D_{e,i}) \nabla c_i] + R_i + S_i \quad (7-16)$$

### Convection in Porous Media

Convection (also called advection) describes the movement of a species, such as a pollutant, with the bulk fluid velocity. The velocity field  $\mathbf{u}$  corresponds to a superficial volume average over a unit volume of the porous medium, including both pores and matrix. This velocity is sometimes called *Darcy velocity*, and defined as volume flow



rates per unit cross section of the medium. This definition makes the velocity field continuous across the boundaries between porous regions and regions with free flow.



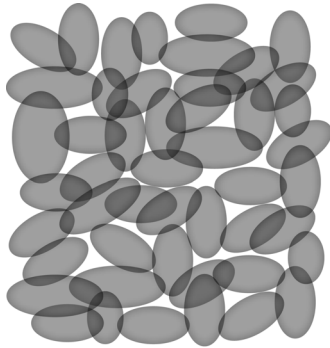
The velocity field to be used in the Model Inputs section on the physics interface can, for example, be prescribed using the velocity field from a Darcy's Law or a Brinkman Equations interface.

The average linear fluid velocities  $\mathbf{u}_a$ , provides an estimate of the fluid velocity within the pores:

$$\mathbf{u}_a = \frac{\mathbf{u}}{\varepsilon_p} \quad \text{Saturated}$$

$$\mathbf{u}_a = \frac{\mathbf{u}}{\theta_l} \quad \text{Unsaturated}$$

where  $\varepsilon_p$  is the porosity and  $\theta_l = s\varepsilon_p$  the liquid volume fraction, and  $s$  the saturation, a dimensionless number between 0 and 1.



*Figure 7-2: A block of a porous medium consisting of solids and the pore space between the solid grains. The average linear velocity describes how fast the fluid moves within the pores. The Darcy velocity attributes this flow over the entire fluid-solid face.*

#### CONVECTIVE TERM FORMULATION


The Transport of Diluted Species in Porous Media interface includes two formulations of the convective term. The conservative formulation of the species equations in [Equation 7-13](#) is written as:

$$\frac{\partial}{\partial t}(\theta_l c_i) + \frac{\partial}{\partial t}(\rho c_{p,i}) + \frac{\partial}{\partial t}(\theta_g c_{g,i}) + \nabla \cdot \mathbf{u} c_i = \nabla \cdot [(D_{D,i} + D_{e,i}) \nabla c_i] + R_i + S_i \quad (7-17)$$

If the conservative formulation is expanded using the chain rule, then one of the terms from the convection part,  $c_i \nabla \cdot \mathbf{u}$ , would equal zero for an incompressible fluid and would result in the nonconservative formulation described in [Equation 7-13](#).

When using the nonconservative formulation, which is the default, the fluid is assumed incompressible and divergence free:  $\nabla \cdot \mathbf{u} = 0$ . The nonconservative formulation improves the stability of systems coupled to a momentum equation (fluid flow equation).



To switch between the two formulations, click the **Show** button (  ) and select **Advanced Physics Options**. In the section **Advanced Settings** select either **Nonconservative form** (the default) or **Conservative form**. The conservative formulation should be used for compressible flow.

### *Diffusion in Porous Media*

The effective diffusion in porous media,  $D_e$ , depends on the structure of the porous material and the phases involved. Dependent on whether the transport of diluted species occurs in free flow or in saturated or unsaturated porous media, the effective diffusivity is defined as:

$$\begin{aligned} D_e &= D_F && \text{Free Flow} \\ D_e &= \frac{\varepsilon_p}{\tau_L} D_L && \text{Saturated Porous Media} \\ D_e &= \frac{\theta_l}{\tau_L} D_L && \text{Unsaturated Porous Media} \\ D_e &= \frac{\theta_l}{\tau_L} D_L + \frac{\theta_g}{\tau_G} k_G D_G && \text{Unsaturated with Volatilization} \end{aligned}$$

Here  $D_F$ ,  $D_L$ , and  $D_G$  are the single-phase diffusion coefficients for the species diluted in a fluid, a pure liquid, and a gas phase, respectively (SI unit:  $\text{m}^2/\text{s}$ ), and  $\tau_F$ ,  $\tau_L$ , and  $\tau_G$  are the corresponding tortuosity factors (dimensionless).

The tortuosity factor accounts for the reduced diffusivity due to the fact that the solid grains impede Brownian motion. The interface provides predefined expressions to

compute the tortuosity factors in partially saturated porous media according to the Millington and Quirk model (Ref. 3):

$$\tau_L = \theta_l^{-7/3} \varepsilon_p^2, \tau_G = \theta_g^{-7/3} \varepsilon_p^2$$

and Bruggeman model

$$\tau_L = \theta_l^{-5/2} \varepsilon_p^2, \tau_G = \theta_g^{-5/2} \varepsilon_p^2$$

For saturated porous media  $\theta_l = \varepsilon_p$ . The fluid tortuosity for the Millington and Quirk model is

$$\tau_L = \varepsilon_p^{-1/3}$$

and for the Bruggeman model the tortuosity is defined as

$$\tau_L = \varepsilon_p^{-1/2}$$

User defined expressions for the tortuosity factor can also be applied.

## Dispersion

---

The contribution of dispersion to the mixing of species typically overshadows the contribution from molecular diffusion, except when the fluid velocity is very low.

The spreading of mass, as a fluid travel through a porous medium is caused by several contributing effects. Local variations in fluid velocity lead to mechanical mixing referred to as dispersion occurs because the fluid in the pore space flows around solid particles, so the velocity field varies within pore channels. The spreading in the direction parallel to the flow, or *longitudinal dispersivity*, typically exceeds the *transverse dispersivity* from up to an order of magnitude. Being driven by the concentration gradient alone, molecular diffusion rate is low relative to the mechanical dispersion, except at very low fluid velocities.

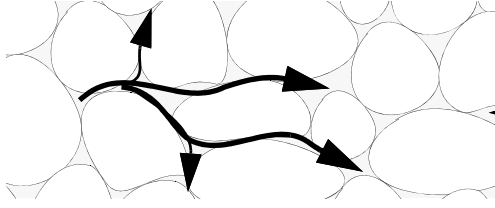


Figure 7-3: Spreading of fluid around solid particles in a porous medium.

The dispersion is controlled through the dispersion tensor  $D_D$ . The tensor components can either be given by user-defined values or expressions or derived from the directional dispersivities.

Using the longitudinal and transverse dispersivities in 2D, the dispersivity tensor components are (Ref. 5):

$$D_{Dii} = \alpha_L \frac{u_i^2}{|\mathbf{u}|} + \alpha_T \frac{u_j^2}{|\mathbf{u}|}$$

$$D_{Dij} = D_{Dji} = (\alpha_L - \alpha_T) \frac{u_i u_j}{|\mathbf{u}|}$$

In these equations,  $D_{Dii}$  (SI unit:  $\text{m}^2/\text{s}$ ) are the principal components of the dispersivity tensor, and  $D_{Dji}$  and  $D_{Dij}$  are the cross terms. The parameters  $\alpha_L$  and  $\alpha_T$  (SI unit: m) specify the longitudinal and transverse dispersivities; and  $u_i$  (SI unit: m/s) stands for the velocity field components.

