COMSOL MULTIPHYSICS

CHEMICAL ENGINEERING MODULE

USER'S GUIDE

VERSION 3.4

COMSOL

How to contact COMSOL:

Benelux
COMSOL BV
Röntgenlaan 19
2719 DX Zoetermeer
The Netherlands
Phone: +31 (0) 79 363 4230
Fax: +31 (0) 79 361 4212
info@femlab.nl
www.femlab.nl

Denmark
COMSOL A/S
Diplomvej 376
2800 Kgs. Lyngby
Phone: +45 88 70 82 00
Fax: +45 88 70 80 90
info@comsol.dk
www.comsol.dk

Finland
COMSOL OY
Arabianranta 6
FIN-00560 Helsinki
Phone: +358 9 2510 400
Fax: +358 9 2510 4010
info@comsol.fi
www.comsol.fi

France
COMSOL France
WTC, 5 pl. Robert Schuman
F-38000 Grenoble
Phone: +33 (0)4 76 46 49 01
Fax: +33 (0)4 76 46 07 42
info@comsol.fr
www.comsol.fr

Germany
FEMLAB GmbH
Berliner Str. 4
D-37073 Göttingen
Phone: +49-551-99721-0
Fax: +49-551-99721-29
info@femlab.de
www.femlab.de

Italy
COMSOL S.r.l.
Via Vittorio Emanuele II, 22
25122 Brescia
Phone: +39-030-3793800
Fax: +39-030-3793899
info.it@comsol.com
www.it.comsol.com

Norway
COMSOL AS
Søndre gate 7
NO-7485 Trondheim
Phone: +47 73 84 24 00
Fax: +47 73 84 24 01
info@comsol.no
www.comsol.no

Sweden
COMSOL AB
Tegnérgatan 23
SE-111 40 Stockholm
Phone: +46 8 412 95 00
Fax: +46 8 412 95 10
info@comsol.se
www.comsol.se

Switzerland
FEMLAB GmbH
Technoparkstrasse 1
CH-8005 Zürich
Phone: +41 (0)44 445 2140
Fax: +41 (0)44 445 2141
info@femlab.ch
www.femlab.ch

United Kingdom
COMSOL Ltd.
UH Innovation Centre
College Lane
Hatfield
Hertfordshire AL10 9AB
Phone:+44-(0)-1707 284747
Fax: +44-(0)-1707 284746
info.uk@comsol.com
www.uk.comsol.com

United States
COMSOL, Inc.
1 New England Executive Park
Suite 350
Burlington, MA 01803
Phone: +1-781-273-3322
Fax: +1-781-273-6603

COMSOL, Inc.
10850 Wilshire Boulevard
Suite 800
Los Angeles, CA 90024
Phone: +1-310-441-4800
Fax: +1-310-441-0868

COMSOL, Inc.
744 Cowper Street
Palo Alto, CA 94301
Phone: +1-650-324-9935
Fax: +1-650-324-9936

info@comsol.com
www.comsol.com

For a complete list of international representatives, visit www.comsol.com/contact

Company home page
www.comsol.com

COMSOL user forums
www.comsol.com/support/forums

Chemical Engineering Module User's Guide

Version: October 2007 COMSOL 3.4

CONTENTS

Chapter 1: Introduction

Chapter 2: Overview of the Modeling Interface

Chapter 3: Modeling Guide

Chapter 4: Modeling Transport and Reactions

Chapter 5: Momentum Transport

Chapter 6: Energy Transport

Chapter 7: Mass Transport

Chapter 10: Glossary

1

Introduction

Modeling in chemical engineering has become ever more important due to the development of new processes and prototypes. The combination of experimental work with theoretical analyses in computer models has shown to accelerate understanding as well as decrease development costs for new processes. In the past, sophisticated modeling tools were privileges that only large companies could afford, where savings in bulk production justified the costs in computer software and specialized engineers. Today, modeling is a natural part of an engineer's education, and realistic models of advanced systems are feasible on a personal computer.

The Chemical Engineering Module 3.4 is an optional package that extends the COMSOL Multiphysics® modeling environment with customized user interfaces and functionality optimized for the analysis of transport phenomena coupled to chemical reactions. We have developed it for a wide audience including researchers, developers, teachers, and students.

The Chemical Engineering Module is tailor-made for the modeling of chemical systems with respect to fluid flow, chemical composition, and temperature as a function of space and time. It consists of a number of modeling interfaces for the modeling of laminar and turbulent fluid flows, multiphase flow, multicomponent mass transport, and energy transport in reacting systems. These interfaces contain all the steps in the modeling process: drawing the geometry, specifying the chemical and transport properties, meshing, solving, and postprocessing.

You can model unit operations equipment, separation processes, and chemical reactors. In addition to its application in traditional chemical industries, the Chemical Engineering Module is a popular tool for investigations in energy conversion (for example, fuel cells and combustion processes), applications such as microlaboratories in biotechnology, and in the development of sensors and equipment in analytical chemistry.

The Chemical Engineering Module is also widely used in education within chemical engineering, chemical reaction engineering, electrochemical engineering, biotechnology, and transport phenomena.

The notations and structure in this module were inspired by the book *Transport Phenomena* by Bird, Stewart, and Lightfoot. This work defines the field of transport phenomena and presents the first encounter with these types of processes for many engineering students. Its basic structure divides transport phenomena into momentum transport, energy transport, and mass transport, which is also the underlying organization of the Chemical Engineering Module.

In the field of chemical reaction engineering, we have taken inspiration from the most popular book in the field, *Elements of Chemical Reaction Engineering* by H. Scott Fogler. In fact, a special version of the Chemical Engineering Module, devoted to tubular reactors, appears in the 4th edition of this major reaction engineering text.

The documentation set for the Chemical Engineering Module consists of three books. Firstly, the *Chemical Engineering Module User's Guide*, reviews new features in the version 3.4 release and introduces the basic modeling process. The chapter "Overview of the Modeling Interface" presents the different application modes available in the module. The "Modeling Guide" chapter also discusses the modeling strategy for various test cases. Throughout the manual the models increase in complexity by coupling heat and mass transport with momentum transport.

Secondly, the *Chemical Engineering Module Model Library* starts with a section that has a number of examples treating reactor models and simulations of unit operations. These models are simple yet contain the basic couplings needed in advanced modeling of transport-reaction processes. A second section in that book adds advanced models of a more specific nature. The authors obtained these examples by reproducing models that have appeared in international scientific journals or from customers. Each model comes with a theoretical background as well as step-by-step instructions for creating the model from scratch. Furthermore, we supply these models as COMSOL

Multiphysics Model MPH-files that you can open them in COMSOL Multiphysics for further investigation. This way you can follow along with the printed discussion as well as use them as a starting point or template for your own modeling and applications. SI units are used to describe the relevant properties, parameters, and dimensions in most examples, but COMSOL Multiphysics also supports other unit systems.

Finally, the *Chemical Engineering Module Reference Guide* contains descriptions of how the model equations are formulated and implemented in COMSOL Multiphysics.

We hope the Chemical Engineering Module becomes valuable for your modeling work, and we are convinced that the effort you put into understanding COMSOL Multiphysics will be repaid several times over. If you have any feedback on the models in this set, please let us know. Likewise, we welcome any suggestions for additional models that we could add to the library, including those that you have developed. Feel free to contact us at info@comsol.com.

Typographical Conventions

All COMSOL manuals use a set of consistent typographical conventions that should make it easy for you to follow the discussion, realize what you can expect to see on the screen, and know which data you must enter into various data-entry fields. In particular, you should be aware of these conventions:

- A **boldface** font of the shown size and style indicates that the given word(s) appear exactly that way on the COMSOL graphical user interface (for toolbar buttons in the corresponding tooltip). For instance, we often refer to the **Model Navigator**, which is the window that appears when you start a new modeling session in COMSOL; the corresponding window on the screen has the title **Model Navigator**. As another example, the instructions might say to click the **Multiphysics** button, and the boldface font indicates that you can expect to see a button with that exact label on the COMSOL user interface.
- The names of other items on the graphical user interface that do not have direct labels contain a leading uppercase letter. For instance, we often refer to the Draw toolbar; this vertical bar containing many icons appears on the left side of the user interface during geometry modeling. However, nowhere on the screen will you see the term “Draw” referring to this toolbar (if it were on the screen, we would print it in this manual as the **Draw** menu).
- The symbol **>** indicates a menu item or an item in a folder in the **Model Navigator**. For example, **Physics>Equation System>Subdomain Settings** is equivalent to: On the **Physics** menu, point to **Equation System** and then click **Subdomain Settings**.

COMSOL Multiphysics>Heat Transfer>Conduction means: Open the **COMSOL Multiphysics** folder, open the **Heat Transfer** folder, and select **Conduction**.

- A `Code` (monospace) font indicates keyboard entries in the user interface. You might see an instruction such as “Type `1.25` in the **Current density** edit field.” The monospace font also indicates COMSOL Script codes.
- An *italic* font indicates the introduction of important terminology. Expect to find an explanation in the same paragraph or in the Glossary. The names of books in the COMSOL documentation set also appear using an italic font.

2

Overview of the Modeling Interface

The Chemical Engineering Module consists of a number of modeling interfaces, called application modes, which form the backbone of the module. These are based on the equations for momentum transport, energy transport, and mass transport. This chapter gives an overview of these application modes and the modeling interfaces with predefined multiphysics couplings between application modes.

The Application Modes and Predefined Multiphysics Couplings

Under the *Chemical Engineering Module* folder in the *Model Navigator*, you find folders with application modes for *Momentum Transport*, *Energy Transport*, and *Mass Transport*. Also under this folder are the *Fluid With Variable Density*, *Fluid-Chemical Reactions Interactions*, and the *Rotating Machinery* folders, which contain predefined couplings of some of the application modes in the transport folders.

The following tables provide complete lists of the application modes and the predefined multiphysics couplings in each folder.

TABLE 2-1: THE MOMENTUM TRANSPORT APPLICATION MODES

APPLICATION MODE FOLDER	APPLICATION MODE	DESCRIPTION
Laminar Flow		
	Incompressible Navier-Stokes Equations	General momentum balance and continuity equations for fluids with constant density.
	Non-Newtonian Flow	Momentum balance and continuity equations with Carreau, Power Law, and user-defined rheological models.
Turbulent Flow		
	k-ε Turbulence Model	Standard k-ε turbulence model with lift-off wall functions.
	k-ω Turbulence Model	k-ω turbulence model according to Wilcox with lift-off wall functions.
Porous Media Flow		
	Brinkman Equations	Brinkman's extension of Darcy's law that includes viscous momentum transport.
	Darcy's Law	Darcy's law combined with the continuity equation for fluids with constant or variable density.
Multiphase Flow		
	Bubbly Flow, Laminar	Two-phase laminar flow for gas bubbles in liquids.

TABLE 2-1: THE MOMENTUM TRANSPORT APPLICATION MODES

APPLICATION MODE FOLDER	APPLICATION MODE	DESCRIPTION
	Bubbly Flow, Turbulent	Two-phase flow for gas bubbles in liquids combined with a k-ε turbulence model.
	Level Set	Tracking of a moving interface within a given velocity field using the level set method.
	Level Set Two-Phase Flow, Laminar	Momentum balance and continuity equations coupled to a level set equation for two-phase laminar flow.
	Level Set Two-Phase Flow, Turbulent	Momentum balance and continuity equations coupled to a level set equation for two-phase flow combined with a k-ε turbulence model.
	Mixture Model, Laminar	Two-phase flow for solid particles in liquids or a dispersed liquid in a continuous liquid phase.
	Mixture Model, Turbulent	Two-phase flow for solid particles in liquids or a dispersed liquid in a continuous liquid phase combined with a k-ε turbulence model.

The corresponding contents in the application mode folder for *Energy Transport* are tabulated below.

TABLE 2-2: THE ENERGY TRANSPORT APPLICATION MODES

APPLICATION MODE	DESCRIPTION
Conduction	Energy balance with transport by heat conduction.
Conduction and Convection	Energy balance with transport by conduction and convection.

The following table lists the application modes in the *Mass Transport* folder.

TABLE 2-3: THE MASS TRANSPORT APPLICATION MODES

APPLICATION MODE	DESCRIPTION
Diffusion	Material balance and transport by diffusion in a solvent.
Convection and Diffusion	Material balance and transport by diffusion and convection for species in dilute solutions.

TABLE 2-3: THE MASS TRANSPORT APPLICATION MODES

APPLICATION MODE	DESCRIPTION
Maxwell-Stefan Convection and Diffusion	Material balance and transport through diffusion and convection in multicomponent solutions.
Nernst-Planck	Material balance and transport by diffusion, convection, and migration of ions in dilute solutions.
Nernst-Planck with Electroneutrality Condition	Material balance and transport by diffusion, convection, and migration of ions in dilute solutions combined with the electroneutrality condition.
Electrokinetic Flow	Material balance and transport by diffusion, convection, and migration of ions in dilute solutions.

The above application modes are available in 2D, 2D axisymmetry, and 3D. The *Energy Transport* and *Mass Transport* application modes are also available in 1D and 1D axisymmetry. All application modes are available for both stationary and time-dependent analyses.

In 1D and 2D, you also find the *Energy Transport* and *Mass Transport* application modes in the *Pseudo 2D* and *Pseudo 3D* folders. These so-called *Pseudo* application modes neglect the transport of both energy and mass through conduction and diffusion, respectively, in the direction of the flow. Furthermore, the convective transport of energy and mass takes place only in the axial direction and at steady state. This makes it possible to use the time variable, t, as a space coordinate to move along the axial direction in, for example, a tubular reactor.

You can freely combine the application modes above to model, for example, reacting systems and nonisothermal flow. Alternatively, you can use predefined couplings in the *Flow With Variable Density* folder, the *Fluid-Chemical Reactions Interaction* folder and the *Rotating Machinery* folder. The following tables gives an overview of the predefined couplings available in these folders.

TABLE 2-4: THE PREDEFINED COUPLINGS IN THE FLOW WITH VARIABLE DENSITY FOLDER

APPLICATION MODE FOLDER	PREDEFINED COUPLING OR APPLICATION MODE	DESCRIPTION
Non-Isothermal Flow		For this group of predefined couplings, the fluid density varies with temperature and pressure. The momentum and continuity equations are coupled to an energy transport application mode.
	Laminar Flow	Laminar nonisothermal flow with variable density.
	k-ε Turbulence Model	Nonisothermal turbulent flow with variable density using a k-ε turbulence model. Requires the use of temperature boundary layer approximations at walls with given temperatures.
	k-ω Turbulence Model	Nonisothermal turbulent flow with variable density using a k-ω turbulence model, according to Wilcox, and lift-off wall functions. Requires the use of temperature boundary layer approximations at walls with given temperatures.
Weakly Compressible Momentum Transport		This group contains application modes accounting for variable fluid density according to a user-defined function. There are no predefined couplings to energy nor transport application modes.
	Weakly Compressible Navier-Stokes	Laminar flow with variable density.
	k-ε Turbulence Model	Turbulent flow with variable density using a k-ε turbulence model.
	k-ω Turbulence Model	Turbulent flow with variable density using a k-ω turbulence model, according to Wilcox, and lift-off wall functions.

TABLE 2-5: THE PREDEFINED COUPLING IN THE FLUID-CHEMICAL REACTIONS INTERACTION FOLDER

PREDEFINED COUPLING	DESCRIPTION
Reacting Flow	Predefined coupling for reacting flows. It automatically couples the Incompressible Navier-Stokes equations with a mass transport equation.

TABLE 2-6: THE PREDEFINED COUPLING IN THE ROTATING MACHINERY FOLDER FOLDER

PREDEFINED COUPLING	DESCRIPTION
Rotating Navier-Stokes	Predefined combination of application modes for rotating fluid devices. Combines an Incompressible Navier-Stokes with a Moving Mesh (ALE) application mode for modeling of rotating fluid applications such as stirrers.

The above tables give you an overview of the application modes and the predefined couplings interfaces. In addition to these types of modeling interfaces, you also have access to the PDE modes, in which you can freely combine other multiphysics models with the above-mentioned application modes.

3

Modeling Guide

The first section in this chapter gives an overview of the modeling process in the Chemical Engineering Module, from the selection of application modes to the postprocessing of results.

The second section contains some important guidelines for the model set up and verification steps. This section also discusses some common issues in the modeling of transport-reaction processes.

The last section includes an exercise where you can apply the guidelines introduced in this chapter.

The Modeling Process

The modeling process in the Chemical Engineering Module consists of six main steps:

1. Selecting the appropriate application mode or predefined coupling in the *Model Navigator*.
2. Drawing or importing the model geometry in the *Draw Mode*.
3. Setting up the subdomain equations and boundary conditions in the *Physics Mode*.
4. Meshing in the *Mesh Mode*.
5. Solving in the *Solve Mode*.
6. Postprocessing in the *Postprocessing Mode*.

The steps listed above are all available in the graphical user interface. Once you have defined a model, you can go back and make changes in input data, equations, boundary conditions, and mesh. You can also restart the solver, for example, using the existing solution as initial condition or initial guess. You can even alter the geometry—the equations and boundary conditions are still available through the associative geometry feature. By adding an application mode, you can account for a phenomenon not previously described in a model. To do this, simply go back to the **Model Navigator** through the **Multiphysics** menu. This action still retains the existing geometry, equations, boundary conditions, and current solution, which you can build upon for further development of the model.

Preparing for Modeling

The modeling of transport-reaction processes usually involves highly nonlinear terms in the equations, which can have their origin in the reaction kinetics, in the convective term in the transport equations, and in the strong coupling between different transport phenomena. These nonlinearities result in specific requirements for the preparation of a model and its setup.

Input Data

The collection and verification of input data to the model is critical when working with nonlinear models because it can be difficult to even get a starting solution with incorrect data.

It is recommended that you compare different sources of data before you start modeling. If possible, also check the data's dependency of the modeled variables, for example, the dependence of a diffusion coefficient or a rate constant on temperature.

You should also be aware of the range of operating conditions in your model, so that you do not use data outside of its validity range. This can be difficult to achieve because you may not know the range of operating conditions when you first set up a model.

Once you have a good grasp of your input data, you can proceed to the model specification.

Decoupling, Initial Guesses, and Initial Conditions in Transport and Reaction Processes

The strong couplings in transport-reaction processes can lead to difficulties in reaching convergence unless you have good initial guesses for the problem.

If possible, initially plan to run a decoupled model of the different transport phenomena. This not only helps in the detection of potential errors in input data but can also result in good initial guesses for solving the nonlinear model of the fully coupled problem later on. For example, when modeling a reactor, you might set up an isothermal model as the first step. Once this model works properly, you can add the effects of a nonisothermal operation and restart the solver from the existing isothermal solution.

The nonlinearities in transport-reaction models have implications for the definition of the model's initial conditions. It is often important that the initial conditions are in agreement with the boundary conditions at $t = 0$. For example, if you define a zero concentration as an initial condition for a species in a domain and at the same time set a nonzero concentration at a boundary, this can result in large problems in reaching convergence for a highly nonlinear model. It is therefore good practice to thoroughly consider the definition of initial and boundary conditions before setting up a model.

Space Dimensions

When setting up the first model of a transport-reaction system, initially consider using as few space dimensions as possible. For example, you can start by setting up a reactor model in 0D (a space-independent model) to check the validity of input data and reaction kinetics. The sister product to the Chemical Engineering Module, the COMSOL Reaction Engineering Lab®, can be of great help in setting up complex reaction kinetics models for nonisothermal systems in space-independent models. It also includes an export function that automatically sets up the corresponding space-dependent model in the Chemical Engineering Module.

If a 0D model is completely irrelevant for your process, initially define the model in 1D, 2D, or 2D axisymmetry before taking the step to 3D. A 1D or 2D model has several advantages: it takes a shorter time to run, it requires a comparably small amount of computer memory, and it is easier to verify and validate because the results are easier to generate and interpret. Moreover, you always verify a 1D or 2D model using mesh convergence tests and other tests that might be very costly in 3D.

In this context, do not forget the so-called *Pseudo* application modes for mass and energy transport in tubular reactors at steady state. These application modes use the time variable, t, to simulate the axial direction of the reactor.

Selecting Application Modes and Predefined Couplings

Note that you can also follow the decoupling modeling strategy described earlier in this chapter by starting with a predefined coupling that you initially decouple in the *Solver Manager* in the graphical user interface. This means that you can start with the predefined coupling interfaces also when you initially plan to solve the decoupled problem. See "The Solver Manager and Solver Scripting" on page 401 in the *COMSOL Multiphysics User's Guide* for more information about the Solver Manager.

In addition to the predefined couplings, you have to be aware of the validity of the equations in the application modes before you start setting up a model. For example, whether the flow regime is turbulent or laminar decides the selection of application modes. Note that even if you, for example, mistakenly start with a laminar flow model for a turbulent flow regime, you can easily switch to a turbulence model without having to redefine the model.

Another issue that can be difficult to determine is the correct way to describe mass transport. Should you treat your solution as infinitely dilute or should you use the full Maxwell-Stefan multicomponent equations found in the corresponding application mode? As a rule of thumb, treat most aqueous solutions as infinitely dilute solutions, whereas solutions using air as the solvent require the full Maxwell-Stefan equations.

In energy balances, the selection of the appropriate material transport model (infinitely dilute solutions or Maxwell-Stefan multicomponent systems) is also of great importance. In a dilute solution, the heat capacity of the solvent dominates the fluid. For a Maxwell-Stefan multicomponent system, on the other hand, you have to account for variations in composition for this property.

Quick Start Model

Introduction

In the Chemical Engineering Module you readily set up and solve space- and time-dependent chemical engineering problems typically involving momentum, energy, and mass transport. Start by defining the processes occurring in a system by selecting one or more of the available *application modes*. The application modes are the graphical interfaces for the transport-reaction equations. Through the application mode's dialog boxes you associate the transport-reaction equations with your specific system geometry and material properties. After the geometry has been discretized by a mesh, you are ready to solve the problem.

The modeling steps outlined above will be illustrated as you go through this example. The example also provides some tips on modeling preparations and on how to solve coupled problems.

Model Definition

In this model you investigate the unimolecular decomposition of a chemical passing through a parallel plate reactor. The heat sensitive compound is present in a water solution. After entering the reactor, the liquid first experiences expansion, due to a step in the bottom plate. Before exiting, the fluid also passes a heated cylinder.

The full 3D representation of the reactor geometry is given in Figure 3-1.

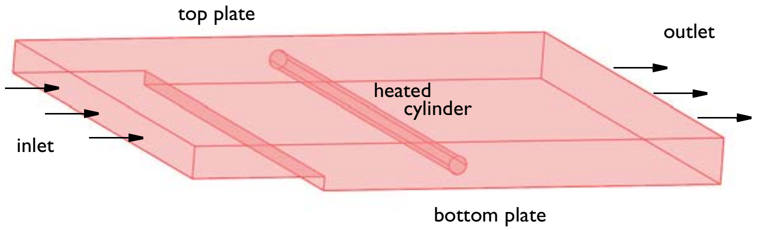


Figure 3-1: 3D geometry of a parallel plate reactor. The reacting fluid is heated as it passes the cylinder.

The short inlet section of the reactor is considerably wider than it is high. With such a geometry, it is reasonable to assume that the laminar flow develops a parabolic velocity profile between the top and bottom plate. At the same time, the velocity between the side walls is expected to be close to constant (Ref. 1). As a consequence, you can reduce the modeling domain to 2D without dramatically reducing the validity of the simulation (see Figure 3-2).

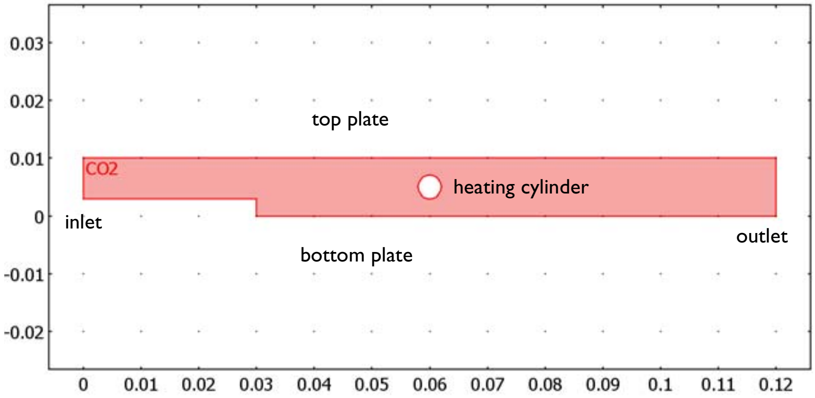


Figure 3-2: Neglecting edge effects, the modeling geometry can be reduced to 2D.

CHEMISTRY

A heat sensitive chemical (A) undergoes thermal decomposition into fragments (F) according to the following unimolecular reaction:

$$\mathrm{A} \xrightarrow{k} \mathrm{F}$$

The reaction rate (mol/(m^3·s)) is given by:

$$\mathrm{rate} = kc_A$$

where rate constant k (s^{-1}) is temperature dependent according to the Arrhenius equation:

$$k = A\exp\left(-\frac{E}{R_g T}\right) \tag{3-1}$$

In Equation 3-1, A is the frequency factor ($1 \cdot 10^{10}$ 1/s), E the activation energy ($72 \cdot 10^3$ J/mol), R_g the gas constant (8.314 J/(mol·K)), and T the temperature (K).

In addition, the decomposition reaction is exothermic, and the rate of energy expelled is given by:

$$Q = \mathrm{rate} \cdot H$$

where H is the heat of reaction (100 kJ/mol).

The conversion of species A in the reactor is a function of the residence time, that is, dependent on the detailed fluid flow. Furthermore, the decomposition is influenced by the temperature distribution. A coupled system of transport equations thus describes reactor.

MOMENTUM TRANSPORT

In this model the fluid flow is described by the incompressible Navier-Stokes equations:

$$\begin{gathered} \rho\frac{\partial \mathbf{u}}{\partial t} - \nabla \cdot \eta(\nabla \mathbf{u} + (\nabla \mathbf{u})^T) + \rho \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p = \mathbf{F} \\ \nabla \cdot \mathbf{u} = 0 \end{gathered} \tag{3-2}$$

Here, η denotes the dynamic viscosity (Ns/m^2), $\mathbf{u}$ the velocity (m/s), ρ the density of the fluid (kg/m^3), p the pressure (Pa), and $\mathbf{F}$ is a body force term (N/m^3).

Equation 3-2 describes the flow of incompressible Newtonian fluids, and you would typically use this formulation when the flow regime is laminar and the fluid density is near constant. The applicability of these assumptions is verified in the section "The Flow Regime" on page 21.

Apart from the domain equations you also need to select proper boundary conditions. At the inlet you specify a velocity vector normal to the boundary:

$$\mathbf{u} \cdot \mathbf{n} = u_0 \tag{3-3}$$

At the outlet boundary you specify a pressure:

$$p = p_0 \tag{3-4}$$

Finally, at the surfaces of the reactor plates and the heating cylinder you set the velocity to zero, that is, a no-slip boundary condition:

$$\mathbf{u} = \mathbf{0} \tag{3-5}$$

By selecting the Incompressible Navier-Stokes application mode you can easily associate the momentum balance (Equation 3-2) and boundary conditions (Equation 3-3 to Equation 3-5) with your modeling geometry.

ENERGY TRANSPORT

The energy balance equation applied to the reactor domain considers heat transfer through convection and conduction:

$$\rho C_p \frac{\partial T}{\partial t} + \nabla \cdot (-k\nabla T) + \rho C_p \mathbf{u} \cdot \nabla T = Q \qquad (3\text{-}6)$$

In Equation 3-6, C_p denotes the specific heat capacity (J/(kg·K)), k is the thermal conductivity (W/(m·K)), and Q is a sink or source term (W/m^3).

At the inlet and at the surface of the heating cylinder you set a temperature boundary condition:

$$T = T_0 \qquad (3\text{-}7)$$

$$T = T_{\mathrm{cyl}} \qquad (3\text{-}8)$$

At the outlet you set a Convective flux boundary condition. This assumes that all energy passing through this boundary does so by means of convective transport. Equivalently this means that the heat flux due to conduction across the boundary is zero

$$\mathbf{q}_{\mathrm{cond}} \cdot \mathbf{n} = -k\nabla T \cdot \mathbf{n} = 0 \qquad (3\text{-}9)$$

so that the resulting equation for the total heat flux becomes

$$\mathbf{q} \cdot \mathbf{n} = \rho C_p T \mathbf{u} \cdot \mathbf{n} \qquad (3\text{-}10)$$

This is a useful boundary condition, particularly in convection-dominated energy balances where the outlet temperature is unknown.

Finally, assume that no energy is transported across the reactor plates, that is, a thermal insulation boundary condition

$$\mathbf{q} \cdot \mathbf{n} = 0 \qquad (3\text{-}11)$$

Using the Convection and Conduction application mode, you can associate the energy balance (Equation 3-6) and boundary conditions (Equation 3-7 to Equation 3-11) with the modeling geometry.

MASS TRANSPORT

The mass transfer in the reactor domain is given by the convection and diffusion equation:

$$\frac{\partial c_i}{\partial t} + \nabla \cdot (-D_i \nabla c_i + c_i \mathbf{u}) = R_i \qquad (3\text{-}12)$$

where D_i denotes its diffusion coefficient ($\mathrm{m^2/s}$), and R_i denotes the reaction term ($\mathrm{mol/(m^3{\cdot}s)}$).

Equation 3-12 assumes that the species i is diluted in a solvent.

For the boundary conditions, specify the concentration of A at the inlet

$$c_i = c_{i,0} \qquad (3\text{-}13)$$

At the outlet, specify that the mass flow through the boundary is convection dominated. This assumes that any mass flux due to diffusion across this boundary is zero

$$\mathbf{n} \cdot (-D_i \nabla c_i) = 0 \qquad (3\text{-}14)$$

so that

$$\mathbf{N}_i \cdot \mathbf{n} = c_i \mathbf{u} \cdot \mathbf{n} \qquad (3\text{-}15)$$

Finally, at the surfaces of the reactor plates and the heating cylinder, assume that no mass is transported across the boundaries, that is, an insulation boundary condition:

$$\mathbf{N}_i \cdot \mathbf{n} = 0 \qquad (3\text{-}16)$$

By selecting the Convection and Diffusion application mode you can easily associate the mass balance (Equation 3-12) and boundary conditions (Equation 3-13 to Equation 3-16) with the modeling geometry.

PREPARING FOR MODELING

Before you can start modeling you need to gather the physical data that characterize your reacting flow. For instance, flow modeling requires you to supply the fluid density and viscosity. Mass transport requires knowledge of diffusivities and the reaction kinetics.

Another part of the preparations involves selecting the appropriate application modes and investigating the couplings between different transport equations.

Transport Properties

The term transport properties refers to the physical properties occurring in the transport equations (see the previous section). The momentum and heat transfer equations (Equation 3-2 and Equation 3-6) have *fluid-specific* transport properties:

- Viscosity (η)
- Density (ρ)
- Thermal conductivity (k)
- Heat capacity (C_p)

The mass transport equation (Equation 3-12) contains *species-specific* transport properties:

- Diffusivities (D_i)

You must enter appropriate values of the transport properties into the application modes to ensure accurate simulation results. In the present example, water with the dissolved compound A enters the reactor at 300 K. Because water is the solvent, you can assume that its physical properties are representative for the entire fluid. The warmest part of the reactor is held at 325 K. Table 3-1 lists the transport properties of water as well as the diffusivity of A in water at 300 K and 325 K.

TABLE 3-1: PHYSICAL PROPERTIES OF LIQUID WATER

PROPERTY	AT 300 K	AT 325 K
Density (kg/m^3)	997	987
Viscosity (Ns/m^2)	$8.5\cdot10^{-4}$	$5.3\cdot10^{-4}$
Thermal conductivity (W/(m·K))	0.62	0.66
Heat capacity (J/(kg·K))	4180	4182
Diffusivity (m^2/s)	$2.0\cdot10^{-9}$	$2.0\cdot10^{-9}$

Initially, constant values of the transport properties are used in the model. At a later stage you make use of the built-in Materials/Coefficients Library in COMSOL Multiphysics to load the temperature-dependent transport properties of water into the model. This increases the accuracy of the model but also the level of coupling in the equation system.

The Flow Regime

The Reynolds number indicates whether a flow is in the laminar or turbulent regime:

$$\mathrm{Re} = \frac{\rho u d}{\eta} \tag{3-17}$$

As a rule of thumb, a Reynolds value between of 2000 and 2500 marks the transition from stable streamlines to stable turbulent flow. It is always good practice to evaluate the Reynolds number related to the specific flow conditions of the model, because its magnitude guides you to choose the appropriate flow model and corresponding application mode.

In the present example, you can evaluate the Reynolds number using values from Table 3-1 and setting the velocity to $5{\cdot}10^{-4}$ m/s and the characteristic length to 0.007 m:

$$\mathrm{Re} = \frac{997 \cdot 5 \cdot 10^{-4} \cdot 0.007}{8.5 \cdot 10^{-4}} = 4 \tag{3-18}$$

Calculating the Reynolds number at 325 K produces a near identical result.

The Reynolds numbers are well within the limits of the laminar flow regime. Furthermore, considering that the moderate heating induced by the cylinder in the reactor induces only minor variations of the fluid density, the Incompressible Navier-Stokes application mode is appropriate to model the flow.

Dilute or Concentrated Mixtures

When modeling mass transport, it is advisable to discriminate between dilute and concentrated mixtures. For dilute mixtures, Fick's Law is adequate to describe the diffusional transport. Furthermore, the transport properties of the fluid can be assumed to be those of the solvent. For concentrated mixtures, on the other hand, mass transport should include Maxwell-Stefan multicomponent diffusion. Also, the transport properties of the fluid then depends of the mixture composition.