In order to facilitate modeling of stratified porous media in 3D, the tensor formulation by Burnett and Frind (Ref. 4) can be used. Consider a transverse isotropic media, where the strata are piled up in the  $z$  direction, the dispersivity tensor components are:

$$\begin{aligned}
D_{Lxx} &= \alpha_1 \frac{u^2}{|\mathbf{u}|} + \alpha_2 \frac{v^2}{|\mathbf{u}|} + \alpha_3 \frac{w^2}{|\mathbf{u}|} \\
D_{Lyy} &= \alpha_1 \frac{v^2}{|\mathbf{u}|} + \alpha_2 \frac{u^2}{|\mathbf{u}|} + \alpha_3 \frac{w^2}{|\mathbf{u}|} \\
D_{Lzz} &= \alpha_1 \frac{w^2}{|\mathbf{u}|} + \alpha_2 \frac{u^2}{|\mathbf{u}|} + \alpha_3 \frac{v^2}{|\mathbf{u}|} \\
D_{Lxy} &= D_{Lyx} = (\alpha_1 - \alpha_2) \frac{uv}{|\mathbf{u}|} \\
D_{Lxz} &= D_{Lzx} = (\alpha_1 - \alpha_3) \frac{uw}{|\mathbf{u}|} \\
D_{Lyz} &= D_{Lzy} = (\alpha_1 - \alpha_3) \frac{vw}{|\mathbf{u}|}
\end{aligned} \tag{7-18}$$

In Equation 7-18 the fluid velocities  $u$ ,  $v$ , and  $w$  correspond to the components of the velocity field  $\mathbf{u}$  in the  $x$ ,  $y$ , and  $z$  directions, respectively, and  $\alpha_1$  (SI unit: m) is the longitudinal dispersivity. If  $z$  is the vertical axis,  $\alpha_2$  and  $\alpha_3$  are the dispersivities in the transverse horizontal and transverse vertical directions, respectively (SI unit: m). Setting  $\alpha_2 = \alpha_3$  gives the expressions for isotropic media shown in Bear (Ref. 5 and Ref. 6).

### *Adsorption*

---

As species travel through a porous medium they typically attach to (adsorb), and detach (desorb) from the solid phase, which slows chemical transport through the porous medium. Adsorption and desorption respectively reduces or increases species concentrations in the fluid. The adsorption properties vary between chemicals, so a plume containing multiple species can separate into components (Ref. 7). The Adsorption feature includes four predefined and one user defined relationships to predict the solid concentrations,  $c_{pi}$  from the concentration in the liquid phase,  $c_i$ :

$$\begin{aligned}
c_P &= c_{P\max} \frac{K_L c}{1 + K_L c} && \text{Langmuir} \\
c_P &= K_F \left( \frac{c}{c_{\text{ref}}} \right)^N && \text{Freundlich} \\
c_P &= c_{P\max} \frac{b_T c}{(1 + (b_T c)^{N_T})^{1/N_T}} && \text{Toth} \\
c_P &= \frac{K_B c_0 c}{(c_S - c) \left( 1 + (K_B - 1) \frac{c}{c_S} \right)} && \text{BET}
\end{aligned} \tag{7-19}$$

The above equations contains the following parameters:

- Freundlich ([Ref. 8](#))
  - Freundlich constant  $K_F$  (SI unit: mol/kg), Freundlich exponent  $N_F$  (dimensionless), and reference concentration  $c_{\text{ref}}$  (SI unit: mol/m<sup>3</sup>).
- Langmuir ([Ref. 9](#))
  - Langmuir constant  $K_L$  (SI unit: m<sup>3</sup>/mol), and adsorption maximum  $c_{P\max}$  (SI unit: mol/kg).
- Toth ([Ref. 10](#))
  - Toth constant  $b_T$  (SI unit: m<sup>3</sup>/mol), Toth exponent  $N_T$  (dimensionless), and adsorption maximum  $c_{P\max}$  (SI unit: mol/kg).
- BET (Brunauer–Emmett–Teller, [Ref. 11](#))
  - BET constant  $K_B$  (dimensionless), the monolayer adsorption capacity  $c_0$  (SI unit: mol/kg) and the saturation concentration,  $c_S$  (SI unit: mol/m<sup>3</sup>).

These predefined expressions are adsorption isotherms that describe the amount of species sorbed to the solid. Applying the isotherms, as defined in the [Adsorption](#) section above, assumes that adsorption is at equilibrium locally. This implies that the species phase transfer, between the fluid and the solid phase (within the porous medium), is much faster than the transport within the fluid phase.

Using a [Species Source](#) feature, arbitrary expressions can be entered to define, for example, nonequilibrium and temperature-dependent adsorption laws, in the manner of Fetter ([Ref. 12](#)), or Bear and Verruijt ([Ref. 13](#)).

The retardation factor, **RF**, describes how adsorption slows the solute velocity,  $\mathbf{u}_c$ , relative to the average linear velocity of the fluid,  $\mathbf{u}_a$ , as in

$$\text{RF} = 1 + \frac{\rho_b}{\theta} \frac{\partial c_p}{\partial c} = \frac{\mathbf{u}_a}{\mathbf{u}_c}$$

If the contaminant moves at the average linear velocity of the fluid for  $\text{RF} = 1$ . For  $\text{RF} > 1$ , the contaminant velocity is lower than the fluid velocity owing to residence time on solids.

## Reactions

---

Chemical reactions of all types influence species transport in porous media. Examples include biodegradation, radioactive decay, transformation to tracked products, temperature- and pressure-dependent functions, exothermic reactions, and endothermic reactions. The reactions represent change in species concentration per unit volume porous medium per time. Reaction terms are used on the right-hand side of the governing equation to represent these processes. For reactions in a fluid phase, multiply the expression by the fluid volume fraction  $\theta$ . Similarly, solid phase reaction expressions include the bulk density,  $\rho_b$ , and gas phase reactions include the gas volume fraction,  $\alpha_v$ .

The following expressions define some common types of reactions:

$$\begin{aligned} R_{Li} &= -\theta \frac{\ln 2}{\lambda_{Li}} c_i && \text{Radioactive decay — liquid} \\ R_{Pi} &= -\rho_b \frac{\ln 2}{\lambda_{Pi}} \left( \frac{\partial c_{Pi}}{\partial c_i} \right) c_i && \text{Radioactive decay — solid} \\ R_{Gi} &= -\frac{\ln 2}{\lambda_{Gi}} \alpha_v \left( \frac{\partial c_{Gi}}{\partial c} \right) c_i && \text{Radioactive decay — gas} \\ R_{Lk} &= \theta \zeta_{Li} c_i && \text{Creation from parent } c_{Li} \text{ — liquid} \\ R_{Pk} &= \rho_b \zeta_{Pi} \left( \frac{\partial c_{Pi}}{\partial c_i} \right) c_i && \text{Creation from sorbed parent } c_{Pi} \text{ — solid} \\ R_{Gk} &= -\zeta \alpha_v \left( \frac{\partial c_{Gi}}{\partial c} \right) c_i && \text{Reaction — gas} \end{aligned}$$

where  $\lambda$  is the chemical half life,  $\zeta$  is a reaction rate, and the subscripts  $L$ ,  $P$ , and  $G$  denote liquid, solid, and gas phases, respectively. In the equations, the reactions either depend on liquid concentration  $c_i$  or solid phase concentrations  $c_{Pi}$  obtained using the sorption derivative with  $c_i$  or gas phase concentration  $c_{Gi}$  depending on the gas volume fraction, the volatilization, and the liquid concentration.

Reaction rates can vary with results from other equations in your model, such as temperature. For example, enter the Arrhenius rate law given in [Ref. 14](#):

$$\zeta_T = \zeta_R \exp \left[ \frac{E_a(T - T_R)}{R_u T T_R} \right] \quad (7-20)$$

In [Equation 7-20](#),  $T$  denotes the current absolute temperature,  $T_R$  denotes the reference absolute temperature,  $E_a$  is the activation energy, and  $R_u$  is the universal gas constant.

### *Mass Transport in Fractures*

---

When thin fractures occur in porous media, fluid flow tends to move faster along the fracture than in the surrounding media. The transport of chemical species therefore also occur also faster in the direction of the fractures.

The fluid flow in a fracture can be modeled using Darcy's law formulated in a thin sheet of porous medium (a fracture):

$$\mathbf{u} = \frac{\kappa}{\mu} \nabla_t p$$

Here  $\mathbf{u}$  is the tangential Darcy velocity,  $\kappa$  is the fracture permeability,  $\mu$  the fluid's dynamic viscosity, and  $\nabla_t p$  is the tangential gradient of the fluid pressure.

The equation to solve for mass transport of species  $c_i$  in a thin fracture, embedded in a porous media, is derived from [Equation 7-13](#). The resulting equation is:

$$d_{fr} \left( \frac{\partial \rho_b c_{p,i}}{\partial t} + \frac{\partial \varepsilon_p c_i}{\partial t} + \nabla_t \cdot (D_{e,i} \nabla_t c_i) + \mathbf{u} \cdot \nabla_t c_i \right) = d_{fr} R_i + d_{fr} S_i + n_0 \quad (7-21)$$

Here  $d_{fr}$  is the fracture thickness,  $c_{p,i}$  the amount of species adsorbed to (or desorbed from) the porous matrix (moles per unit dry weight of the solid),  $\varepsilon_p$  is the fracture porosity, and  $D_e$  is the effective diffusivity. The first two terms on the right hand side represent source terms from reactions, and  $n_0$  corresponds to out-of plane flux from the adjacent porous domain.

In order to arrive at the tangential differential equation, the gradient is split into the contributions normal and tangential to the fracture:

$$\nabla c_i = \nabla_n c_i + \nabla_t c_i$$



The normal gradient is defined in the direction normal to the boundary representing the fracture and the tangential gradient is defined along the boundary. Assuming that the variations in the normal (thin) direction of the fracture are negligible compared to those in the tangential direction, the gradient is simplified as:

$$\nabla c_i = \nabla_t c_i$$

Using , the transport along fracture boundaries alone is solved for. In this case the transport in the surrounding porous media neglected and the out-of plane flux  $n_0$  vanishes.





See [Fracture](#) for more information about the boundary feature solving [Equation 7-21](#). See for more information about the physics interface solving the equation on boundaries only.

## References

1. R. Codina, “A discontinuity-capturing crosswind-dissipation for the finite element solution of the convection-diffusion equation”, *Computer Methods in Applied Mechanics and Engineering*, vol. 110, pp. 325–342, 1993.
2. P.V. Danckwerts, “Continuous flow systems: Distribution of residence times”, *Chem. Eng. Sci.*, vol. 2, no. 1, 1953.
3. R.J. Millington and J.M. Quirk, “Permeability of Porous Solids”, *Trans. Faraday Soc.*, vol. 57, pp. 1200–1207, 1961.
4. R.D. Burnett and E.O. Frind, “An Alternating Direction Galerkin Technique for Simulation of Groundwater Contaminant Transport in Three Dimensions: 2. Dimensionality Effects”, *Water Resour. Res.*, vol. 23, no. 4, pp. 695–705, 1987.
5. J. Bear, *Hydraulics of Groundwater*, McGraw Hill, 1979.
6. J. Bear, *Dynamics of Fluids in Porous Media*, Elsevier Scientific Publishing, 1972.
7. D.M. Mackay, D.L. Freyberg, P.V. Roberts, and J.A. Cherry, “A Natural Gradient Experiment on Solute Transport in a Sand Aquifer: 1. Approach and Overview of Plume Movement”, *Water Resour. Res.*, vol. 22, no. 13, pp. 2017–2030, 1986.
8. H.M.F. Freundlich, “Über die Adsorption in Lösungen” (in German), *Z. Phys. Chem.*, vol 57, pp. 385–470, 1906

9. I. Langmuir, “The Constitution and Fundamental Properties of Solids and Liquids, Part I Solids”, *J. Amer. Chem. Soc.*, vol. 38, pp. 2221–2295, 1916.
10. J. Toth, “State equation of the solid gas interface layer”, *Acta Chem. Acad. Hung.* vol. 69, pp. 311–317, 1971.
11. S. Brunauer, P.H. Emmett, E. Teller, “Adsorption of gases in multimolecular layers”, *J. Am. Chem. Soc.*, vol. 60 (2), pp. 309–319, 1938.
12. C.W. Fetter, *Contaminant Hydrogeology*, Prentice Hall, 1999.
13. J. Bear and A. Verruijt, *Modeling Groundwater Flow and Pollution*, D. Reidel Publishing, 1994.
14. I. Langmuir, “Chemical Reactions at Low Temperatures”, *J. Amer. Chem. Soc.*, vol. 37, 1915.
15. J.M. Coulson and J.F. Richardson, *Chemical Engineering*, vol. 2, 4th ed., Pergamon Press, Oxford, U.K., 1991.
16. J.M. Coulson and J.F. Richardson, *Chemical Engineering*, vol. 1, 4th ed., Pergamon Press, Oxford, U.K., 1991.
17. D.E Rosner, *Transport Processes in Chemically Reacting Flow Systems*, ISBN-13: 978-1483130262, Butterworth–Heinemann, 1986.
18. R.B. Bird, W.E. Stewart, and E.N. Lightfoot, *Transport Phenomena*, 2nd ed., John Wiley & Sons, Inc., 2007.

# Heat Transfer and Nonisothermal Flow Interfaces

There are several physics interfaces included in the Polymer Flow Module to model heat transfer and nonisothermal flow. These physics interfaces are found under the **Heat Transfer** branch (  ) as well as those under the **Fluid Flow>Nonisothermal Flow** branch (  ).

The basic functionality for simulating heat transfer in homogeneous media is described in [The Heat Transfer Interfaces](#) in the *COMSOL Multiphysics Reference Manual*.

In this chapter:

- [Modeling Heat Transfer in the Polymer Flow Module](#)
- [Boundary Features in the Heat Transfer Interfaces](#)
- [The Nonisothermal Flow, Laminar Flow and Viscoelastic Flow Interfaces](#)
- [Theory for the Nonisothermal Viscoelastic Flow](#)

# Modeling Heat Transfer in the Polymer Flow Module

Heat transfer is an important phenomenon in many industrial processes. Typical examples are polymer extrusion, injection molding, and mixing of non-Newtonian fluids. Due to the strong temperature dependence of the constitutive parameters, it is necessary to include thermal variations in nonisothermal flow of non-Newtonian fluids. In flows through narrow passages, considerable heat is generated in the flowing fluid by viscous dissipation.

Other typical examples of industrial heat transfer are heating and cooling operations. Fluid plays a major role in transporting heat and a detailed description of the flow field is necessary to accurately describe such processes. The temperature, in turn, affects the fluid properties and can alter the flow field. Natural convection is an example of this.

## *Specifying Material Properties for Non-Newtonian Materials*


---


For many complex fluids, the apparent viscosity has a strong dependence on the temperature. The Polymer Flow Module helps you specify the material properties easily.

For inelastic non-Newtonian fluids, two alternatives are available. The first option is to specify the material using the non-Newtonian material property groups (see [Fluid Flow Material Properties: Inelastic Non-Newtonian Material Model](#) in the *COMSOL Multiphysics Reference Manual*). Alternatively, it is possible to use built-in functionality to account for temperature variations using predefined thermal functions (see [Fluid Properties](#) for more details). The later option is also available for viscoelastic fluids.

## *Selecting the Right Physics Interface*

---

The **Heat Transfer** branch (  ) has a number of physics interfaces that can be used to model energy transport. One or more of these can be added; either by themselves or together with other physics interfaces, typically flow physics interfaces.

The **Nonisothermal Flow, Laminar Flow** (  ) interface, **Viscoelastic Flow** interface (  ) (located under the **Fluid Flow** branch) and **Conjugate Heat Transfer** (  ) ( located under the **Heat Transfer** branch) multiphysics interfaces automatically couple the flow and

heat equations and they also provide other functionality, that is not readily available when adding the interfaces separately.



- [Theory for the Heat Transfer Module](#) in the *Heat Transfer Module User's Guide*
- 

#### *Coupling to Other Physics Interfaces*

Often, you are simulating applications that couple heat transfer in the flow to another type of phenomenon described by another physics interface. This can, for example, include chemical reactions and mass transport, as covered by the physics interfaces in the Chemical Species Transport branch.

In addition, if you also have the Heat Transfer Module, it includes more detailed descriptions and tools for simulating energy transport, such as surface-to-surface and participating media radiation.

# Boundary Features in the Heat Transfer Interfaces

Additionally to the basic boundary conditions for the **Heat Transfer in Fluids** and **Heat Transfer in Solids and Fluids** interfaces that are described in [The Heat Transfer Interfaces](#) in the *COMSOL Multiphysics Reference Manual*, the Polymer Flow Module provides boundary conditions that are tailored to model inflow and open boundary conditions for fluid flows.

At the inlet boundary of a fluid domain, the [Inflow](#) boundary condition defines a heat flux that accounts for the energy that would normally be brought by the fluid flow if the channel upstream to the inlet were modeled.