In COMSOL Multiphysics, the Convection and Diffusion application mode is appropriate for dilute mixtures, while the Maxwell-Stefan Diffusion and Convection application mode is recommended for concentrated mixtures.

APPLICATION MODE	DIFFUSION MODEL	FLUID TRANSPORT PROPERTIES
Convection and Diffusion	Fick's Law	Solvent
Maxwell-Stefan Diffusion and Convection	Maxwell-Stefan Diffusion	Composition dependent

As a rule of thumb, you can consider concentrations of up to 10 mol% of a solute in a solvent as a dilute mixture.

In the example at hand, the compound A is dissolved in water at concentration of 1000 mol/m^3. As the concentration of pure water is 55,500 mol/m^3, the molar fraction of A is approximately 2%. Because the mixture is dilute, it is appropriate to select the Convection and Diffusion application mode for mass transport and to select the transport properties of water as representative values for the mixture.

Solving Coupled Models

As noted previously, the chemistry occurring in the reactor depends both on the fluid flow and the temperature distribution in the reactor. More explicitly, the mass transport equation

$$\frac{\partial c_i}{\partial t} + \nabla \cdot (-D_i \nabla c_i + c_i \mathbf{u}) = R_i \tag{3-19}$$

depends on the velocity vector, **u**, which is solved for in the momentum transfer equation

$$\begin{gathered} \rho \frac{\partial \mathbf{u}}{\partial t} - \nabla \cdot \eta (\nabla \mathbf{u} + (\nabla \mathbf{u})^T) + \rho \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p = \mathbf{F} \\ \nabla \cdot \mathbf{u} = 0 \end{gathered} \tag{3-20}$$

Furthermore, the source term R_i in Equation 3-19 is a function of the temperature, which is the dependent variable of the energy transport equation

$$\rho C_p \frac{\partial T}{\partial t} + \nabla \cdot (-k \nabla T) + \rho C_p \mathbf{u} \cdot \nabla T = Q \tag{3-21}$$

When attempting to solve a coupled system of equations such as the one illustrated above, it is often a good idea analyze the couplings involved and the approach the solution in a stepwise fashion.

In the current example, you first assume that the transport properties that are independent of the temperature of the reacting liquid. Furthermore, the heat of reaction, Q, is neglected. This leads to one-way couplings between the transport equations:

- The momentum transport is independent of the energy and mass transport
- The energy transport depends only on the momentum transport
- The mass transport depends on both the momentum transport and the energy transport

This structure suggests that it is possible to solve the problem sequentially in the following order: momentum transport, energy transport, and mass transport.

Consider an further advancement of the model, where the fluid density, viscosity, and thermal conductivity vary with temperature. Also assume that a non-negligible heat of reaction is taken into account. This leads to a fully coupled problem:

- The momentum transport depends on the energy transport
- The energy transport depends on both momentum and mass transport
- The mass transport depends on both the momentum transport and the energy transport

In this case you must solve the equations describing all transport phenomena simultaneously. In some instances this might only be feasible if a good initial guess is provided to the software solvers. Again, a sequential approach is encouraged, where couplings are introduced one at a time to generate more and more accurate solutions.

Results

Figure 3-3 shows the velocity field in the reactor domain along with arrows indicating the velocity magnitude.

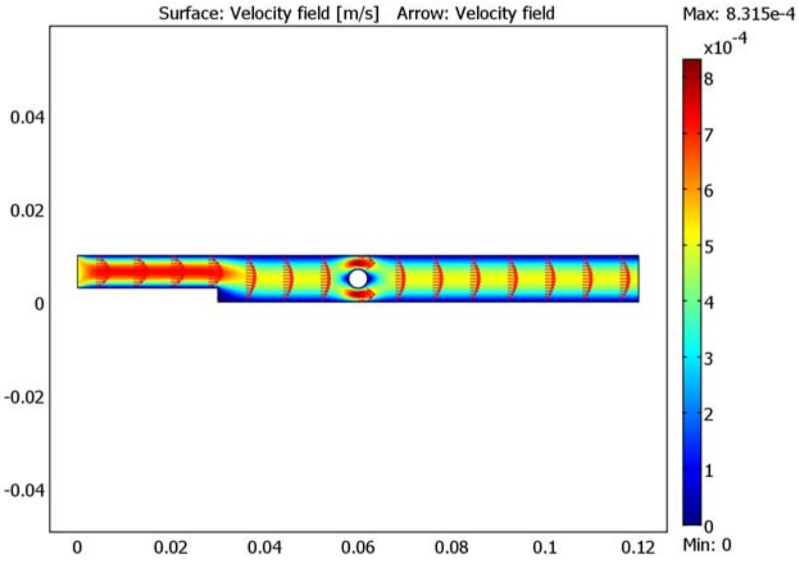


Figure 3-3: Velocity field (m/s) in the reactor.

The fluid cross-sectional area increases at the step and decreases at the cylinder, leading to the corresponding decrease and increase in fluid velocity. Recirculation zones appear after the step and the cylinder.

The water solution enters the reactor at a temperature of 300 K and is heated as it passes the cylinder (325 K). Figure 3-4 shows the temperature distribution in the reactor domain at steady state.

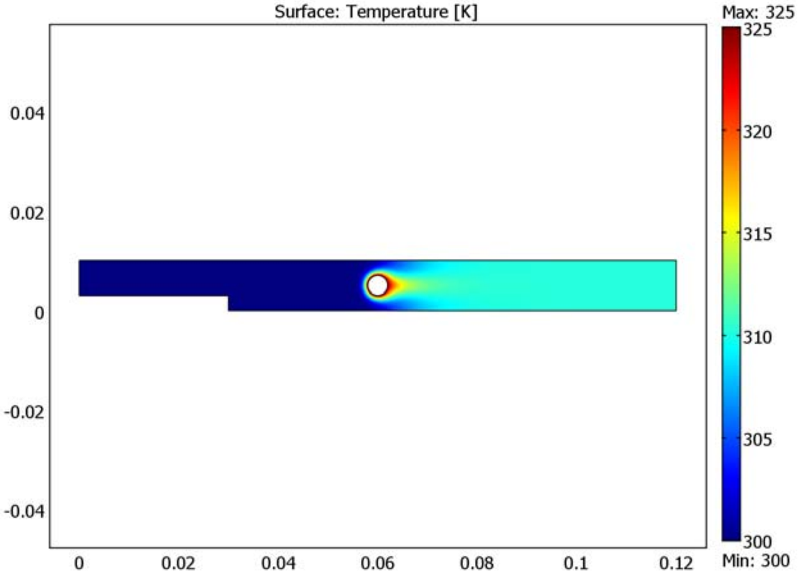


Figure 3-4: A water solution enters the reactor at 300 K and is heated by a cylinder kept at 325 K.

At the reactor inlet, the concentration of A is 1000 mol/m^3. Figure 3-5 shows the concentration A as the compound undergoes decomposition.

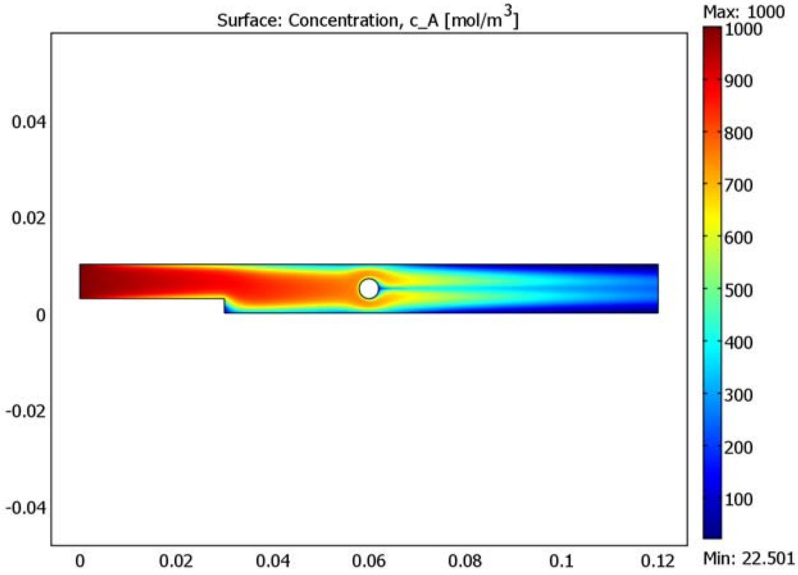


Figure 3-5: Concentration of A (mol/m^3) as function of position in the reactor.

These plots make it possible to identify some general trends. It is clear that decomposition occurs mainly after the liquid has been heated by the cylinder. In the first half of the reactor, where the temperature is relatively low, decomposition is still fairly advanced near the wall and after the step. This is due to the longer residence times in these areas. In the second part of the reactor, where heating takes place, regions with relatively high A concentration are visible. This also makes physical sense because the water velocity is relatively high.

When the model accounts for the heat of reaction, the temperature distribution in the entire reactor is affected. As shown in Figure 3-6, the maximum fluid temperature now exceeds the temperature of the heating cylinder. Furthermore, the water temperature is higher than 300 K also in the region between the inlet and the cylinder.

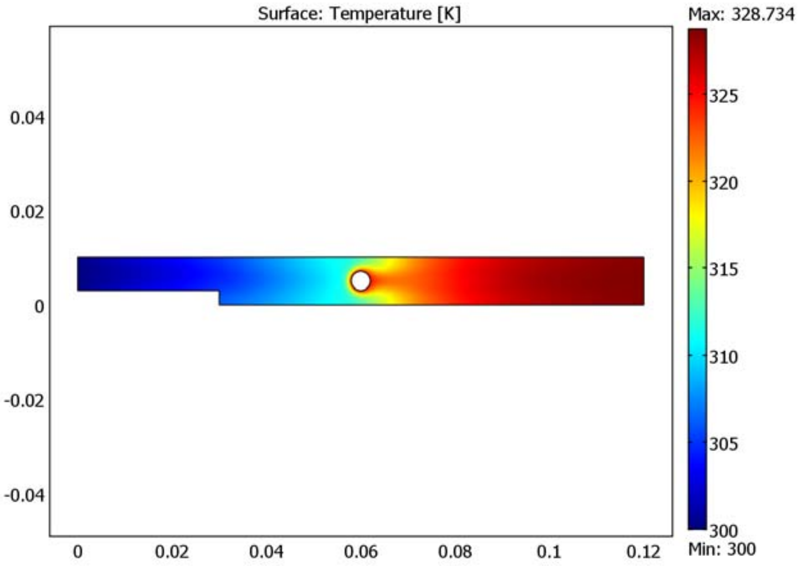


Figure 3-6: Reactor temperature (K) when the heat of reaction is taken into account.

Figure 3-7 plots the rate of reaction as a function of the position in the reactor. Clearly, significant reaction now occurs in the first part of the reactor, before the heating cylinder.

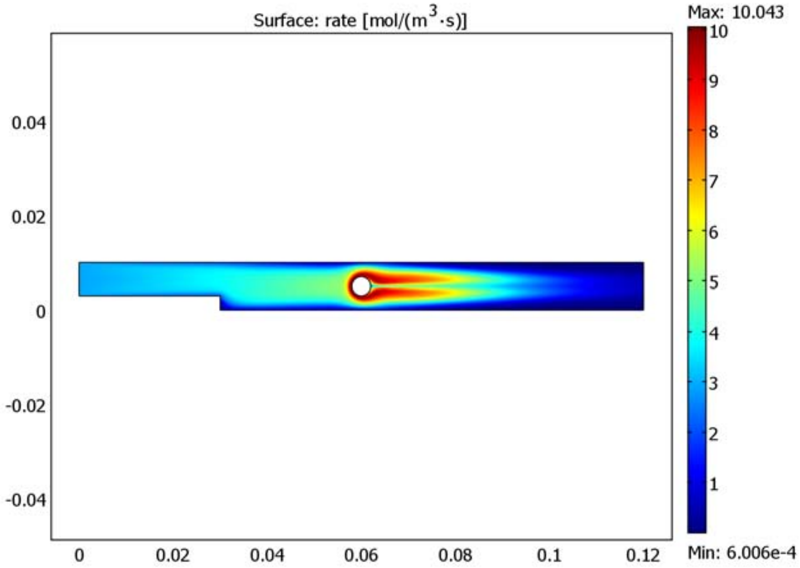


Figure 3-7: Significant decomposition of A occurs in the first half of the reactor.

Reference

1. H. Schlichting, *Boundary Layer Theory, 4th ed.*, McGraw Hill, 1960, page 168.

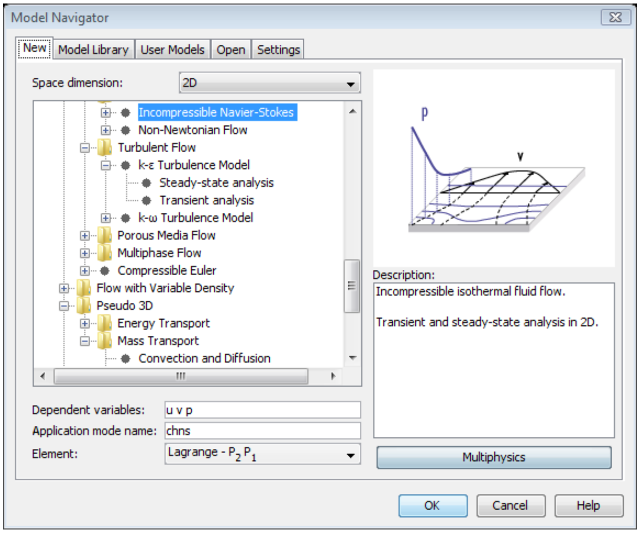
Model Library path: `Chemical_Engineering_Module/Transport_and_Reactions/thermal_decomposition`

To open the model in COMSOL Multiphysics, proceed to the Model Library and following the path given above. Access the **Model Library** page from the **Model Navigator** window, either as you start COMSOL Multiphysics, or by choosing **File>Open Model Library** in the COMSOL Multiphysics main user interface.

MODEL NAVIGATOR

1 Start COMSOL Multiphysics.

2 In the **Application Modes** list, select **Chemical Engineering Module>Momentum Transport>Laminar Flow>Incompressible Navier-Stokes**.



Space dimension: 2D

Description:
Incompressible isothermal fluid flow.

Transient and steady-state analysis in 2D.

Dependent variables: u v p

Application mode name: chns

Element: Lagrange - P_2 P_1

3 Click **OK**.

GEOMETRY MODELING

1 Press Shift and click the **Rectangle/Square** button on the Draw toolbar.

2 Type in the following:

PROPERTY	VALUE
Width	0.12
Height	0.01

3 Click **OK**.

4 Again, press Shift and click the **Rectangle/Square** button.

5 Type in the following:

PROPERTY	VALUE
Width	`0.03`
Height	`0.003`

6 Click **OK**.

7 Select all geometry objects by pressing Ctrl+A.

8 Click the **Difference** button on the Draw toolbar.

9 Press Shift and click the **Ellipse/Circle (Centered)** button.

10 Type in the following:

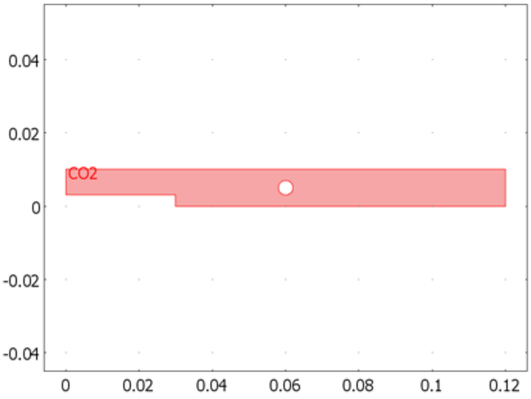
PROPERTY	VALUE
Radius	`0.002`
x	`0.06`
y	`0.005`

11 Click **OK**.

12 Select all geometry objects by pressing Ctrl+A.

13 Click the **Difference** button on the Draw toolbar.

14 Click the **Zoom Extents** button on the Main toolbar.



PHYSICS SETTINGS

Subdomain Settings—Incompressible Navier-Stokes

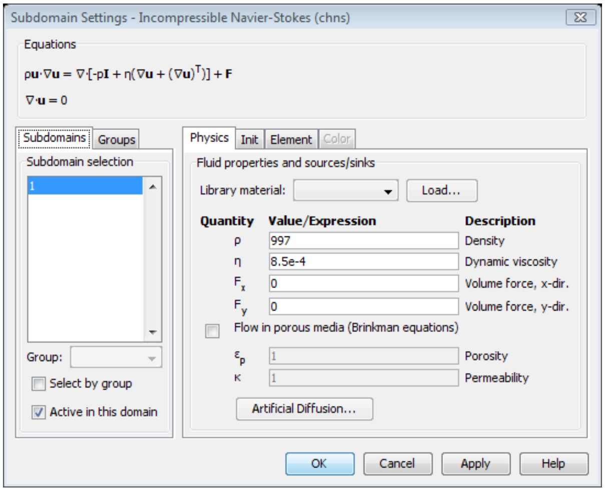
Start by setting up and solving the equations for the fluid flow.