The [Open Boundary](#) condition is used to model the boundary between a fluid domain and the rest of the same fluid domain that is not represented in the geometry. At the open boundary both inflow and outflow conditions are supported.

## *Inflow*

---

Use this node to model inflow of heat through a virtual domain, at inlet boundaries of nonsolid domains. The upstream temperature  $T_{\text{ustr}}$  and upstream absolute pressure  $p_{\text{ustr}}$  are known at the outer boundary of the virtual domain. The inflow condition is often similar to a temperature condition. However, using this condition, the upstream temperature is not strictly enforced at the inlet. The higher the flow rate, the smallest the difference between the inlet temperature and the upstream temperature. In case of heat sources or temperature constraints close to the inlet boundary, the inflow condition induces a temperature profile that approaches the temperature profile that would be obtained by computing the solution in the virtual domain upstream the inlet.

This boundary condition estimates the heat flux through the inlet boundary from a Danckwerts condition on the enthalpy:

$$-\mathbf{n} \cdot \mathbf{q} = \rho \Delta H \mathbf{u} \cdot \mathbf{n} \quad (8-1)$$

where  $\Delta H$  is the sensible enthalpy. See [Theory for the Inflow Boundary Condition](#) in the *Heat Transfer Module User's Guide* for details.

## UPSTREAM PROPERTIES

The **Upstream temperature**  $T_{\text{ustr}}$  should be specified. For **User defined**, enter a value or expression. Else, select any available input (like **Ambient temperature** defined in an **Ambient Properties** node under **Definitions**).

Select the **Specify upstream absolute pressure** check box to set a different value than the absolute inlet pressure for the **Upstream absolute pressure**  $p_{\text{ustr}}$ , and account for work done by pressure forces in the expression of the enthalpy. For **User defined**, enter a value or expression. Else, select an **Ambient absolute pressure** defined in an **Ambient Properties** node under **Definitions**.

### *Open Boundary*

---

Use this condition to define a boundary as the limit between a fluid domain of the geometry and the rest of the same fluid domain that is not represented in the geometry. At the open boundary both inflow and outflow conditions are supported.

For an outgoing fluid flow across the boundary, it applies a zero conductive flux condition:

$$-\mathbf{n} \cdot \mathbf{q} = 0, \quad \mathbf{n} \cdot \mathbf{u} \geq 0 \quad (8-2)$$

For an incoming flow of velocity field  $\mathbf{u}$  across the boundary, it accounts by default for the heat flux induced by the flow rate through a Danckwerts condition on the enthalpy:

$$-\mathbf{n} \cdot \mathbf{q} = \rho \Delta H \mathbf{u} \cdot \mathbf{n}, \quad \mathbf{n} \cdot \mathbf{u} < 0 \quad (8-3)$$

where  $\Delta H$  is the sensible enthalpy. See [Theory for the Inflow Boundary Condition](#) in the *Heat Transfer Module User's Guide* for details.

Alternatively the open boundary condition can set a constraint on the temperature for an incoming flow

$$T = T_{\text{ustr}}, \quad \mathbf{n} \cdot \mathbf{u} < 0 \quad (8-4)$$

Both conditions are based on the knowledge of the upstream temperature at the outer boundary of the virtual domain, but the first one should be preferred to account for the feedback of the model heat sources and temperature constraints on the inlet temperature profile.

### UPSTREAM PROPERTIES

Enter the **Upstream temperature**  $T_{\text{ustr}}$  outside the open boundary. For **User defined**, enter a value or expression. Else, select any available input (like an **Ambient temperature** defined in an **Ambient Properties** node under **Definitions**).

### INFLOW CONDITION

Unfold this section to change the inflow condition:

- The **Flux (Danckwerts)** condition (default) prescribes the flux defined from the upstream temperature and the fluid velocity at the boundary. Using this condition the temperature condition is not strictly enforced at the inlet. Instead a flux is set on this boundary. The higher the flow rate, the smallest difference between the inlet temperature and upstream temperature.
- The **Discontinuous Galerkin constraints** option enforces the upstream temperature on the boundary



# The Nonisothermal Flow, Laminar Flow and Viscoelastic Flow Interfaces

In this section:

- [Advantages of Using the Multiphysics Interfaces](#)
- [The Nonisothermal Flow, Laminar Flow and Viscoelastic Flow Interfaces](#)

## *Advantages of Using the Multiphysics Interfaces*


---

The **Nonisothermal Flow** interfaces combine the heat equation with flow equations in a Fluid domain. The advantage of using the multiphysics interfaces is that predefined couplings are available in both directions. In particular, the physics interfaces use the same definition of the density, which can therefore be a function of both pressure and temperature. Solving this coupled system of equations usually requires numerical stabilization accounting for the couplings, which the predefined multiphysics interfaces also set up.

## *The Nonisothermal Flow, Laminar Flow and Viscoelastic Flow Interfaces*

---

The **Nonisothermal Flow and Conjugate Heat Transfer Interfaces** that combine the heat equation with laminar flow in a fluid domain are described in the *COMSOL Multiphysics Reference Manual*. The Polymer Flow module has additional functionality to couple the viscoelastic flow and heat transfer.

When the **Viscoelastic Flow** multiphysics interface () is added from the **Fluid Flow>Nonisothermal Flow** branch of the **Model Wizard** or **Add Physics** windows, the **Heat Transfer in Fluids** and the **Viscoelastic Flow** interfaces are added to the **Model Builder**. In addition, the **Multiphysics** node is added, which automatically includes the multiphysics coupling feature **Nonisothermal Flow**.

### **ENERGY BALANCE**

If the coupled interface is **Laminar Flow**, the **Include viscous dissipation** check box is selected by default to account for the heat source corresponding to viscous heating. Because it may induce an extra computational cost it should be only selected in application where such effect is expected. If no information on this is available,

selecting the option ensures that the energy balance for the heat and the flow equation is respected.

If the coupled interface is **Viscoelastic Flow**, the **Include irreversible losses** check box is available. The check box is selected by default to account for the heat source corresponding to the heating due to irreversible losses.

The Polymer Flow module has additional functionality to simplify the solution of the natural convection problems. The Boussinesq approximation can be used to include buoyancy effects, without having to solve the compressible formulation of the Navier-Stokes equation. The approximation is accurate when density variations are small, that reduces the nonlinearity of the problem. The Boussinesq approximation assumes that variations in density have no effect on the flow field, except that they give rise to buoyancy forces.

## MATERIAL PROPERTIES

### *Boussinesq Approximation*

When the **Compressibility** setting in the fluid flow interface is set to **Incompressible**, select the **Boussinesq approximation** check box in order to use material data evaluated at the reference temperature and reference pressure. If gravity is included in the physics, the buoyancy contribution in the momentum equation is linearized with respect to temperature.

### *Density*

Select an option from the **Specify density** list: **From heat transfer interface** (the default); **From fluid flow interface**; **Custom, linearized density**; or **Custom**:

- For **From heat transfer interface**: define the **Density**  $\rho$  in the Thermodynamics, Fluid section of the **Fluid** node, in the Heat Transfer coupled interface. Depending on the **Fluid type** option in this node, the density may be taken from material, set directly, or computed by using the ideal gas law. The same value is automatically set in the **Fluid Properties** section of the **Fluid Properties** node, in the Fluid Flow coupled interface.
- For **From fluid flow interface**: define the **Density**  $\rho$  in the **Fluid Properties** section of the **Fluid Properties** node, in the Fluid Flow coupled interface. The same value is automatically set in the Thermodynamics, Fluid section of the **Fluid** node, in the Heat Transfer coupled interface.
- For **Custom, linearized density**, enter the **Reference density**  $\rho_{\text{ref}}$  (SI unit:  $\text{kg}/\text{m}^3$ ) and the **Coefficient of thermal expansion**  $\alpha_p$  (SI unit:  $1/\text{K}$ ), or select **From material**, or

select a variable in the list if available. Regardless how the properties are defined they should be constant. If material properties are not constant you should consider using any of the other options to define the density. Also, if **Custom, linearized density** is used for incompressible flow, the density  $\rho$  is evaluated to  $\rho_{\text{ref}}$ . In this case  $\alpha_p$  is not used unless **Boussinesq approximation** is selected. The same value is automatically set in the **Fluid Properties** section of the **Fluid Properties** node, in the Fluid Flow coupled interface, and in the Thermodynamics, Fluid section of the **Fluid** node, in the Heat Transfer coupled interface.

- For **Custom**, enter a **Density**  $\rho$  (SI unit:  $\text{kg}/\text{m}^3$ ), or select a density in the list if available. The same value is automatically set in the **Fluid Properties** section of the **Fluid Properties** node, in the Fluid Flow coupled interface, and in the Thermodynamics, Fluid section of the **Fluid** node, in the Heat Transfer coupled interface.

The density definition in the **Nonisothermal Flow** node ensures that the same definition of the density is used in the fluid flow and heat transfer interfaces. When **Include gravity** is selected and the **Compressibility** is set to **Incompressible flow** in the fluid interface properties, the gravity forces are defined using the coefficient of thermal expansion. Along with the fact that the material properties are evaluated for a constant temperature and pressure, this gravity force definition corresponds to the Boussinesq approximation. Unless the density is defined as **Custom, linearized density** the coefficient of thermal expansion is evaluated from the fluid density.

The Viscoelastic Flow interface only supports incompressible flow.

# Theory for the Nonisothermal Viscoelastic Flow

The theory for the Viscoelastic Flow interface is described in this section:

- [Nonisothermal Viscoelastic Flow Theory](#)
- [References for the Nonisothermal Viscoelastic Flow Interfaces](#)

## *Nonisothermal Viscoelastic Flow Theory*

---

In many industrial application involving the polymer processing, the flow conditions are nonisothermal. The material properties of viscoelastic fluids, such as viscosity and relaxation time show significant temperature dependence. Therefore, the viscoelastic stress depends on the temperature distribution. Another important aspect to consider is the internal heat production.

The **Nonisothermal Flow, Viscoelastic Flow** interface contains the equation for the flow of viscoelastic fluid:

$$\rho \nabla \cdot \mathbf{u} = 0 \quad (8-5)$$

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho (\mathbf{u} \cdot \nabla) \mathbf{u} = \nabla \cdot [-p \mathbf{I} + \boldsymbol{\tau}] + \mathbf{F} \quad (8-6)$$

where

- $\rho$  is the density (SI unit: kg/m<sup>3</sup>)
- $\mathbf{u}$  is the velocity vector (SI unit: m/s)
- $p$  is the pressure (SI unit: Pa)
- $\mathbf{F}$  is the body force vector (SI unit: N/m<sup>3</sup>)
- $\boldsymbol{\tau}$  is the extra stress tensor (SI unit: Pa).

The extra stress tensor is defined as a sum of a viscous and a viscoelastic contribution as

$$\boldsymbol{\tau} = 2\mu_s \mathbf{S} + \mathbf{T}_e \quad (8-7)$$

where  $\mu_s$  is the solvent viscosity,  $\mathbf{S}$  is the strain-rate tensor, and  $\mathbf{T}_e$  is the elastic stress tensor that can be represented as a sum of the individual modes. The system of

equations [Equation 8-5](#)-[Equation 8-6](#) needs to be supplemented by the constitutive model for the viscoelastic stress.

**Nonisothermal Flow, Viscoelastic Flow** also adds the heat equation which for a viscoelastic fluid is given by

$$\rho C_p \left( \frac{\partial T}{\partial t} + (\mathbf{u} \cdot \nabla) T \right) = -(\nabla \cdot \mathbf{q}) + \tau : \mathbf{S} + Q \quad (8-8)$$

where

- $C_p$  is the specific heat capacity at constant pressure (SI unit: J/(kg·K))
- $T$  is the absolute temperature (SI unit: K)
- $\mathbf{q}$  is the heat flux by conduction (SI unit: W/m<sup>2</sup>)
- $\tau$  is the extra stress tensor (SI unit: Pa) given by ([Equation 8-7](#))
- $\mathbf{S}$  is the strain-rate tensor (SI unit: 1/s)
- $Q$  contains heat sources other than viscous heating (SI unit: W/m<sup>3</sup>)

Generally speaking, for viscoelastic flows, the internal heat production term consists of an irreversible dissipation part and a reversible part ([Ref. 1](#)). The reversible part of the internal heat production is neglected in the [Equation 8-8](#). The irreversible part is given by

$$\tau : \mathbf{S} \quad (8-9)$$




For inelastic constitutive models, the irreversible losses are equal to the viscous dissipation.

### *References for the Nonisothermal Viscoelastic Flow Interfaces*

- 
1. G.W.M. Peters and F.P.T. Baaijens, “Modelling of non-isothermal viscoelastic flows,” *J. Nonnewton. Fluid. Mech.*, vol. 68, pp. 205–224, 1997.



# Chemorheology and Curing Kinetics

This chapter describes the Curing Reaction interface () which is found under the **Heat Transfer>Curing** branch () and the Curing Reaction Heating () multiphysics coupling.

# The Curing Reaction Interface

In this section:

- [The Curing Reaction Interface](#)
- [Domain, Boundary, and Global Nodes for the Curing Reaction Interface](#)
- [Curing Reaction Heating](#)





## Theory for the Curing Reaction Interface

---

### *The Curing Reaction Interface*

---

The **Curing Reaction (creq)** interface () is found under the **Heat Transfer>Curing** branch () when adding physics interfaces. This interface is intended for modeling the curing of polymers during the heat treatment.

When the Curing Reaction interface is added, three default nodes are added to the **Model Builder**: a **Curing Kinetics** node, an **Initial Values** node, and a **No Flux** node. Then, from the **Physics** toolbar, you can add other nodes that implement, for example, boundary conditions. You can also right-click **Curing Reaction** to select physics features from the context menu.

The interface supports simulation of curing reaction in 1D, 2D, and 3D as well as for axisymmetric components in 2D. The dependent variable is the conversion,  $cr$ .

#### 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 `creq`.



HEAT TRANSFER

Polymer curing is a temperature dependent process. Select the temperature field to use from the **Temperature** list.

ADVANCED SETTINGS

To display this section, click the **Show More Options** button (  ) and select **Advanced Physics Options** in the **Show More Options** dialog box. Normally these settings do not need to be changed.

*Pseudo Time Stepping*

The **Use pseudo time stepping for stationary equation form** option adds pseudo time derivatives to the equation when the **Stationary** equation form is used in order to speed up convergence. When selected, a CFL number expression should also be defined. For the default Automatic option, the local CFL number (from the Courant–Friedrichs–Lewy condition) is determined by a PID regulator. For more information, see Pseudo Time Stepping.

DISCRETIZATION

The Curing Reaction interface uses **Discontinuous Lagrange** functions with **Constant** element order.

DEPENDENT VARIABLES

The dependent variable is the conversion, *cr*. If required, the name of the dependent variable can be edited.

	<ul style="list-style-type: none"><li>• <a href="#">Domain, Boundary, and Global Nodes for the Curing Reaction Interface</a></li><li>• <a href="#">Theory for the Curing Reaction Interface</a></li></ul>
---	---

*Domain, Boundary, and Global Nodes for the Curing Reaction Interface*

The Curing Interface has these domain, boundary, pair, and global nodes (listed in alphabetical order), which 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).

- [Chemorheology](#)
- [Conversion](#)
- [Curing Kinetics](#)
- [No Flux](#)
- [Initial Values](#)
- [Open Boundary](#)
- [Symmetry](#)



For 2D axisymmetric components, COMSOL Multiphysics takes the axial symmetry boundaries (at  $r = 0$ ) into account and automatically adds an **Axial Symmetry** node to the component that is valid on the axial symmetry boundaries only.

## *Curing Kinetics*

The **Curing Kinetics** node defines equation to analyze the evolution of the conversion degree during curing reactions. Define the rate expressions as required.

### **CONVECTION**

To specify the velocity field, select the source of the **Velocity field**. For **User defined**, enter values or expressions for the velocity components in the input fields. This input option is always available. You can also select the velocity field solved for by a Fluid Flow interface added to the model component. These physics interfaces are available for selection in the **Velocity field** list.

### **CURING KINETICS**

Use this section to specify the curing reaction rate expression.