1 Select the menu item **Physics>Subdomain Settings**.

2 Select **1** from the **Subdomain selection** list and type in the following fluid properties:

EDIT FIELD	VALUE
Density	997
Dynamic viscosity	`8.5e-4`

In this model you first make use of constant values for the physical properties of the fluid. Later on, temperature dependent expressions will be loaded from the Materials/ Coefficients Library, replacing the constants. Using these expressions increases the accuracy of the model but also introduces stronger coupling between model equations. This approach, to increase the level of detail and coupling in a stepwise fashion, is often a good one when working with highly coupled Multiphysics models.



Subdomain Settings - Incompressible Navier-Stokes (chns)

Equations

$\rho\mathbf{u}\cdot\nabla\mathbf{u} = \nabla\cdot[-p\mathbf{I} + \eta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)] + \mathbf{F}$

$\nabla\cdot\mathbf{u} = 0$

Subdomain selection: 1

Group:

□ Select by group

☑ Active in this domain

Fluid properties and sources/sinks

Library material:

Quantity	Value/Expression	Description
ρ	997	Density
η	8.5e-4	Dynamic viscosity
F_x	0	Volume force, x-dir.
F_y	0	Volume force, y-dir.

□ Flow in porous media (Brinkman equations)

ε_p	1	Porosity
κ	1	Permeability

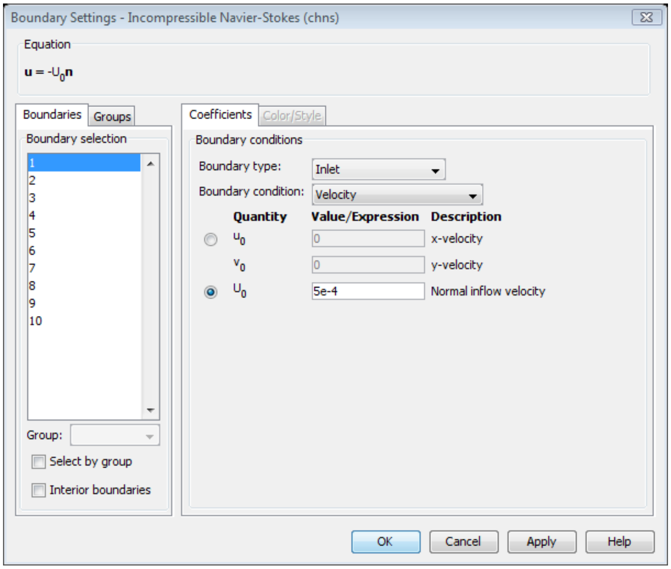
3 Click **OK**.

Boundary Conditions—Incompressible Navier-Stokes

1 Select the menu item **Physics>Boundary Settings**.

2 Set the following boundary conditions:

SETTINGS	BOUNDARY 1	BOUNDARIES 6	ALL OTHERS
Boundary type	Inlet	Outlet	Wall
Boundary condition	Velocity	Pressure	No slip
U_0	5e-4	-	-
P_0	-	0	-



Boundary Settings - Incompressible Navier-Stokes (chns)

Equation

$\mathbf{u} = -U_0\mathbf{n}$

Boundaries | Groups

Boundary selection: 1 (selected), 2, 3, 4, 5, 6, 7, 8, 9, 10

Group:

☐ Select by group

☐ Interior boundaries

Coefficients | Color/Style

Boundary conditions

Boundary type: Inlet

Boundary condition: Velocity

	Quantity	Value/Expression	Description
○	u_0	0	x-velocity
	v_0	0	y-velocity
●	U_0	5e-4	Normal inflow velocity

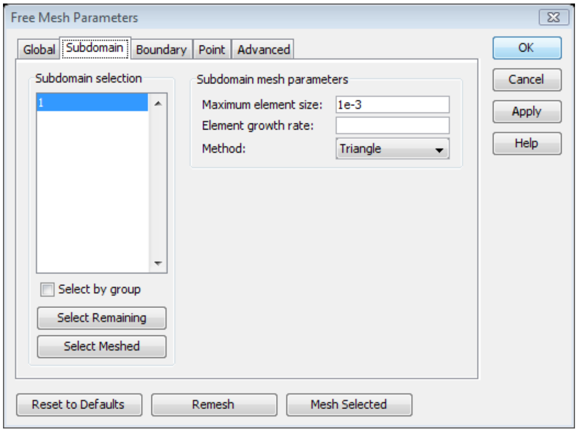
3 Click **OK**.

MESH GENERATION

1 Select the menu item **Mesh>Free Mesh Parameters**.

2 Click the **Subdomain** tab.

3 Type `1e-3` in the **Maximum element size** edit field.



Free Mesh Parameters

Subdomain selection: 1

☐ Select by group

Subdomain mesh parameters

Maximum element size: 1e-3

Element growth rate:

Method: Triangle

4 Click **OK** to close the **Free Mesh Parameters** dialog box.

5 Click the **Initialize Mesh** button on the Main toolbar.

COMPUTING THE SOLUTION

Compute the solution by clicking the **Solve** button on the Main toolbar.

POSTPROCESSING AND VISUALIZATION

1 Click the **Plot Parameters** button on the Main toolbar.

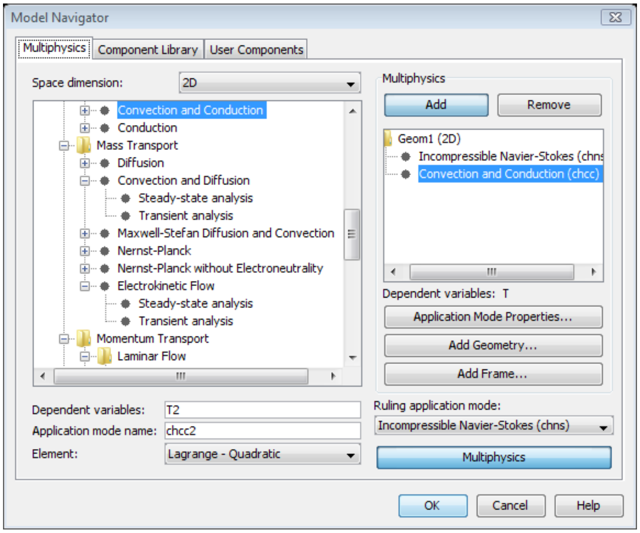
2 In the **Plot type** area on the **General** page, select the **Arrow** check box.

3 Click **OK** to close the **Plot Parameters** dialog box.

MODEL NAVIGATOR

1 Select the menu item **Multiphysics>Model Navigator**.

2 In the **Application Modes** list, select **Chemical Engineering Module>Energy Transport>Convection and Conduction**.



Space dimension: 2D

Multiphysics: Geom1 (2D) — Incompressible Navier-Stokes (chns), Convection and Conduction (chcc)

Dependent variables: T

Dependent variables: T2

Application mode name: chcc2

Element: Lagrange - Quadratic

Ruling application mode: Incompressible Navier-Stokes (chns)

3 Click the **Add** button, then click **OK**.

PHYSICS SETTINGS

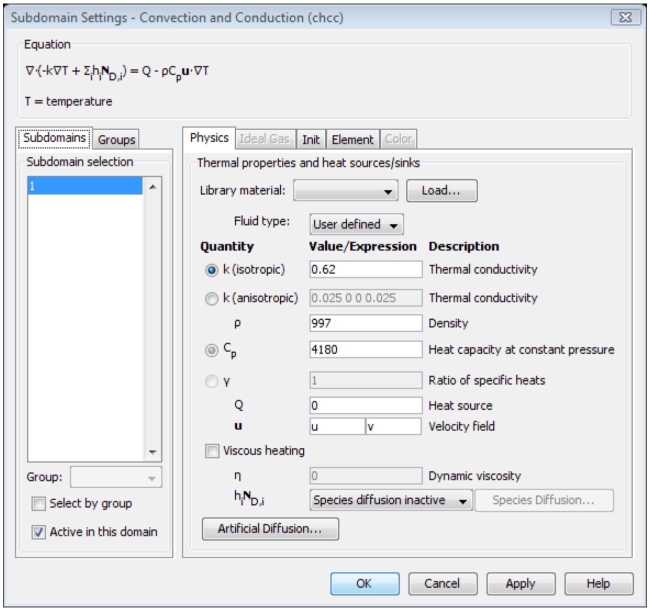
Subdomain Settings—Convection and Conduction

Proceed to set up and couple an energy transport to the flow equations.

1 Select the menu item **Physics>Subdomain Settings**.

2 Select **1** from the **Subdomain selection** list and type in the following entries:

PROPERTY	VALUE
Thermal conductivity	0.62
Density	997
Heat capacity at constant pressure	4180
u	u
v	v



Subdomain Settings - Convection and Conduction (chcc)

Equation

$\nabla\cdot(-k\nabla T + \Sigma_i h_i \mathbf{N}_{D,i}) = Q - \rho C_p \mathbf{u}\cdot\nabla T$

T = temperature

Thermal properties and heat sources/sinks

Library material:

Fluid type: User defined

Quantity	Value/Expression	Description
● k (isotropic)	0.62	Thermal conductivity
○ k (anisotropic)	0.025 0 0 0.025	Thermal conductivity
ρ	997	Density
● C_p	4180	Heat capacity at constant pressure
○ γ	1	Ratio of specific heats
Q	0	Heat source
u	u v	Velocity field

□ Viscous heating

η	0	Dynamic viscosity
$h_i\mathbf{N}_{D,i}$	Species diffusion inactive	

Subdomain selection: 1

□ Select by group

☑ Active in this domain

Entering u and v for the **Velocity field**, tells the software to use dependent variables of the Navier-Stokes application mode, thereby coupling the energy and momentum transport equations.

3 Click the **Init** tab and type 300 in the **T(t_0)** edit field.

4 Click **OK**.

Boundary Conditions—Convection and Conduction

1 Select the menu item **Physics>Boundary Settings**.

2 Set the following boundary conditions:

SETTINGS	BOUNDARY 1	BOUNDARIES 2–5	BOUNDARY 6	BOUNDARIES 7–10
Boundary condition	Temperature	Thermal insulation	Convective flux	Temperature
T_0	300	-	-	325

3 Click **OK**.

COMPUTING THE SOLUTION

Solve the problem by clicking the **Restart** button on the Main toolbar.

POSTPROCESSING AND VISUALIZATION

1 Click the **Plot Parameters** button on the Main toolbar.

2 Go to the **Arrow** page and clear the **Arrow plot** check box.

3 Go to the **Surface** page and select **Convection and Conduction (chcc)>Temperature** from the **Predefined quantities** list.

4 Click **OK**.

MODEL NAVIGATOR

1 Select the menu item **Multiphysics>Model Navigator**.

2 In the **Application Modes** list, select **Chemical Engineering Module>Mass Transport>Convection and Diffusion**.

3 Type c_A in the **Dependent variables** edit field.

4 Click the **Add** button, then click **OK**.

OPTIONS AND SETTINGS

1 Select the menu item **Options>Expressions>Scalar Expressions**.

2 Type in the following entries in the **Scalar Expressions** dialog box (the descriptions are optional):

NAME	EXPRESSION	DESCRIPTION
E	72[kJ/mol]	Activation energy
H	100[kJ/mol]	Heat of reaction
A	1e10[1/s]	Frequency factor
k	A*exp(-E/(8.314[J/(mol*K)]*T))	Rate factor
rate	k*c_A	Reaction rate

Setting up the kinetic parameters and expressions in the **Scalar Expressions** dialog box makes it easy to review and edit these entries.

3 Click **OK**.

PHYSICS SETTINGS

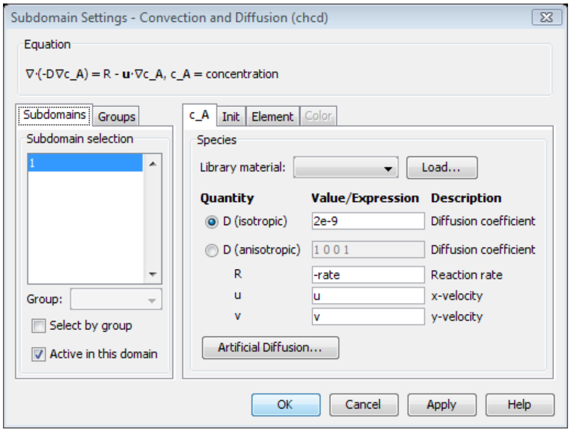
Subdomain Settings—Convection and Diffusion

Now, set up and couple the transport and reaction of mass to the energy and momentum transport equations.

1 Select the menu item **Physics>Subdomain Settings**.

2 Select **1** from the **Subdomain selection** list and type in the following entries:

PROPERTY	VALUE
Diffusion coefficient	`2e-9`
Reaction rate	`-rate`
u	`u`
v	`v`



Subdomain Settings - Convection and Diffusion (chcd)

Equation

∇·(-D∇c_A) = R - **u**·∇c_A, c_A = concentration

Subdomain selection: 1

Group:

☐ Select by group

☑ Active in this domain

Species

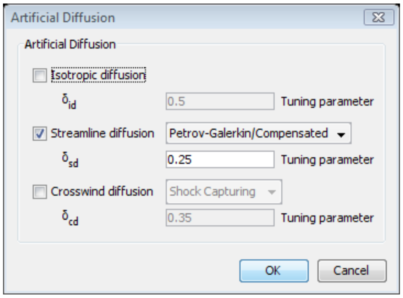
Library material:

Quantity	Value/Expression	Description
● D (isotropic)	2e-9	Diffusion coefficient
○ D (anisotropic)	1 0 0 1	Diffusion coefficient
R	-rate	Reaction rate
u	u	x-velocity
v	v	y-velocity

Because of the low value of the diffusion coefficient in relation to the flow velocity, the problem is strongly convection dominated. Applying artificial diffusion to the mass-transport equation therefore helps to improve convergence when solving the model.

3 Click the **Artificial Diffusion** button.

4 In the **Artificial Diffusion** dialog box, select the **Streamline diffusion** check box and select **Petrov-Galerkin/Compensated** (the default) from the associated list.



Artificial Diffusion

Artificial Diffusion

☐ Isotropic diffusion

δ_{id} 0.5 Tuning parameter

☑ Streamline diffusion Petrov-Galerkin/Compensated

δ_{sd} 0.25 Tuning parameter

☐ Crosswind diffusion Shock Capturing

δ_{cd} 0.35 Tuning parameter

5 Click **OK** twice, first in the **Artificial Diffusion** dialog box and then in the **Subdomain Settings** dialog box.

Boundary Conditions—Convection and Diffusion

1 Select the menu item **Physics>Boundary Settings**.

2 Set the following boundary conditions:

SETTINGS	BOUNDARY 1	BOUNDARY 6	ALL OTHER
Boundary condition	Concentration	Convective flux	Insulation/ Symmetry
c_A_0	1000	-	-

3 Click **OK**.

COMPUTING THE SOLUTION

Solve the problem by clicking the **Restart** button on the Main toolbar.

POSTPROCESSING AND VISUALIZATION

1 Click the **Plot Parameters** button on the Main toolbar.

2 Go to the **Surface** page and select **Convection and Diffusion (chcd)>Concentration, c_A** from the **Predefined quantities** list.

3 Click **OK**.

PHYSICS SETTINGS

Subdomain Settings—Incompressible Navier-Stokes

1 Activate the momentum transport application mode by selecting the menu item **Multiphysics>Incompressible Navier-Stokes (chns)**.

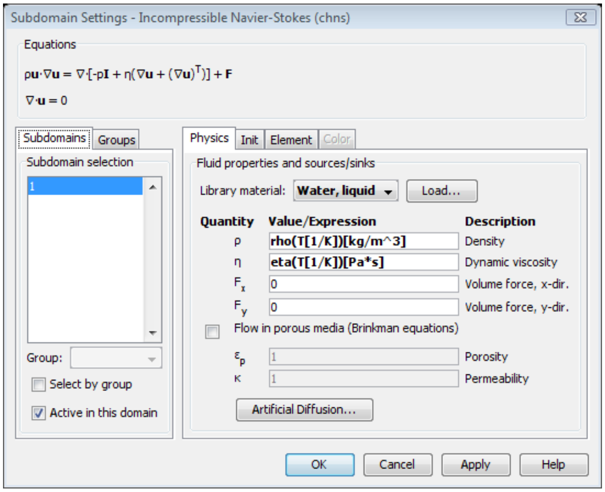
2 Choose **Physics>Subdomain Settings**.

3 Select Subdomain 1 from the **Subdomain selection** list.

4 Click the **Load** button to access the built-in **Materials/Coefficients Library**.

5 Select **Liquids and Gases>Liquids>Water, liquid** from the **Materials** tree.

6 Click **OK** to close the **Materials/Coefficients Library** dialog box.



Subdomain Settings - Incompressible Navier-Stokes (chns)

Equations

$\rho \mathbf{u}\cdot\nabla\mathbf{u} = \nabla\cdot[-p\mathbf{I} + \eta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)] + \mathbf{F}$

$\nabla\cdot\mathbf{u} = 0$

Subdomain selection: 1

Group:

☐ Select by group

☑ Active in this domain

Fluid properties and sources/sinks

Library material: Water, liquid

Quantity	Value/Expression	Description
ρ	rho(T[1/K])[kg/m^3]	Density
η	eta(T[1/K])[Pa*s]	Dynamic viscosity
F_x	0	Volume force, x-dir.
F_y	0	Volume force, y-dir.