#### *Kinetic model*

The following kinetic models are available: n-th order, Sestak–Berggren, and Kamal—Sourour.

The following parameters are required for the **n-th order** model:

- **Polynomial order**  $n_r$ . The default is 1.
- **Frequency factor**  $k_0$ . The default is 1 1/s.
- **Activation energy**  $Q_0$ . The default is 0 J/mol.

The following parameters are required for the **Sestak–Berggren** model:

- **Polynomial order**  $n_r$ . The default is 1.
- **Polynomial order**  $m_r$ . The default is 0.
- **Frequency factor**  $k_0$ . The default is 1 1/s.
- **Activation energy**  $Q_0$ . The default is 0 J/mol.

The following parameters are required for the **Kamal–Sourour** model:

- **Polynomial order**  $n_r$ . The default is 1.
- **Polynomial order**  $m_r$ . The default is 0.
- **Frequency factor**  $k_{01}$ . The default is 1 1/s.
- **Activation energy**  $Q_{01}$ . The default is 0 J/mol.
- **Frequency factor**  $k_{02}$ . The default is 1 1/s.
- **Activation energy**  $Q_{02}$ . The default is 0 J/mol.

#### *Reaction rate limiter*

A diffusion factor  $f_d$  is introduced to account for the reaction rate decrease as the conversion reaches the critical value. Select a **Reaction rate limiter** model: **None**, **Diffusion factor**, and **Modified diffusion factor**.

- When the default, None is kept, the reaction limiter  $f_d$  is set to 1.
- For **Diffusion factor** enter values for these properties:
  - **Critical conversion**  $c_{rc}$ . The default is 1.
  - **Diffusion constant**  $C_d$ . The default is 500.
- For **Modified diffusion factor** enter values for these properties:
  - **Conversion limiter**  $c_{rf}$ . The default is 1.
  - **Diffusion constant**  $C_f$ . The default is 0.

#### **VOLUME FRACTION**

Use this section to specify the **Volume fraction** of the cured fluid,  $V_f$ . The default is 1.

#### *No Flux*

This node is the default boundary condition on exterior boundaries. It should be used on boundaries across which there is no conversion flux.

### *Conversion*

---

The **Conversion** node is used to specify the conversion at an exterior domain boundary.

### *Initial Values*

---

The **Initial Values** node is used to specify the initial conversion.

### *Open Boundary*

---

Use this node to set up conversion transport across the boundaries, where both convective inflow and outflow can occur. On the parts of the boundary where fluid flows into the domain, an exterior conversion is prescribed. The direction of the flow across the boundary is typically calculated by a fluid flow interface and is provided as an input in the **Curing Kinetics** node.

### *Symmetry*

---

The Symmetry node can be used to represent boundaries where the conversion is symmetric, which implies no conversion transport across the boundary.

### *Chemorheology*

---

Use the Chemorheology node to model changes in the polymer viscosity using the chemoviscosity model that relates the viscosity to the degree of cure (conversion) and temperature.

## **TRANSPORT PROPERTIES**

### *Viscosity factor*

Select **Castro–Macosko** or **Percolation model** to calculate the viscosity factor  $\alpha_c$ .

The following parameters are required for the **Castro–Macosko** model:

- **Gel point**  $c_{r_g}$ . The default is 1.
- **Model constant**  $c_1$ . The default is 0.
- **Model constant**  $c_2$ . The default is 0.

The following parameters are required for the **Percolation model**:

- **Gel point**  $c_{r_g}$ . The default is 1.
- **Percolation exponent**  $p$ . The default is 0.

#### *Compute effective transport properties*

From the **Chemorheology** node, it is possible to generate the viscosity for the cured material. Select **Compute effective transport properties** check box to select the viscosity model that describes the temperature, shear rate, and cure dependencies.

When **Compute effective transport properties** check box is selected, the following predefined viscosity models are available: **Newtonian**, **Power law**, and **Cross**.

The parameters for the selected model can either be specified by a **Fluid** material, or by a **User defined** expression

The following parameters are required for the **Newtonian**:

- Dynamic viscosity  $\mu$

The following parameters are required for the **Power law**:

- **Fluid consistency coefficient**  $m$
- **Flow behavior index**  $n$
- **Lower shear rate limit**  $\dot{\gamma}_{\min}$
- **Reference shear rate**  $\dot{\gamma}_{\text{ref}}$

For the **Cross model**, the following parameters are required:

- **Zero shear rate viscosity**  $\mu_0$
- **Infinite shear rate viscosity**  $\mu_{\text{inf}}$
- **Power index**  $n$
- **Critical stress**  $\tau_{\text{tr}}$


#### *Thermal Function*

The polymer viscosity has a strong dependency on the temperature. Thus, the apparent viscosity is modified using factor  $\alpha_T$ . The thermal effects are activated with five options: **None**, **Arrhenius**, **Williams–Landel–Ferry** (WLF), **Exponential**, and **User defined**.

- When the default, **None**, is kept, the shift function  $\alpha_T$  is set to unity and the viscosity is not modified.

- For **Arrhenius** enter values or expressions for these properties:
  - **Reference temperature**  $T_0$
  - **Activation energy**  $Q$
- For **Williams–Landel–Ferry** enter values or expressions for these properties:
  - **Calibration temperature**  $T_{WLF}$
  - **WLF constant 1**  $C_{1WLF}$
  - **WLF constant 2**  $C_{2WLF}$
- For **Exponential** enter values or expressions for these properties:
  - **Reference temperature**  $T_0$
  - **Temperature sensitivity**  $b$
- For **User defined** enter the expressions for  $\alpha_T$

#### Material Generation

**Create/Update Cured Material** (  ) option can be selected when **Compute effective transport properties** check box is selected. When you click it, a new material will be created under **Materials**. It contains the material parameters required for selected viscosity model. The name of this material is Cured Material.




For this type of Materials to work correctly, it is important that you do not edit the generated nodes manually. By clicking the **Create/Update Cured Material** button again, you can reset the generated node.



See also

- [Non-Newtonian Flow](#)
- [Materials](#) in the Materials chapter in the *COMSOL Multiphysics Reference Manual*.

#### Curing Reaction Heating

The curing reaction is exothermic. The Curing Reaction Heating multiphysics coupling (  ) adds the source term  $Q_{c_r}$  (SI unit:  $W/m^3$ ) to account for reaction heating in the heat equation:

$$\rho C_p \frac{\partial T}{\partial t} + \rho C_p \mathbf{u} \cdot \nabla T + \nabla \cdot \mathbf{q} = Q_{c_r} \quad (9-1)$$

The reaction heating is

$$Q_{c_r} = \rho H_{c_r} R_{c_r}$$

where  $R_{c_r}$  is the reaction rate,  $H_{c_r}$  is the reaction heat.



## Modeling Heat Transfer in the Polymer Flow Module

### SETTINGS

The **Label** is the default multiphysics coupling feature name.

The **Name** is used primarily as a scope prefix for variables defined by the coupling node. Refer to such variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different coupling nodes or 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** is `crh1`.

### DOMAIN SELECTION

When nodes are added from the context menu, you can select **All domains** (the default) or select **Manual** from the **Selection** list to choose specific domains.

### COUPLED INTERFACES

This section controls which individual interfaces are coupled by the current coupling feature. If a physics interface is deleted and then added to the model again, then in order to reestablish the coupling, you need to choose the correct physics interface again from the **Curing Reaction** or **Heat Transfer** interfaces lists.

### REACTION HEAT SOURCE

Enter a reaction heat  $H_{c_r}$  (SI unit: J/kg).

# Theory for the Curing Reaction Interface

The theory for the Curing Reaction interface is described in this section:

- [Kinetics Models](#)
- [Pseudo Time Stepping](#)
- [Chemorheological Models](#)
- [Reference for the Curing Reaction Interface](#)

## *Kinetics Models*

---

Curing is a process of forming the crosslinks that join polymer chains together. The term curing refers to the crosslinking of thermosetting resins, for example an unsaturated polyester and epoxy resin.

The evolution of the degree of cure or conversion  $c_r$  is described by the convection equation with a source term:

$$\frac{\partial c_r}{\partial t} + \mathbf{u} \cdot \nabla c_r = R_{c_r}$$

where  $R_{c_r}$  is the reaction rate.