☐ Flow in porous media (Brinkman equations)

ε_p	1	Porosity
κ	1	Permeability

Notice that the temperature-dependent functions now appear in the **Density** and **Dynamic viscosity** edit fields.

7 Click **OK**.

Subdomain Settings—Convection and Conduction

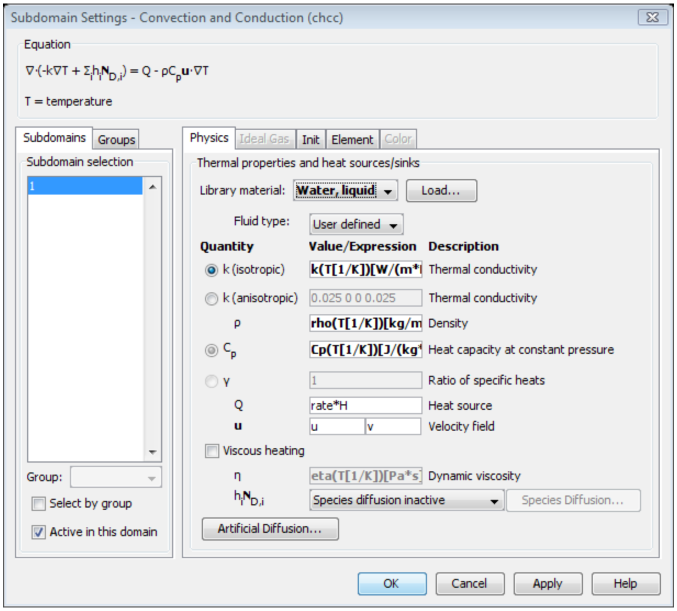
1 Activate the energy transport application mode by choosing **Multiphysics>Convection and Conduction (chcc)**.

2 Choose **Physics>Subdomain Settings**.

3 Select Subdomain 1 from the **Subdomain selection** list.

4 Select **Water, liquid** from the **Library material** list. Notice that the temperature-dependent functions now appear in the **Thermal conductivity**, **Density**, and **Heat capacity at constant pressure** edit fields.

5 Type `rate*H` in the **Q** edit field. This includes the effect of the exothermic reaction in the energy transport equation. The exothermicity of the reaction is 100 kJ/mol.



Subdomain Settings - Convection and Conduction (chcc)

Equation

$\nabla\cdot(-k\nabla T + \Sigma_i h_i \mathbf{N}_{D,i}) = Q - \rho C_p \mathbf{u}\cdot\nabla T$

T = temperature

Subdomain selection: 1

Group:

☐ Select by group

☑ Active in this domain

Thermal properties and heat sources/sinks

Library material: Water, liquid

Fluid type: User defined

Quantity	Value/Expression	Description
● k (isotropic)	k(T[1/K])[W/(m*I	Thermal conductivity
○ k (anisotropic)	0.025 0 0 0.025	Thermal conductivity
ρ	rho(T[1/K])[kg/m	Density
○ C_p	Cp(T[1/K])[J/(kg'	Heat capacity at constant pressure
○ γ	1	Ratio of specific heats
Q	rate*H	Heat source
u	u | v	Velocity field

☐ Viscous heating

Quantity	Value/Expression	Description
η	eta(T[1/K])[Pa*s]	Dynamic viscosity
$h_i \mathbf{N}_{D,i}$	Species diffusion inactive	Species Diffusion...

6 Click **OK**.

COMPUTING THE SOLUTION

Click the **Restart** button on the Main toolbar to compute the solution.

POSTPROCESSING AND VISUALIZATION

The concentration of A is plotted as a result of the current settings.

1 Click the **Plot Parameters** button on the Main toolbar.

2 Click the **Surface** tab. On the **Surface Data** page, select **Convection and Conduction (chcc)>Temperature** from the **Predefined quantities** list.

3 Click **OK**.

With these final steps you have completed the detailed modeling of a reacting system, involving fully coupled equations for mass, energy, and momentum transport. The application modes of the Chemical Engineering Module make it easy to add and couple transport equations, allowing you to set up and solve complex models in a controlled, stepwise fashion.

4

Modeling Transport and Reactions

This chapter presents the most important features available in the application modes in the Chemical Engineering Module. Features for laminar and turbulent flow, non-isothermal flow, two-phase flow, multicomponent transport, and reaction kinetics in non-isothermal systems are discussed based on their implementation in the graphical user interface.

Laminar and Turbulent Flow

Introduction

The underlying structure of the application modes for fluid flow in the Chemical Engineering Module are momentum balances coupled to a mass balance. In laminar flow, this consists of one momentum balance for each spatial coordinate direction and a mass balance ensuring the preservation of fluid mass. This is expressed by the Navier-Stokes equations.

In theory, the Navier-Stokes equations can also describe turbulent flow. However, this would require unreasonably large mesh resolutions, which leads to large memory requirements and long simulation times. For this reason, *turbulence models* are used in combination with the Navier-Stokes equations, which represent the turbulence eddies in the flow and replace their convective contribution in the momentum balance with viscous momentum transport. This viscosity, which is an additional entity to the molecular viscosity, is referred to as the *eddy viscosity*.

The turbulence models included as application modes in the Chemical Engineering Module add two equations to the Navier-Stokes equations. The k-ε and k-ω models include balances of the turbulence kinetic energy, k (m^2/s^2), and of the dissipation, ε (m^2/s^3), or specific dissipation, ω (1/s), of this energy.

This section discusses the features available for modeling fluid flow in the Chemical Engineering Module. It covers activating and switching turbulence models and selecting boundary conditions.

Activating and Switching Turbulence Models

There are three ways of setting up turbulent flow simulations in the Chemical Engineering Module using the predefined turbulence models:

- You can select one of the k-ε or k-ω turbulence model application modes in the Model Navigator.
- You can also start with the Incompressible Navier-Stokes application mode and then activate a turbulence model if the Navier-Stokes equations cannot resolve the flow. The software automatically reformulates the equations with the exception that you

need to specify additional boundary conditions for the turbulence model. For example, the eddy viscosity automatically replaces the molecular viscosity.

- You can also select the k-ε or k-ω turbulence model application modes and switch between the two within the same model. The equations are automatically reformulated.

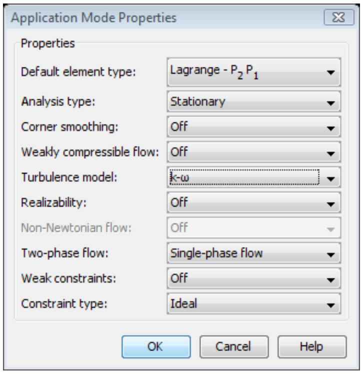
You can often solve a model using the k-ε turbulence model to get a first estimate of the solution and then change to the k-ω turbulence model and re-solve the problem. This procedure is highly recommended if convergence problems arise when using the k-ω turbulence model from the beginning.

The following example shows how to change turbulence model in an already solved model.

1 Start COMSOL Multiphysics by double-clicking on its desktop icon.
2 Click the **Model Library** tab in the **Model Navigator**.
3 Select **Chemical Engineering Module>Fluid Flow>turbulent_backstep**.

   This opens the **turbulent_backstep** model in the Postprocessing mode, showing the resulting flow field using the k-ε Turbulence Model application mode. You can now switch to the k-ω turbulence model.
4 Select **Properties** from the **Physics** menu.

   This opens the **Application Mode Properties** dialog box. The dialog box allows you to select turbulence model or remove the turbulence equations, add two-phase flow, and include effects of weak compressibility.



Application Mode Properties

Properties

Default element type: Lagrange - $P_2 P_1$

Analysis type: Stationary

Corner smoothing: Off

Weakly compressible flow: Off

Turbulence model: k-ω

Realizability: Off

Non-Newtonian flow: Off

Two-phase flow: Single-phase flow

Weak constraints: Off

Constraint type: Ideal

5 Select **k-ω** from the **Turbulence model** list.

6 Click **OK**.

You can now solve the model using the k-ω turbulence model with the solution from k-ε as a starting point. This particular model uses adaptive meshing. However, because the mesh has already been generated using the k-ε turbulence model, you can deactivate adaptive meshing when re-solving the model.

7 Click the **Solver Parameters** button on the Main toolbar.

8 Clear the **Adaptive mesh** check box.

9 Click **OK**.

10 Click the **Restart** button on the Main toolbar to resolve the problem using the k-ω turbulence model.

Selecting Boundary Conditions

There are six types of predefined boundary settings in the fluid-flow application modes in the Chemical Engineering Module: Wall, Inlet, Outlet, Symmetry boundary, Open boundary, and the Stress type. Each of these types is briefly explained below.

THE WALL CONDITIONS

The Wall boundary condition type allows you to select **No slip**, **Logarithmic wall function**, **Slip**, **Sliding wall**, and **Moving wall** boundary conditions.

- The physically correct boundary condition for a solid wall is the **No slip** condition. This condition sets the velocity to zero in all directions. For turbulent flow, use the **Logarithmic wall function** instead because you cannot afford to resolve the boundary layer.
- In the case that you do not want to resolve the boundary layer, use the **Slip** condition. This is a reasonable approximation if the flow at this specific wall does not have a great impact on the solution in the domain. The condition implies that there are no tangential forces on the boundary and the velocity component perpendicular to the boundary is zero.
- When a wall moves with a prescribed tangential velocity, use the **Sliding wall** condition.
- For moving frames, which are used in, for example, the Rotating Machinery application mode, you need to be able to set wall movement in any direction. You can then use the **Moving/leaking wall** condition, which assumes that the wall is moving with a prescribed velocity. The same condition can be used if you want to model fluid leaking through a stationary wall. For laminar flow, this is identical to

setting the velocity vector. For turbulent flows, a logarithmic wall function formulation is used.

THE INLET CONDITIONS

When you need to specify an Inlet boundary condition, you have several different options depending on what you know about your system. These options are the following:

- When you know the pressure at the inlet, use the **Pressure, no viscous stress** condition. For laminar flows, this condition is the same as the **Pressure, no viscous stress** condition used for outflows. This means that depending on the pressure in the rest of the domain, this condition can result in an outflow of fluid. For turbulent flows it is important that a boundary with this condition does become an inflow, because the turbulence variables require a net inflow to be correct.
- If you want to specify the velocity vector at an inlet or the velocity in the direction perpendicular to a boundary, use the **Velocity** boundary condition. This is the default boundary condition for inlets. For laminar flows, you can also select the **Laminar inflow** condition, which automatically defines a fully developed laminar flow profile at the inlet.

For turbulence models, you need estimates of the turbulent length and turbulent intensity for both the **Velocity** and **Pressure, no viscous stress** boundary conditions. Alternatively, you can specify values for the turbulent kinetic energy and the dissipation rate (when using the k-ε model) or the turbulent kinetic energy and the specific dissipation rate (when using the k-ω model).

THE OUTLET CONDITIONS

The options for the Outlet boundary condition types are similar to the Inlet types; the difference is found in the turbulence models.

- When you know the pressure at an outlet, you can use the **Pressure, no viscous stress** condition or the **Pressure** condition. The **Pressure, no viscous stress** condition is the default boundary condition for outlets. As in the **Inlet** case, the pressure field in the rest of the domain determines if this actually becomes an outlet. For turbulence models it is analogous to the inlet case, and it is important that these conditions actually lead to outflow of fluid.
- You can specify the velocity vector and a velocity component perpendicular to the boundary using the **Velocity** boundary condition. In the case you want to specify a fully developed laminar flow profile, select the **Laminar outflow** condition, which

automatically calculates the velocity profile for a boundary with arbitrary cross section.

- The **Normal stress** condition is an alternative way to specify a pressure level. Unlike the pressure conditions discussed above, the **Normal stress** condition does not lock the pressure pointwise, but allows the pressure to deviate slightly from the specified value.
- When you want to avoid setting any constraints on the pressure at a boundary, use the **No viscous stress** condition. Because this boundary condition leaves the pressure field completely unconstrained, it can lead to problems in convergence if the boundary surface is relatively large. Therefore, use this condition with care.

THE SYMMETRY BOUNDARY CONDITION

As the name reveals, you can use this condition to define symmetry at a boundary. This condition assures that there are no net velocity components or momentum transport perpendicular to a boundary. Note that there is also a specific **Axial symmetry** condition for the corresponding application modes modeled in this domain.

THE OPEN BOUNDARY CONDITIONS

The Open boundary type contains two different conditions. You can use them in the following cases:

- If you want to specify that a boundary is adjacent to the surrounding with a known pressure level, use the **Normal stress** condition. The pressure is not locked pointwise but can adjust slightly to the flow field.
- If you do not want to constrain the pressure to a given value, you can select the **No viscous** stress condition. Since this leaves the pressure field completely unconstrained, this condition can lead to problems in convergence if the boundary surface is relatively large. Therefore, use this condition with care.

Use both conditions with care when combined with turbulence models. If there are large inflows or outflows of mass at these boundaries, these conditions can result in convergence problems. You can then switch to the Stress conditions (see below) or to the Outlet or Inlet conditions.

THE STRESS CONDITIONS

You can use the Stress conditions to set the direction and magnitude of the stress at a boundary.

- If you want to specify the components of the total stress vector, you can select the **General stress** condition.

- If you want to set the magnitude of the total stress perpendicular to the boundary, you can select the **Normal stress**.
- If you want to set the magnitude of the total stress perpendicular to a boundary, and at the same time eliminate the velocity components tangential to the boundary, you can select the **Normal stress, normal flow** condition.

Note that you can combine these conditions with Inlet, Outlet, and Open boundary conditions for the turbulent kinetic energy and the dissipation or specific dissipation for turbulence models.

Non-Isothermal Flow and Flow with Variable Density

The Chemical Engineering Module contains a suite of application modes for flow with variable density. The application modes found in this folder in the Model Navigator differ in the extent of predefinition of the multiphysics couplings and fluid properties.

The Non-Isothermal Flow application mode automatically couples the Weakly Compressible Navier-Stoke application mode with the Convection and Conduction application mode from the Energy Transport folder. In addition, for ideal gases it predefines the density expression.

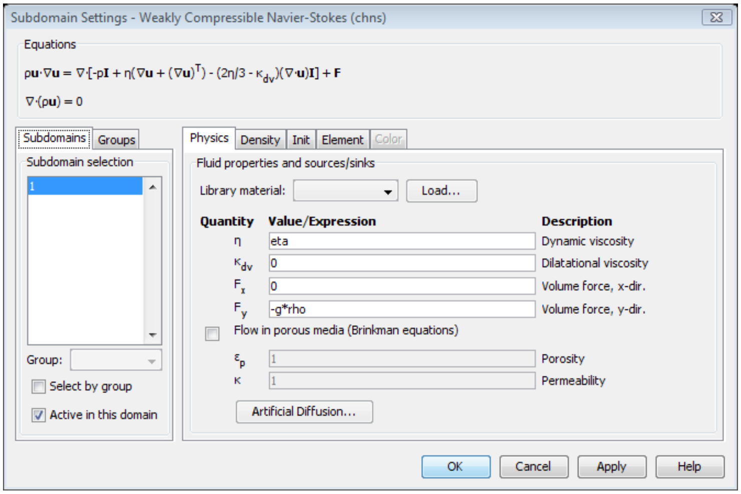
If you have a relation for the fluid's density dependence on pressure or temperature, or both pressure and temperature, the application mode automatically adds the proper stabilization to the equations.

If the fluid density varies but you still want to assume a constant temperature, then use the Weakly Compressible Navier-Stokes application mode. This application mode can add a stabilization term when density depends on pressure, for example in the case of density variations due to composition for ideal gases.

The setup of problems involving gravity using the Non-Isothermal Flow application mode requires some careful manipulation of the boundary conditions. The following section contains a discussion about this specific type of problem, which is relevant for, among other things, the modeling of free convection.

Gravity and Boundary Conditions

The influence of gravity on the flow pattern is often an important issue when modeling flow in fluids with variable density. You can account for this influence in the model equations by adding to the momentum balances the volume force $-\rho\mathbf{g}$, where ρ denotes density (kg/m^3) and $\mathbf{g}$ the gravity vector (m /s^2). To do this in any of the Chemical Engineering Module's application modes for free fluid flow, enter the components of this vector in the **Volume force** edit fields in the **Subdomain Settings** dialog box (see Figure 4-1).