One of the simplest models is the  $n$ th-order reaction

$$R_{c_r} = k_r(1 - c_r)^{n_r}$$

where  $n_r$  is the polynomial order and  $k_r$  is a temperature-dependent reaction rate constant that follows an Arrhenius equation

$$k_r = k_0 \exp\left(-\frac{Q}{RT}\right)$$

where  $Q$  is the activation energy and  $k_0$  is the frequency factor.

The Sestak–Berggren autocatalytic model describes a reaction that is slow initially, increases in rate as the reaction proceeds, before slowing down again as the reactants are depleted



$$R_{c_r} = k_r c_r^{m_r} (1 - c_r)^{n_r}$$

where  $m_r$  is the polynomial order.

Kamal–Sourour is a combination of the  $n$ th order and Sestak–Berggren models

$$R_{c_r} = (k_1 + k_2 c_r^{m_r}) (1 - c_r)^{n_r}$$

where  $k_1$  and  $k_2$  are temperature-dependent reaction rate constants

$$k_{1,2} = k_{0,2} \exp\left(-\frac{Q_{1,2}}{RT}\right)$$

where  $Q_{1,2}$  are the activation energies and  $k_{0,2}$  the frequency factors.

The curing reaction stops before the complete conversion is achieved. The diffusion factor is introduced in curing model to take into account the drastic decrease of the rate of crosslinking due to vitrification. The reaction rate is limited by multiplying  $R_{c_r}$  by the diffusion factor

$$f_d = \frac{1}{1 + \exp[C_d(c_r - c_{r_c})]}$$

or by the modified diffusion factor

$$f_d = \frac{2}{1 + \exp\left[\frac{c_r - c_{rf}}{C_f}\right]}$$

where  $c_{rf}$ ,  $c_{r_c}$ ,  $C_f$  and  $C_d$  are the model parameters.

### *Pseudo Time Stepping*

In order to improve the solution robustness, pseudo time stepping can be used for the Curing Reaction interface when solving a stationary model. Using pseudo time stepping, a fictitious time derivative term:

$$\frac{c_r - \text{nojac}(c_r)}{\Delta \tilde{t}}$$

is added to the left-hand side of the curing reaction equation. Here  $\Delta \tilde{t}$  is the pseudo time step. Since  $c_r - \text{nojac}(c_r)$  is always zero, this term does not affect the final

solution. It does, however, affect the discrete equation system and effectively transforms a nonlinear iteration into a time step of size  $\Delta\tilde{t}$ .



- See also [Pseudo Time Stepping](#) in the *COMSOL Multiphysics Reference Manual*.

## Chemorheological Models

Polymer melts are usually non-Newtonian fluids. For the inelastic non-Newtonian models, it is possible to define an apparent viscosity from a generalized Newtonian relationship between the deviatoric stress tensor and the strain-rate tensor

$$\mathbf{K} = 2\mu_{\text{app}}(\dot{\gamma}) \mathbf{S}$$

where  $\dot{\gamma} = \sqrt{2\mathbf{S}:\mathbf{S}}$  is the shear rate.

For the thermosets, the apparent viscosity depends both on the temperature and the degree of cure. The viscosity of the polymer decreases with the temperature. On the other hand, the raising temperature will increase the reaction rate, thus leading to increase in the conversion and viscosity

$$\mu_{\text{app}}(T, c_r, \dot{\gamma}) = \alpha_c(c_r)\mu_{\text{app}}(T, c_{r0}, \dot{\gamma})$$

The Castro–Macosko model is often used to describe the changes of the apparent viscosity with degree of conversion

$$\alpha_c = \left( \frac{c_{rg}}{c_{rg} - c_r} \right)^{c_1 + c_2 c_r} \quad (9-2)$$

where  $c_{rg}$  is the conversion at which the viscosity of the melt grows drastically and  $c_1$  and  $c_2$  are the model parameters.

Another often used model is a percolation model is

$$\alpha_c = \left( 1 - \frac{c_r}{c_{rg}} \right)^{-p_1} \quad (9-3)$$

where  $p$  is the percolation exponent.

Power law and Cross non-Newtonian inelastic models are available in the Curing Reaction interface to specify the shear-rate dependence.

### POWER LAW

The Power law model prescribes

$$\mu_{\text{app}} = m \left( \frac{\max(\dot{\gamma}, \dot{\gamma}_{\min})}{\dot{\gamma}_{\text{ref}}} \right)^{n-1} \quad (9-4)$$

where  $m, n$  are scalars that can be set to arbitrary values and  $\dot{\gamma}_{\text{ref}}$  denotes a reference shear rate where  $\dot{\gamma}_{\min}$  is a lower limit for the evaluation of the shear rate magnitude. The temperature and conversion dependent Power law viscosity becomes

$$\mu_{\text{app}} = m \alpha_c \alpha_T \left( \frac{\max(\dot{\gamma}, \dot{\gamma}_{\min})}{\dot{\gamma}_{\text{ref}}} \right)^{n-1} \quad (9-5)$$

### CROSS MODEL

Combining the Cross model for shear-rate dependence and thermal effects gives

$$\mu_{\text{app}} = \alpha_c \left[ \mu_{\infty} + \frac{\alpha_T \mu_0 - \mu_{\infty}}{1 + \left( \frac{\alpha_T \mu_0 \dot{\gamma}}{\tau_{\text{tr}}} \right)^{1-n}} \right] \quad (9-6)$$

where  $\tau_{\text{tr}}$  is the critical stress,  $\mu_0$  is the zero shear rate viscosity,  $\mu_{\infty}$  is the infinite shear-rate viscosity, and  $n$  is a dimensionless parameter.

### THERMAL EFFECTS

Several models are available to describe the temperature dependence.

#### *Arrhenius*

One commonly used thermal function is defined by the Arrhenius equation:

$$\alpha_T = \exp \left[ \frac{Q}{R} \left( \frac{1}{T} - \frac{1}{T_0} \right) \right]$$

where  $Q$  denotes the activation energy,  $R$  is the universal gas constant, and  $T$  and  $T_0$  are the temperature and reference temperature, respectively.

#### *Williams-Landel-Ferry*

Williams-Landel-Ferry model is defined as

$$\log(\alpha_T) = \frac{-C_{1\text{WLF}}(T - T_{\text{WLF}})}{C_{2\text{WLF}} + T - T_{\text{WLF}}} \quad (9-7)$$

where  $C_{1\text{WLF}}$  and  $C_{2\text{WLF}}$  are model constants, and  $T_{\text{WLF}}$  is the reference temperature.

#### *Exponential*

Exponential model is defined as

$$\alpha_T = \exp[-b(T - T_0)] \quad (9-8)$$

where  $b$  denotes a temperature sensitivity, and  $T_0$  is the reference temperature.

Viscosity factor calculated using [Equation 9-2](#) or [Equation 9-3](#) can be used together with the non-Newtonian inelastic models included in the fluid flow interfaces. The models are defined in the fluid flow interfaces and provide a complete model for the apparent viscosity that is temperature and shear-rate dependent.



- [Non-Newtonian Flow](#)
- 

#### *Reference for the Curing Reaction Interface*

---

I. A. Yousefi, P.G. Lafleur, and R. Gauvin, “Kinetic studies of thermoset cure reactions: A review,” *Polymer Composites*, vol. 18, no. 2, 1997.

## Parameter Estimation

Modeling of flow non-Newtonian fluids using rheological models involves an estimation of number of material parameters from the experiments. The material model parameter values can be calibrated to experimental data using nonlinear least-squares **Parameter Estimation** techniques. The Parameter Estimation functionality is available under **Optimization** in the **Physics** toolbar.

Three optimization solvers are available with the Polymer Flow Module in the Parameter Estimation study step: the derivative-free BOBYQA solver, the gradient-based IPOPT and Levenberg–Marquardt.

For further information about the **Parameter Estimation** functionality and the optimization solvers, see

- [Parameter Estimation](#) and [Global Least-Squares Objective](#) in the *Optimization Module User's Guide*.
- [The Parameter Estimation Study](#) in the *Optimization Module User's Guide*.