Equations

$$\rho\mathbf{u}\cdot\nabla\mathbf{u} = \nabla\cdot[-p\mathbf{I} + \eta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - (2\eta/3 - \kappa_{dv})(\nabla\cdot\mathbf{u})\mathbf{I}] + \mathbf{F}$$

$$\nabla\cdot(\rho\mathbf{u}) = 0$$

Subdomain selection: 1

Group:

□ Select by group

☑ Active in this domain

Fluid properties and sources/sinks

Library material:

Quantity	Value/Expression	Description
η	eta	Dynamic viscosity
κ_{dv}	0	Dilatational viscosity
F_x	0	Volume force, x-dir.
F_y	-g*rho	Volume force, y-dir.

□ Flow in porous media (Brinkman equations)

ε_p	1	Porosity
κ	1	Permeability

*Figure 4-1: The term -g*rho adds the influence of gravity in the momentum balance in the vertical (y) direction.*

When you add this term, you have to be careful when setting up boundary conditions at the outlets. The following simple example, which is included in the Model Library, demonstrates some possible boundary settings for this type of problems.

Model Library path:

`Chemical_Engineering_Module/Fluid_Flow/gravity_tutorial`

Consider a horizontal channel of width 0.5 m and length 2 m (see Figure 4-2).The gravity vector **g** is aligned in the negative y direction, that is, $F_y = -\rho g$. In this case, the fluid has a dynamic viscosity, η, of 1.0 Pa·s and a density, ρ, of 1000 kg/m^3.

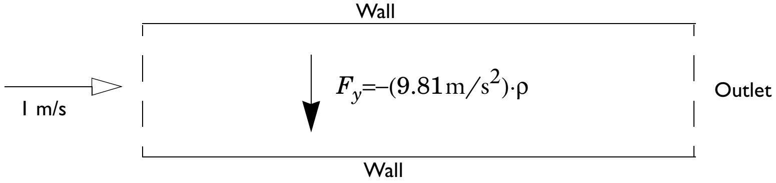


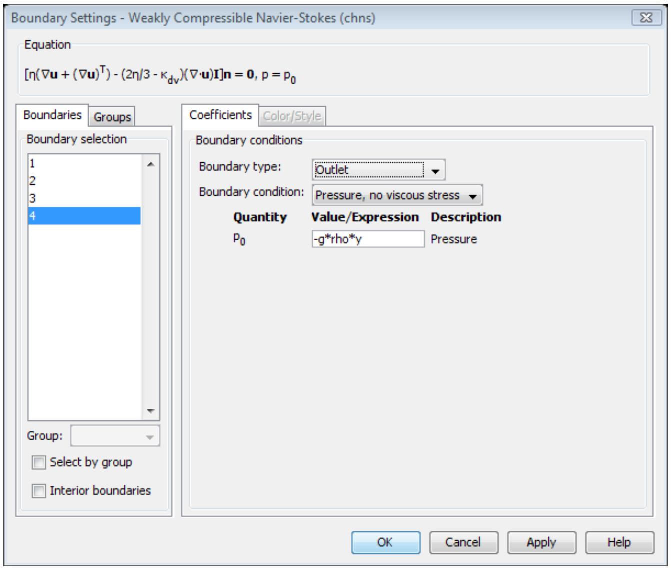
Figure 4-2: A simple example of flow with gravity volume force.

Assume that the channel continues on the outlet side, which means that the fluid leaves the domain following horizontal streamlines. There are three alternatives to simulate this behavior: a pressure profile, a point setting, or a pressure shift.

PRESSURE PROFILE

In a column of fluid, there is a hydrostatic pressure equal to $-\rho g(y - y_0)$ where y_0 is a horizontal reference location (m). In this case $y_0 = 0$. To specify this pressure profile at the outlet, select the **Pressure, no viscous stress** or **Pressure** boundary condition for the

Outlet boundary condition type and use the pressure setting $p_0 = -g\rho y$ (Pa) (see Figure 4-3).



Boundary Settings - Weakly Compressible Navier-Stokes (chns)

Equation

$[\eta(\nabla \mathbf{u} + (\nabla \mathbf{u})^T) - (2\eta/3 - \kappa_{dv})(\nabla \cdot \mathbf{u})\mathbf{I}]\mathbf{n} = \mathbf{0},\ p = p_0$

Boundary type: Outlet

Boundary condition: Pressure, no viscous stress

Quantity	Value/Expression	Description
p_0	-g*rho*y	Pressure

*Figure 4-3: Compensation of the pressure level for the hydrostatic pressure is here obtained through the expression, -g*rho*y.*

Figure 4-4 shows the simulated result, where $p = 0$ at $y = 0$ and where flow leaves straight out of the domain as expected.

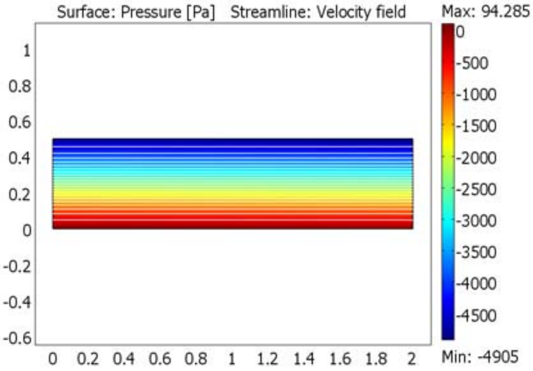


Figure 4-4: The pressure and velocity field with a prescribed pressure profile at the outlet.

POINT SETTING

If you do not want to prescribe a pressure profile at the outlet, you can alternatively prescribe vanishing viscous stresses at the outlet and combine them with a point setting at the outlet. To do so, select **No viscous stress** as the boundary condition at the outlet and a point setting of $p = 0$ in the lower-right corner of the domain. The result is almost identical to the result in Figure 4-4.

PRESSURE SHIFT

Much of the pressure gradient is present only to balance the hydrostatic pressure. You can use this knowledge to introduce a shift in the pressure variable. The theory, described in the section "The Boussinesq Approximation" on page 126, implies that the pressure, p, is replaced by a new pressure variable, P (Pa)

$$\nabla p = \nabla(P + \rho_0 \Phi) \tag{4-1}$$

Here, $\nabla(\rho_0\Phi)$ is the pressure gradient that balances the hydrostatic volume force if the fluid has a constant density ρ_0. The term $\nabla(\rho_0\Phi)$ can be canceled for the term $-\rho_0 g$ in the volume force, leaving $F_y = g(\rho_0 - \rho)$.

Applied to this case, F_y equals zero because the density is constant. In general, F_y can be nonzero, but it is in almost all cases small compared to the hydrostatic pressure. It is therefore possible to use the **Normal stress** boundary condition with zeros stress at the outlet. The result appears in Figure 4-5, which shows that the hydrostatic pressure gradient has been removed and only the pressure gradient necessary to drive the flow remains.

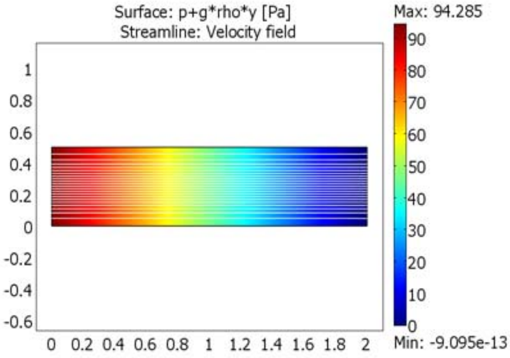


Figure 4-5: Solution to the problem outlined in Figure 4-2 with a shift in the pressure variable and zeros normal stress prescribed on the outlet.

If you select the **Pressure** or **Normal stress** boundary condition for the outlet without compensating for the hydrostatic pressure, you allow for the liquid to “fall” out of the domain, which in the case of a long channel is not a realistic description.

Free and Porous Media Flow

Fluid flows in porous media can be modeled by either using Darcy's law or the Brinkman equations. The Darcy's Law application mode is preferable when the major driving force for the flow is the pressure gradient, and the momentum transfer by shear stresses within the fluid is negligible. This is usually true in the case of low hydraulic permeability of a porous medium.

The Brinkman equations present an extension of Darcy's law. They include terms that account for the viscous transport in the momentum balances, and they treat both the pressure and the flow velocity vector as dependent variables.

An important advantage of the Brinkman Equations application mode is the ability to model a combination of subdomains of free flow with subdomains of porous media flow.

In porous subdomains, the flow variables and fluid properties are averaged over a control volume surrounding a point. This control volume must be small compared to the typical macroscopic dimensions of the problem, but large enough to be representative of the averaged porous structure.

Porosity is defined as the fraction of the control volume occupied by pores. Thus, porosity can vary from zero, for pure solid regions, to one, for subdomains of free flow.

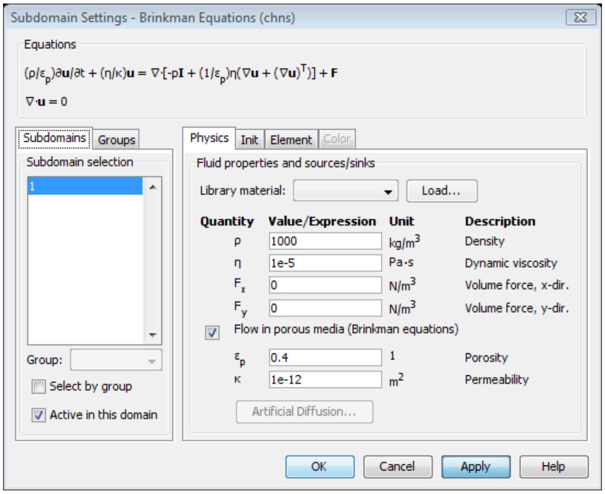
Momentum Transport

The physical properties of the fluid, such as density and viscosity, as well as pressure are defined as so-called intrinsic volume averages, which correspond to a unit volume of pores. Through this definition, they present the relevant physical parameters that can be measured experimentally, and are assumed to be continuous with the corresponding parameters in free flow.

The flow velocities are defined as so-called superficial volume averages, which are averaged over unit volumes of the medium including both pores and matrix. They are sometimes called Darcy velocities, defined as volume flow rates per unit cross section of the medium. Such a definition makes the velocity field continuous across the boundaries between porous subdomains and free flow subdomains.

Through these definitions, the flow can be modeled by using the same set of dependent variables for all subdomains; of both porous and free flow. The distinction between porous and free flow subdomains are made by switching on and off certain terms in the governing equations. The boundaries between adjacent subdomains are treated as interior boundaries.

The Brinkman Equations application mode is fully integrated with the Incompressible Navier-Stokes application mode and other application modes for fluid flow. You can switch between free flow and porous media flow in a given subdomain by selecting the **Flow in porous media (Brinkman equations)** check box in the **Subdomain Settings** dialog box; see Figure 4-6.



Subdomain Settings - Brinkman Equations (chns)

Equations

$(\rho/\varepsilon_p)\partial\mathbf{u}/\partial t + (\eta/\kappa)\mathbf{u} = \nabla\cdot[-p\mathbf{I} + (1/\varepsilon_p)\eta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)] + \mathbf{F}$

$\nabla\cdot\mathbf{u} = 0$

Subdomain selection: 1

Group:

□ Select by group

☑ Active in this domain

Fluid properties and sources/sinks

Library material:

Quantity	Value/Expression	Unit	Description
ρ	1000	kg/m^3	Density
η	1e-5	Pa·s	Dynamic viscosity
F_x	0	N/m^3	Volume force, x-dir.
F_y	0	N/m^3	Volume force, y-dir.

☑ Flow in porous media (Brinkman equations)

Quantity	Value/Expression	Unit	Description
ε_p	0.4	1	Porosity
κ	1e-12	m^2	Permeability

Figure 4-6: Physics page when Flow in porous media is switched on.

Mass Transport

For the free-flow domains, the total molar flux, $\mathbf{N}$ $(\mathrm{mol/(m^2 \cdot s^1)})$, of a species:

$$\mathbf{N} = -D\nabla c + c\mathbf{u} \tag{4-2}$$

where c is the molar concentration of the species $(\mathrm{mol/m^3})$, $\mathbf{u}$ is the velocity vector (m/s), and D is the diffusion coefficient $(\mathrm{m^2/s})$. The mass balance gives the following convection and diffusion transport equation for the concentration:

$$\frac{\partial c}{\partial t} + \nabla \cdot \mathbf{N} = 0 \tag{4-3}$$

In porous subdomains, the molar flux is the superficial volume-averaged molar flux (number of moles transferred per unit area and time):

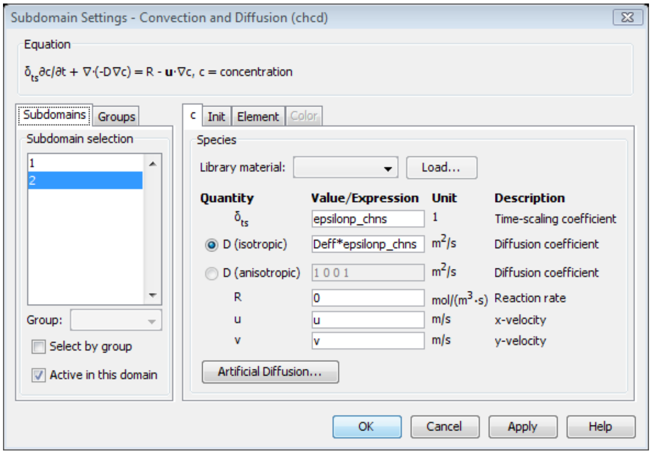
$$\mathbf{N} = -\varepsilon_p D_{\mathrm{eff}} \nabla c + c\mathbf{u} \tag{4-4}$$

where $\mathbf{u}$ is the superficial volume-averaged velocity vector (m/s) of the mixture, ε_p is the porosity, and D_{eff} is the effective diffusion coefficient. Note that you need to provide an expression for D_{eff}, which depends on the type of the porous medium.

The concentration, c, in Equation 4-4 represents the intrinsic volume-averaged molar concentration of the species, which is the number of moles per unit volume of the medium. The resulting transport equation for the porous subdomains is:

$$\frac{\partial \varepsilon_p c}{\partial t} + \nabla \cdot \mathbf{N} = 0 \tag{4-5}$$

Both the concentration and flux remain continuous across the boundaries of adjacent porous subdomains and free-flow subdomains. This makes it possible to use a single Convection and Diffusion application mode for modeling the species transport. Figure 4-7 demonstrates how you to use the Convection and Diffusion application mode for the porous subdomains.



Subdomain Settings - Convection and Diffusion (chcd)

Equation

$\delta_{ts}\partial c/\partial t + \nabla\cdot(-D\nabla c) = R - \mathbf{u}\cdot\nabla c$, c = concentration

Subdomain selection: 1, 2 (2 selected)

Group:

□ Select by group

☑ Active in this domain

Species

Library material:

Quantity	Value/Expression	Unit	Description
δ_{ts}	epsilonp_chns	1	Time-scaling coefficient
● D (isotropic)	Deff*epsilonp_chns	m^2/s	Diffusion coefficient
○ D (anisotropic)	1 0 0 1	m^2/s	Diffusion coefficient
R	0	$mol/(m^3 \cdot s)$	Reaction rate
u	u	m/s	x-velocity
v	v	m/s	y-velocity

Figure 4-7: Settings for the species transport in a porous subdomain. The corresponding application mode for the flow field is shown in the Multiphysics menu and the predefined variable epsilonp_chns *corresponds to the porosity* ε_p *in this application mode.*

For a multicomponent mixture, a transport equation similar to Equation 4-5 can be used for each species in the mixture. A sum of all these equations gives the continuity equation for the mixture:

$$\frac{\partial \varepsilon_p \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \qquad (4\text{-}6)$$

where ρ presents the mixture density. Thus, for a mixture of n components, you need to use the continuity equations and $n - 1$ transport equations for the concentrations.

Energy Transport

The temperature is defined per unit volume of the medium as a thermal equilibrium between the pore fluid and solid matrix is assumed. The heat flux is again a superficial volume-averaged quantity.

The heat balance equation is:

$$\rho c_{p,\,\mathrm{eff}} \frac{\partial T}{\partial t} + \rho c_{p,\,\mathrm{eff}}\ \mathbf{u} \cdot \nabla T = \nabla \cdot (k_{\mathrm{eff}} \nabla T) + Q \qquad (4\text{-}7)$$

where k_{eff} is the effective thermal conductivity of the medium (W/(m·K)), and $c_{p,\mathrm{eff}}$ is the effective heat capacity (J/(kg·K)). You need to provide expressions for these effective thermal properties, which can depend on the type of the porous medium used.

Transport in Multicomponent Solutions

There are two different ways of dealing with multicomponent transport in the Chemical Engineering Module. You can use the Convection and Diffusion application mode or the Maxwell-Stefan Diffusion and Convection application mode.

The Nernst-Planck and Nernst-Planck without Electroneutrality application modes are special versions of the Convection and Diffusion application modes for electrolytes. The difference arises in the fact that they also include the contribution of migration due to an electric field in the flux vector for charged species.

The Convection and Diffusion application mode has the property that it only accounts for solute-solvent interactions; that is, it does not account for interactions between solute species. In addition, the application mode does not add any constraints on the sum of the species diffusive mass flux. For "infinitely" dilute solutions, you can use this application mode without any further manipulations.