# I n d e x

- A**
  - absolute pressure 47, 154, 164, 172, 215
  - adsorption 300
  - Application Libraries window 22
  - application library examples
    - convection and diffusion 277
    - laminar flow 45
    - migration in electric field 277
    - rmspf interfaces 85
    - transport of diluted species 270
- B**
  - boundary nodes
    - Brinkman equations 209
    - free and porous media flow 219
    - level set 77, 234
    - phase field 240
    - rmspf interfaces 86
    - ternary phase field 246
    - transport of diluted species 272
  - boundary stress (node) 65
  - Brinkman equations 226
  - Brinkman equations interface 206
    - theory 226
- C**
  - Cahn–Hilliard equation 256, 262
  - CFL number
    - pseudo time stepping, and 85
    - settings 44, 209
  - chemical potential variable 260
  - common settings 20
  - contact angle (node) 88, 180
  - continuity (node) 237, 244
    - incompressible potential flow 81
  - convection 320
  - convective terms, diluted species 312
  - creeping flow 176
  - creeping flow (spf) interface 38
- D**
  - Darcy velocity 320
- documentation 21
- domain nodes
  - Brinkman equations 209
  - free and porous media flow 219
  - level set 77, 234
  - phase field 240
  - rmspf interfaces 86
  - ternary phase field 246
  - transport of diluted species 272
- E**
  - electrode-electrolyte interface coupling
    - (node)
      - transport of diluted species 292
  - emailing COMSOL 23
  - Equilibrium Reaction
    - theory for 310
  - equilibrium reaction (node) 289
  - external fluid interface (node) 178
- F**
  - Faraday’s law 316
  - flow continuity (node) 71
  - fluid flow
    - Brinkman equations theory 226
    - two-phase flow moving mesh theory
      - 195
  - fluid properties (node)
    - incompressible potential flow 77
  - fluid-fluid interface (node) 177
  - flux discontinuity (node) 284
    - transport of diluted species 284
  - free and porous media flow interface 217
    - theory 229
  - Freundlich exponent 326
  - frozen rotor 139
  - fully developed flow 59
- G**
  - general stress (boundary stress condition) 65

- geometry frames 196
  - Ginzburg–Landau equation 257
  - gravity 74
- H** Hygroscopic Swelling 303
- I**
  - inflow (node) 281
  - initial values (node)
    - Brinkman equations 214
    - free and porous media flow 224
    - level set 78, 236, 242
    - phase field 248
    - single-phase, laminar flow 52
    - spf interfaces 52
    - transport of diluted species 278
  - initializing functions 254, 259
  - inlet (node) 57
    - single-phase flow 57
    - ternary phase field 248–249
  - interface normal variable 255, 260
  - interior wall (node)
    - incompressible potential flow 81
    - spf interfaces 68
  - interior wetted wall (node) 244
  - internet resources 20
  - intrinsic volume averages 226
- K** knowledge base, COMSOL 23
- L**
  - laminar flow interface 39
  - laminar three-phase flow, phase field interface 170
  - laminar two-phase flow
    - moving mesh interface 175
  - laminar two-phase flow, level set (tpf) interface 152, 162
  - Langmuir constant 326
  - leaking wall, wall boundary condition 54, 225
  - level set functions, initializing 254, 259
  - level set interface 232–233
    - theory 182, 251
  - level set model (node) 234
  - line mass source (node)
    - fluid flow 73
    - species transport 287
  - line source
    - species transport 314
  - local
    - CFL number 44, 209
  - local CFL number 85
- M**
  - mass based concentrations (node) 278
  - mass conservation, level set equations 187
  - mass source (node)
    - Brinkman equations 213
  - material frames 196
  - mathematics, moving interfaces
    - level set 232–233
    - phase field 239, 245
    - theory 251, 256, 262
  - mesh frames 196
  - mixture properties (node) 246
  - moving interfaces 256, 262
  - MPH-files 22
  - multiphase flow
    - moving mesh theory 195
  - multiphase flow theory 182, 251, 256, 262
- N**
  - Navier–Stokes equations 182
  - Nernst–Einstein relation 276, 295, 299, 316
  - no flow (node)
    - level set 237
  - no flux (node) 280
  - no slip, interior wall boundary condition 68
  - no slip, wall boundary condition 53, 225
  - no viscous stress (open boundary) 64
  - nodes, common settings 20



- nonconservative formulations 187, 312
- normal stress, normal flow (boundary stress condition) 65
- O**
  - Ohm's law 317
  - open boundary (node)
    - incompressible potential flow 78
    - level set 237
    - single-phase flow 64
    - spf interfaces 64
    - transport of diluted species 288
  - outflow (node)
    - transport of diluted species 282
  - outlet (node) 60
    - single-phase flow 60
    - spf interfaces 60
    - ternary phase field 248
- P**
  - pair nodes
    - Brinkman equations 209
    - free and porous media flow 219
    - level set 77, 234
    - phase field 240
    - rmspf interfaces 86
    - ternary phase field 246
    - transport of diluted species 272
  - partially saturated porous media (node) 296
  - periodic condition (node)
    - transport of diluted species 285
  - periodic flow condition (node) 66
    - incompressible potential flow 79
  - phase field interface 239
    - theory 182, 256, 262
  - phase field model (node) 241
  - physics interfaces, common settings 20
  - point mass source (node)
    - fluid flow 73
    - species transport 288
  - point nodes
    - Brinkman equations 209
    - free and porous media flow 219
    - rmspf interfaces 86
    - transport of diluted species 272
  - point source
    - species transport 313
  - porous electrode coupling (node) 291
  - porous media and subsurface flow
    - Brinkman equations interface 206
    - free and porous media flow interface 217
    - theory, Brinkman equations 226
    - theory, free and porous media flow 229
  - porous media transport properties (node) 293, 296
  - Porous Medium (node)
    - Brinkman equations 210, 221
  - potential point constraint (node)
    - incompressible potential flow 79
  - pressure point constraint (node) 72
  - pseudo time stepping
    - settings 44, 209
- R**
  - Reacting Volume 279
  - reaction coefficients (node) 292
  - reactions (node)
    - transport of diluted species 279
  - retardation factor 326
  - rotating machinery (rmspf) interfaces
    - theory 137
  - rotating machinery, laminar flow (rmspf)
    - interface 84
- S**
  - selecting
    - heat transfer interfaces 332
    - single-phase flow interfaces 32
  - single-phase flow interface
    - laminar flow 39
  - sliding wall 56

- slip, interior wall boundary condition 68
- slip, wall boundary condition 54, 225
- spatial frames 196
- standard settings 20
- Stokes equations 38
- Stokes flow 38, 176
- stratified porous media 324
- superficial volume average, porous media 226
- superficial volume averages, porous media 320
- Supporting Electrolytes 316
- surface equilibrium reaction (node) 291
- surface tension force variable 260
- symmetry (node) 63
  - incompressible potential flow 79
  - spf interfaces 63
  - transport of diluted species 284

**T**

- technical support, COMSOL 23
- ternary phase field interface 245
- theory
  - Brinkman equations 226
  - free and porous media flow 229
  - level set 251
  - phase field 256, 262
  - rmspf interfaces 137
  - tpf interfaces 182
  - transport of diluted species in porous media interface 309
  - transport of diluted species interface 308
  - two-phase flow moving mesh 195
- thickness
  - fracture 306
- thin barrier (node)
  - level set 237
- thin diffusion barrier (node) 289
- Thin Impermeable Barrier 289

- tortuosity factors 322
- traction boundary conditions 65
- transport of diluted species in porous media interface 270
  - theory 309
- transport of diluted species interface 266
  - theory 308
- two-phase flow
  - moving mesh, theory 195

**V**

- variables
  - level set interface 255
  - phase field interface 260
- velocity (node)
  - incompressible potential flow 78
- volume averages 226
- volume force (node) 52
  - free and porous media flow 223
  - spf interfaces 52

**W**

- wall (node)
  - free and porous media flow 224
  - incompressible potential flow 78
  - single-phase flow 53
- websites, COMSOL 23
- wetted wall (boundary condition) 157
- wetted wall (node) 243, 249