In a "real" solution, the sum of the diffusive mass fluxes for all species in a solution is zero, because diffusion gives the deviation of the mass flux of a species from the average mass flow with respect to the species mass fraction. This is automatically accounted for by the Maxwell-Stefan Diffusion and Convection application mode. In addition, this application modes also accounts for solute-solute interactions in a solution. If the mass flux of one solute has a significant influence on the total mass flux of the solution, then select the Maxwell-Stefan Diffusion and Convection application mode.

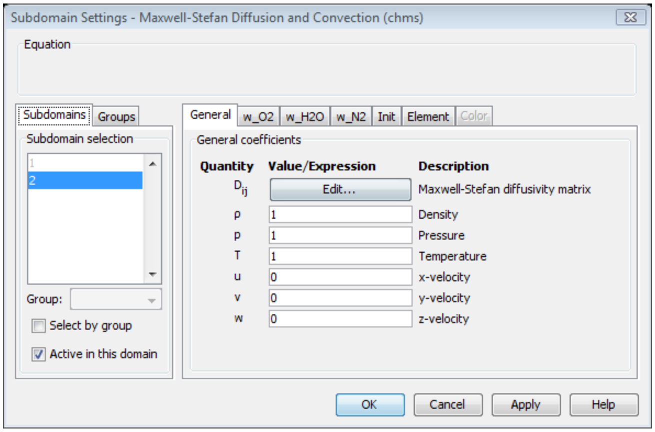
For turbulent flow, the Maxwell-Stefan Diffusion and Convection application mode might not be necessary because the molecular diffusivity is negligible compared to the eddy diffusivity. In such cases, select the Convection and Diffusion application mode instead.

Using the Maxwell-Stefan Diffusion and Convection Application Mode

When using the Maxwell-Stefan Diffusion and Convection application mode, bear in mind that it sets up $n-1$ material balances for a system with n species. The application mode defines the last equation automatically through setting the sum of all mass fluxes to be equal to the total mass flux of the solution, which is obtained from the continuity equation for the solution:

$$\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\mathbf{u}) = 0 \qquad (4\text{-}8)$$

The Maxwell-Stefan Diffusion and Convection application mode has an edit field for coupling the material balances to the Navier-Stokes equations; see the figure below. In the presence of chemical reactions, pressure variations can arise due to changes in composition, which in turn induce a velocity field.



☐ Select by group

☑ Active in this domain

Quantity	Value/Expression	Description
D_{ij}	Edit...	Maxwell-Stefan diffusivity matrix
ρ	1	Density
p	1	Pressure
T	1	Temperature
u	0	x-velocity
v	0	y-velocity
w	0	z-velocity

Figure 4-8: The Maxwell-Stefan Diffusion and Convection application mode has edit fields for the velocity components for the net flow of the solution, u, v, and w for the three space directions.

Take the simple example of forming an oxide or water by a gaseous oxygen-consumption reaction on the surface of a solid. If you set up a material balance in the gas phase, the oxygen reaction at the reacting surface yields a boundary condition where there is a net flux of oxygen out of the gas phase. When oxygen is incorporated as an oxide in the solid phase, or as water, the density increase is large enough so that you can neglect the change in volume of the gas phase domain. Assuming that you have not applied external pressure differences to the system, it is easy to conclude that the velocity component can be set to zero, implying that transport only takes place by diffusion. This is the most common mistake when using this application mode.

If you set the velocity to be equal to zero in the Maxwell-Stefan Diffusion and Convection application mode, the net flux of oxygen out of the system force sa net influx of the n:th species into the system, because the total mass flux has to be zero if the velocity field is set to zero. In the case of air, you could, for example, set up oxygen and water species in the gas phase, along with nitrogen; with nitrogen being the n:th species. In such a case, the model results in a net production of nitrogen at the boundary where oxygen reacts, which is of course incorrect.

What happens in reality is that the consumption of oxygen causes a net flux of air toward the surface where oxygen reacts. The velocity field in the Maxwell-Stefan Diffusion and Convection application mode can therefore not be zero. The net convective term in the transport of nitrogen and water in the gas phase is counteracted by an equally large mass flux through diffusion from the solid wall.

To compute the velocity field, you can couple the Maxwell-Stefan Diffusion and Convection application mode to an equation or a set of equations for the total flow of the mixture, for example the Navier-Stokes equations or Darcy's law in porous media. This is exemplified in the model Fuel Cell Cathode on page 541 in the *Chemical Engineering Module Model Library.*

Reaction Kinetics in Isothermal and Nonisothermal Systems

Optimizing the conditions for chemical reaction is a central driving force in chemical engineering. Chemical reactions that occur in a specific part of a reactor are influenced by the local concentration and temperature. Furthermore, the reactions themselves might produce or consume heat as well as dramatically change the composition of the reacting mixture. Such changes can affect flow properties of the fluid in the reactor. As a consequence, models of chemical engineering applications often need to account for coupled equations describing mass, energy, and momentum transport in order to produce accurate results.

The following sections provide guidelines for how to set up and couple such systems of equations by using application modes in the Chemical Engineering Module.

Isothermal Systems

Consider a general reaction belonging to a set of *j* reactions and involving *i* species:

$$\mathrm{aA} + \mathrm{bB} + \cdots \underset{k_j^r}{\overset{k_j^f}{\rightleftharpoons}} \mathrm{xX} + \mathrm{yY} + \cdots$$

For such a set, the reaction rates r_j (mol/(m^3·s)), can be described by the mass action law:

$$r_j = k_j^f \prod_{i \in \mathrm{react}} c_i^{-\nu_{ij}} - k_j^r \prod_{i \in \mathrm{prod}} c_i^{\nu_{ij}} \qquad (4\text{-}9)$$

Here, k_j^f and k_j^r denote the forward and reverse rate constants, respectively. The concentration of species *i* is denoted as c_i. The stoichiometric coefficients are denoted ν_{ij} and are defined as being negative for reactants and positive for products. For isothermal systems the rate constants are typically given as numerical values.

The rate expression for a given species contains a contribution from each reaction in which the species participates, so that

$$R_i = \sum_j \nu_{ij} r_j$$

First and foremost, chemical reactions relate to the mass balance equations. For instance, the Convection and Diffusion application mode sets up and solves the following equation:

$$\frac{\partial c_i}{\partial t} + \nabla \cdot (-D_i \nabla c_i) + \mathbf{u} \nabla c_i = R_i$$

Here, the first term describes the accumulation of species i over time. The second and third term signify transport due to diffusion and convection, respectively. On the right-hand side, the R_i term represents production or consumption of species i due to chemical reactions.

In order to implement chemical reactions in a model you can set up an application mode to account for each of the reacting species and subsequently enter the rate expressions into the **R** edit field in the **Subdomain Settings** dialog box.

Consider an example where the following reversible reaction is relevant

$$\mathrm{A} \underset{k^r}{\overset{k^f}{\rightleftharpoons}} \mathrm{B}$$

According to Equation 4-9 the reaction rate is formulated as

$$r = k^f c_A - k^r c_B$$

The net rate expressions with respect to species A and B are then

$$R_A = -k^f c_A + k^r c_B$$

$$R_B = k^f c_A - k^r c_B$$

The following steps illustrate how to set up the proper Convection and Diffusion application mode and where to enter the rate expressions in a single-domain geometry:

1 Start COMSOL Multiphysics.

2 In the list of application modes select **Chemical Engineering Module> Mass Transport>Convection and Diffusion**.

3 Type `c_A c_B` in the **Dependent variables** edit field.

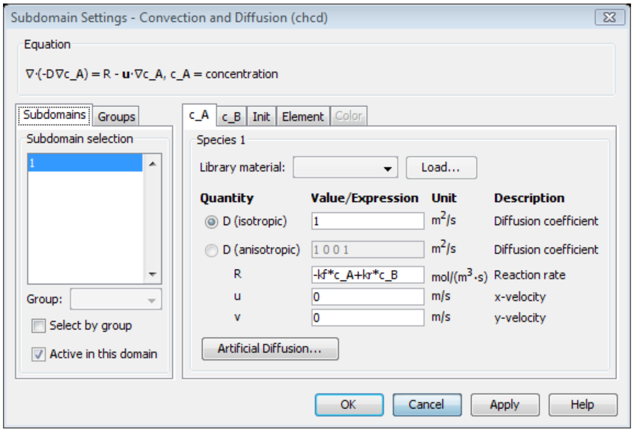
4 Click **OK**.

5 Click the **Rectangle/Square** button on the Draw toolbar and draw an arbitrary rectangle.

6 From the **Physics** menu select **Subdomain Settings**. Select Subdomain 1.

7 On the **c_A** page, type `-kf*c_A+kr*c_B` into the **R** edit field.

8 Go to the **c_B** page and type `kf*c_A-kr*c_B` into the **R** edit field.



Subdomain Settings - Convection and Diffusion (chcd)

Equation

∇·(-D∇c_A) = R - **u**·∇c_A, c_A = concentration

Subdomain selection: 1

Group:

☐ Select by group

☑ Active in this domain

Species 1

Library material:

Quantity	Value/Expression	Unit	Description
● D (isotropic)	1	m^2/s	Diffusion coefficient
○ D (anisotropic)	1 0 0 1	m^2/s	Diffusion coefficient
R	-kf*c_A+kr*c_B	$mol/(m^3 \cdot s)$	Reaction rate
u	0	m/s	x-velocity
v	0	m/s	y-velocity

9 Click **OK**.

Nonisothermal Systems

In addition to the concentration dependency, reaction rates also change with temperature. Arrhenius type expressions typically describe the relation between the rate constants and temperature:

$$k = A\exp\left(-\frac{E}{R_g T}\right) \tag{4-10}$$

In Equation 4-10, A denotes the frequency factor, E the activation energy, and R_g the gas constant (8.314 J/(mol·K)).

To include temperature-dependent rates into a model, simply edit your rate expressions using Arrhenius type rate constants. You can type the full expressions directly into the **Subdomain Settings** dialog of the mass balance application mode. If you have complicated rate expressions you often get a better overview if you enter the equations in the **Expressions** dialog box.

The following steps show how to set up Arrhenius expressions for rate constants, extending the example above:

1 Select the menu item **Options>Expressions>Scalar Expressions**.

2 Enter the following settings:

NAME	EXPRESSION	DESCRIPTION
kf	8.9e7*exp(-167e3/8.314/T)	forward rate constant
kr	3e7*exp(-184e3/8.314/T)	reverse rate constant

3 Click **OK**.

Another temperature effect related to kinetics is the heat developed due to chemical reaction:

$$Q = -\sum_j H_j r_j \tag{4-11}$$

Equation 4-11 sums the heat of reaction, H_j (J/mol), times the reaction rate over all reactions occurring in the system.

To properly account for the effect of the heat source, Q, in a reacting fluid, you need to couple an energy transport application mode to the mass transport application mode. Moving forward with the previous example, add a Convection and Conduction application mode to the existing Convection and Diffusion mode. In the **Subdomain Settings** dialog box for the Convection and Conduction application mode, you can enter the expression for the reaction heat source in the **Q** edit field.

1 From the **Multiphysics** menu, select **Model Navigator**.

2 In the list of application modes select **Chemical Engineering Module> Energy Transport>Convection and Conduction**.

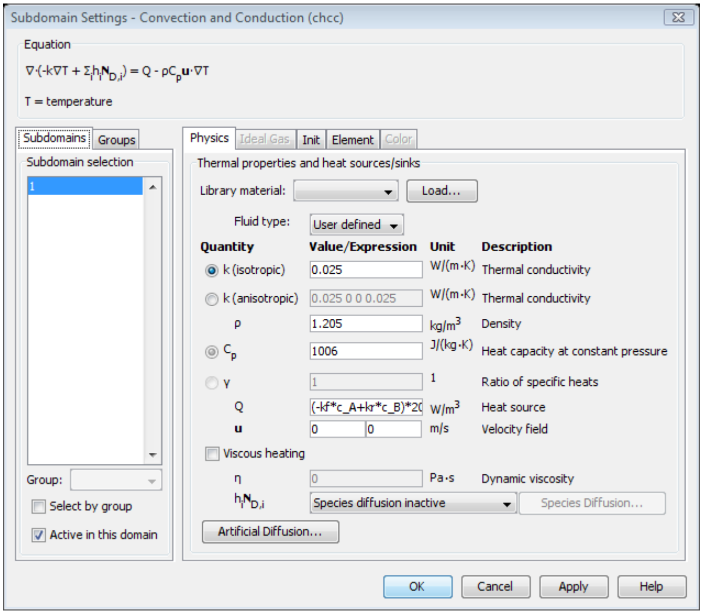
Note that the dependent variable is the temperature, denoted T.

3 Click **Add**, then click **OK**.

4 From the **Physics** menu, select **Subdomain Settings**.

5 Type `(-kf*c_A+kr*c_B)*200e3` into the **Q** edit field.

This expression represents the reaction rate times the heat of reaction.



Subdomain Settings - Convection and Conduction (chcc)

Equation

$\nabla\cdot(-k\nabla T + \Sigma_i h_i \mathbf{N}_{D,i}) = Q - \rho C_p \mathbf{u}\cdot\nabla T$

T = temperature

Subdomain selection: 1

Group:

☐ Select by group

☑ Active in this domain

Thermal properties and heat sources/sinks

Library material:

Fluid type: User defined

Quantity	Value/Expression	Unit	Description
● k (isotropic)	0.025	W/(m·K)	Thermal conductivity
○ k (anisotropic)	0.025 0 0 0.025	W/(m·K)	Thermal conductivity
ρ	1.205	kg/m³	Density
● C_p	1006	J/(kg·K)	Heat capacity at constant pressure
○ γ	1	1	Ratio of specific heats
Q	(-kf*c_A+kr*c_B)*20	W/m³	Heat source
u	0 0	m/s	Velocity field

☐ Viscous heating

Quantity	Value/Expression	Unit	Description
η	0	Pa·s	Dynamic viscosity
$h_i \mathbf{N}_{D,i}$	Species diffusion inactive		

6 Click **OK**.

Note at this point how the two application modes are coupled: The chemical reactions defined in the Convection and Diffusion application mode affect the Convection and Conduction application mode through the heat source of reaction, Q. The evolved heat affects the temperature distribution, which in turn influences the reaction rate constants, k^f and k^r (see Equation 4-10).

Fluid Flow and Reacting Systems

So far the discussion has focused on chemical kinetics coupled to material and energy transport. In addition, chemical engineering problems typically involve the momentum balance equations, which are solved in order to compute the fluid flow field. Adding the effects of fluid flow to the description of reaction systems can lead to strongly coupled systems of equations. To successfully model such systems, much is gained by first analyzing the structure of these couplings.

Assume that the mass and energy transfer application modes exemplified so far are solved together with a momentum equation describing fluid flow. In order to model an incompressible and Newtonian fluid, use the Incompressible Navier-Stokes application mode. The equations set up by this application mode are

$$\rho\frac{\partial \mathbf{u}}{\partial t} - \nabla \cdot \eta(\nabla \mathbf{u} + (\nabla \mathbf{u})^T) + \rho \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p = \mathbf{F} \qquad (4\text{-}12)$$

$$\nabla \cdot \mathbf{u} = 0$$

In Equation 4-12, $\mathbf{u}$ is the velocity, ρ the density, and η the viscosity (kg/(m·s)). The flow equations are solved together with the mass transport equations

$$\frac{\partial c_i}{\partial t} + \nabla \cdot (-D_i \nabla c_i) + \mathbf{u} \nabla c_i = R_i \qquad (4\text{-}13)$$

and the energy transport equation

$$\rho C_p \frac{\partial T}{\partial t} + \nabla \cdot (-k \nabla T + \rho C_p T \mathbf{u}) = Q \qquad (4\text{-}14)$$

Equation 4-13 and Equation 4-14 are two-way coupled through temperature-dependent kinetics. These equations are furthermore influenced by the flow (Equation 4-12), through the velocity field $\mathbf{u}$.

Follow the steps below to add the Incompressible Navier-Stokes application mode to your model and couple the mass and energy transfer modes to the flow field.

1 Choose **Multiphysics>Model Navigator**.

2 In the list of application modes select **Chemical Engineering Module> Momentum Transport>Laminar Flow>Incompressible Navier-Stokes**.

Note that the dependent variables are the velocity components in 2D, denoted u and v, and the pressure, p.

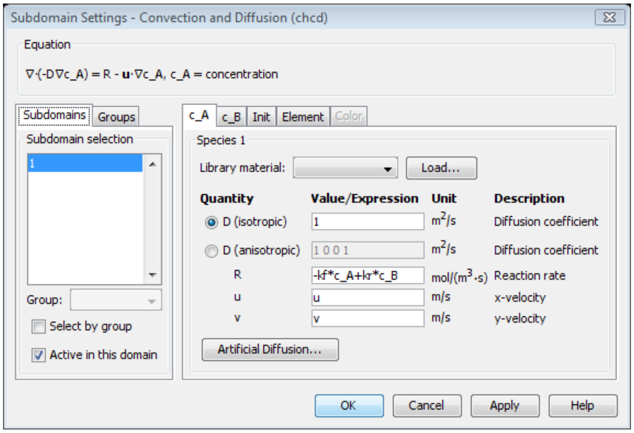
3 Click **Add**, then click **OK**.

4 On the **Multiphysics** menu, select **Convection and Diffusion (chcd)**.

5 From the **Physics** menu, select **Subdomain Settings**.

6 Type u in the **x-velocity** edit field and v in the **y-velocity** edit field.

Now you have coupled the mass transport to the flow by making the convective transport depend on the solution of the momentum equation.



Subdomain Settings - Convection and Diffusion (chcd)

Equation

∇·(-D∇c_A) = R - **u**·∇c_A, c_A = concentration

Subdomain selection: 1

Group:

□ Select by group

☑ Active in this domain

Species 1

Library material:

Quantity	Value/Expression	Unit	Description
● D (isotropic)	1	m^2/s	Diffusion coefficient
○ D (anisotropic)	1 0 0 1	m^2/s	Diffusion coefficient
R	-kf*c_A+kr*c_B	$mol/(m^3 \cdot s)$	Reaction rate
u	u	m/s	x-velocity
v	v	m/s	y-velocity

7 Click **OK**.

8 On the **Multiphysics** menu, select **Convection and Conduction (chcd)**.

9 From the **Physics** menu, select **Subdomain Settings**.

10 Type u in the **x-velocity** edit field and v in the **y-velocity** edit field. You have now coupled the energy transport to the flow.

11 Click **OK**.

In the way just outlined, mass and heat transfer equations have been made dependent on a momentum transport equation by a unidirectional coupling. In an even more complex case, the transport properties of the momentum equation (Equation 4-12) depend on temperature and composition of the reacting fluid, being described by a fully coupled system of equations. To learn more on how to approach coupled multiphysics problems in chemical engineering, see the section “Quick Start Model” on page 16.

Electrochemical Systems

COMSOL Multiphysics and the Chemical Engineering Module are excellent tools for the modeling of electrochemical systems. One of the main issues in the design and operation of electrochemical cells is the current density distribution at the surface of the electrodes and in the electrolyte. With COMSOL Multiphysics you can treat the most advanced cases of current density distribution involving transport of ions through diffusion, migration, and convection. The charge balances can be fully coupled with the kinetics for the electrode reactions involving highly nonlinear reaction kinetics. The transport of species can also be solved with the full description of the fluid flow in the cell.

This section discusses the different application modes that can be used depending on the electrochemical system that you want to model.

Primary and Secondary Current Density Distribution

Primary and secondary current density distribution describes the situation where the mixing of electrolyte is vigorous or where concentration gradients are small, so that ionic migration is the dominating transport mechanism of ionic current. The general material balance for the species in the domain, assuming steady-state conditions, is given by the Nernst-Planck equations

$$\nabla \cdot (-D_i \nabla c_i - z_i u_{mi} F c_i \nabla \phi + c_i \mathbf{u}) = 0$$

where c_i represents the concentration of the ion i, z_i its valence, D_i its diffusivity, u_{mi} its mobility, F denotes Faraday's constant (As/mol), ϕ the ionic potential (V), and $\mathbf{u}$ the velocity vector. The flux terms in the above transport equation describe the flux by diffusion, migration, and convection. Current density can be described through the sum of fluxes of the charged species multiplied by their respective charge and Faraday's constant:

$$\mathbf{i} = -F \sum z_i \mathbf{N}_i$$

where $\mathbf{i}$ is the current density vector (A/m^2). Combining the above two equations, while assuming electroneutrality (which removes the convection term) and negligible concentration gradients (which removes diffusion) leaves:

$$\mathbf{i} = -F\sum -z_i^2 u_{mi} F c_i \nabla\phi$$

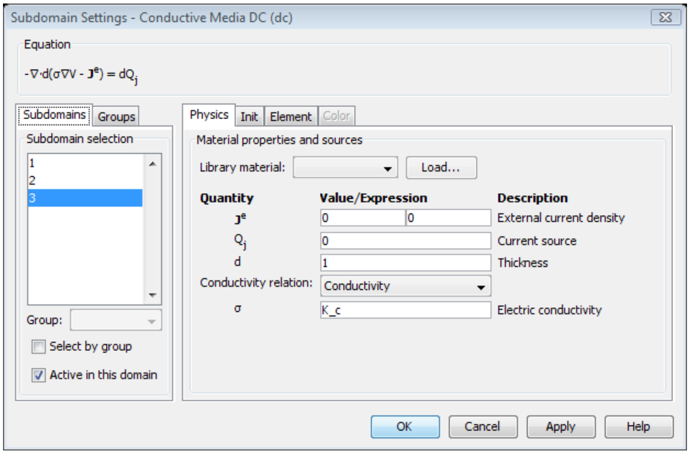
Current density is conserved throughout:

$$\nabla \cdot \mathbf{i} = 0$$

so that combining the valence, ionic mobility, electrolyte composition, and Faraday's constant to a representative conductivity leads to the equation

$$\nabla \cdot (-\kappa \nabla \phi) = 0$$

where κ is the conductivity (S/m). This final equation is basically Ohm's law combined with the conservation of charge and is modeled in COMSOL Multiphysics using the Conductive Media DC application mode (see Figure 4-9).



Subdomain Settings - Conductive Media DC (dc)

Equation

$-\nabla \cdot d(\sigma\nabla V - \mathbf{J}^e) = dQ_j$

Subdomain selection: 3

Group:

☐ Select by group

☑ Active in this domain

Material properties and sources

Library material:

Quantity	Value/Expression		Description
$\mathbf{J}^e$	0	0	External current density
Q_j	0		Current source
d	1		Thickness
Conductivity relation:	Conductivity		
σ	K_c		Electric conductivity

Figure 4-9: The Conductive Media DC application mode's Subdomain Settings dialog allows you to enter the electric (ionic) conductivity of the electrolyte. Setting the external currents to 0 yields the conservation of charges as described by the model equation above.

The boundary conditions for the case of primary current density distribution are valid assuming that the kinetics on the electrode surfaces are extremely fast, which implies that the ionic (electrolyte) potential on these surfaces is constant.

Secondary current distribution takes into account the kinetics of the electrochemical reactions and the ionic potential at the surface of the electrodes cannot be assumed to be constant. In such cases, the overpotential provides the driving force for the electrochemical reactions. The total losses in the cell, at a given total current, consist of the sum of the ohmic losses and the activation overpotential.

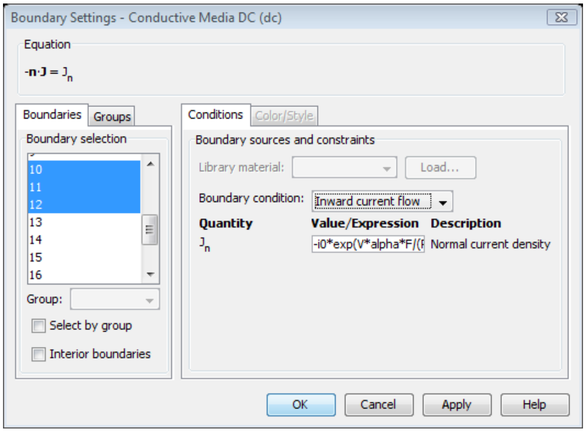
The constant potential at the surface of the electrodes is usually replaced by the Butler-Volmer equation or the Tafel equation. These equations determine the current density perpendicular to the electrode boundary as an exponential function of the activation overpotential according to the following (Butler-Volmer):

$$-\kappa\nabla\phi\cdot\mathbf{n} = i_0\left(\exp\left(\left(\frac{\alpha F(\phi^s-\phi-\Delta\phi^0)}{RT}\right)-\exp\left(-\frac{\alpha F(\phi^s-\phi-\Delta\phi^0)}{RT}\right)\right)\right)$$

where R is the gas constant (J/(mol·K)), α is the symmetry constant for the charged double layer at the electrode surface, ϕ^s is the potential of the electrode, and $\Delta\phi^0$ is the electrode's equilibrium potential relative a reference electrode. The activation overpotential, η(V), is defined according to the following expression:

$$\eta = \phi^s - \phi - \Delta\phi^0$$

You can type both the Butler-Volmer equation and the Tafel equations directly into the **Boundary Settings** dialog box in the user interface. COMSOL Multiphysics detects the nonlinear dependency of the overpotential and selects the proper solver. The figure below shows an example of the Tafel equation used as a boundary condition. In this case, the activation overpotential is given by the variable denoted V.



Boundary Settings - Conductive Media DC (dc)

Equation

-**n**·**J** = J_n

Boundaries | Groups

Boundary selection

10
11
12
13
14
15
16

Group:

□ Select by group

□ Interior boundaries

Conditions | Color/Style

Boundary sources and constraints

Library material: Load...

Boundary condition: Inward current flow

Quantity	**Value/Expression**	**Description**
J_n	-i0*exp(V*alpha*F/(F	Normal current density

Figure 4-10: The current density perpendicular to the surface depends exponentially on the overpotential, here denoted V, according to the Tafel equation.

Tertiary Current Density Distribution

In tertiary current density distribution, you have to consider mass transport through diffusion, convection, and migration— all components of the material balance equation must be included.

A special but important case of tertiary current density distribution is obtained by the so-called supporting electrolyte assumption. This assumption is valid for electrolytes with high concentrations of charged species, which act as "supporting ions" or ions that do not react at the electrode surfaces. The high concentration of the supporting ions is quite common in electrolysis because this gives a high conductivity and reduces ohmic losses and thus the electric energy consumption in the process.

For example, in a sodium sulfate aqueous electrolyte of high concentration containing only a small amount of chloride ions, the sodium and sulfate ions do not usually react at the electrode surface. The larger portion of the current in the bulk of the electrolyte is then carried by the sodium and sulfate ions, the supporting electrolyte, which are present in excess resulting in negligible concentration variations. However, it is not possible to assume that the concentration of the chloride ion is constant, because chloride is present at low concentrations, and it can react through an oxidation reaction to form chlorine at the anode.

In such case, the current density in the bulk electrolyte can be described through Ohm's law as in the primary and secondary cases above, because the supporting electrolyte carries most of the current. The current balance yields:

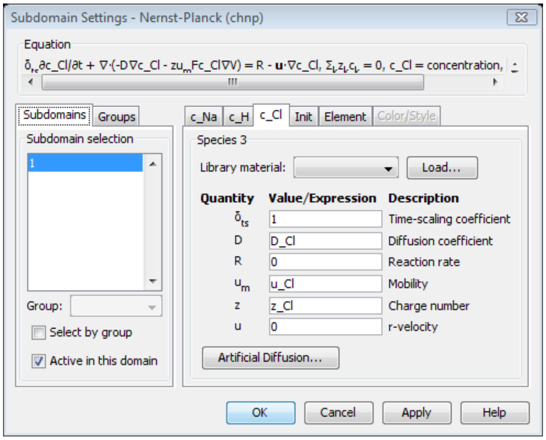
$$\nabla \cdot (-\kappa \nabla \phi) = 0$$

where the reacting species have negligible influence on the ionic conductivity.

The current balance is then coupled to the material balances for the reacting species through the migration term in the Nernst-Planck equation. The description of the material balances for the reacting species accounts for diffusion, migration, and convection:

$$\nabla \cdot (-D_i \nabla c_i - z_i u_{mi} F c_i \nabla \phi + c_i \mathbf{u}) = 0$$

These material balance equations are available in the Nernst-Planck without Electroneutrality application mode in the Chemical Engineering Module. The Subdomain Settings dialog box in Figure 4-11 on page 72 shows that you can identify the coefficient of the equation and just type them in the corresponding edit fields.


Subdomain Settings - Nernst-Planck (chnp)

Equation

$\delta_{ts}\partial c\_Cl/\partial t + \nabla\cdot(-D\nabla c\_Cl - zu_mFc\_Cl\nabla V) = R - \mathbf{u}\cdot\nabla c\_Cl,\ \Sigma_k z_k c_k = 0,\ c\_Cl = \text{concentration},$

Subdomain selection: 1

Group:

☐ Select by group

☑ Active in this domain

Species 3

Library material:

Quantity	Value/Expression	Description
δ_{ts}	1	Time-scaling coefficient
D	D_Cl	Diffusion coefficient
R	0	Reaction rate
u_m	u_Cl	Mobility
z	z_Cl	Charge number
u	0	r-velocity

Figure 4-11: The Subdomain Settings dialog box for a solution consisting of three reacting species.

The concentration variations affect the electrochemical kinetics at the electrode surfaces through an expression that accounts for variations in composition and ionic potential at the surface of the electrodes, for example:

$$-\kappa\nabla\phi\cdot n = i_0\left(\frac{c}{c^0}\exp\left(\left(\frac{\alpha F(\phi^s-\phi-\Delta\phi^0)}{RT}\right)-\exp\left(-\frac{\alpha F(\phi^s-\phi-\Delta\phi^0)}{RT}\right)\right)\right)$$

where c is the concentrations of the reacting species and c^0 denotes its reference concentration, usually set to the concentration for the measured equilibrium potential. The total losses in the cell for tertiary current density distribution consist of ohmic losses, activation overpotential, and concentration overpotential. The concentration overpotential arises when the concentration of reacting species deviates from the reference concentration.

In the absence of a supporting electrolyte, the material balances for all charged species have to be solved. This yields one equation for each dependent variable, in this case the species concentrations. The last variable is the ionic potential, which requires one additional equation. This equation is usually the electroneutrality condition, which states that the sum of all charges is zero in the bulk of an aqueous electrolyte:

$$\sum z_i c_i = 0$$

The material balances and the electroneutrality condition are automatically defined by the Nernst-Planck application mode.

Tubular Reactors and Pseudo Modes

Simulation of chemical reactors may well involve solving a set of coupled transport equations. This can in turn lead to models that are challenging to solve due to convergence issues as well as computer memory requirements.

When modeling tubular reactors it is often possible to reduce the model size without losing accuracy, by making use of symmetry. The Chemical Engineering Module supplies a number of application modes that allow you to easily and efficiently exploit reactor symmetry. These modes are outlined below.

Axial Symmetry

Taking advantage of axial symmetry in tubular geometries is an intuitive approach to reducing model size. An example is illustrated in Figure 4-12, where a straight pipe with constant cross-sectional area reduces to a rectangular geometry in a 2D axisymmetric representation.

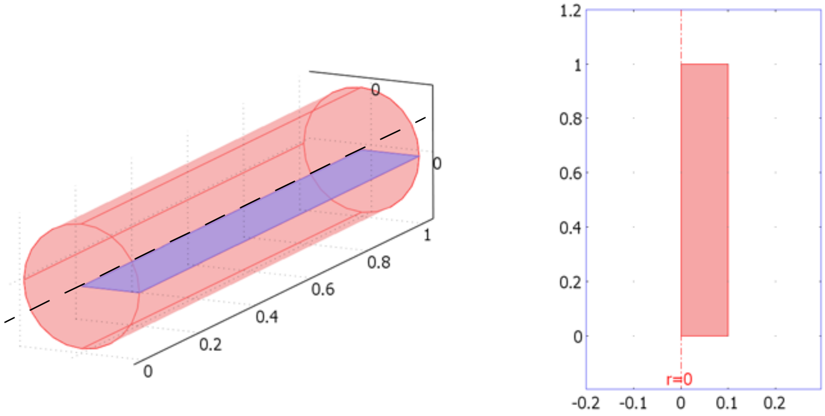
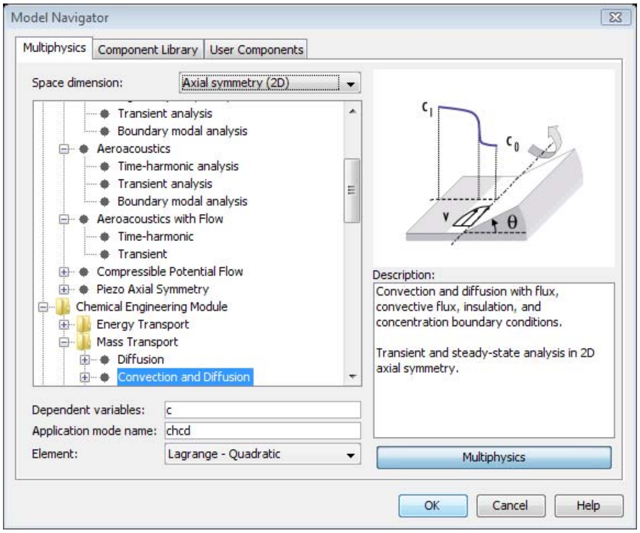


Figure 4-12: A straight pipe with constant cross-sectional area can be reduced to a rectangular geometry in 2D axisymmetry.

You select axial symmetry from the **Space dimension** list in the **Model Navigator**. Typically, you choose the space dimension of your model at the very beginning of the modeling process or when you want to add a new geometry to an existing model. Once the Space dimension has been selected, you are free to add any application modes from the **Application Modes** tree. Note that no matter what space dimension you choose, the application mode dialog boxes have similar contents. The associated transport equations, however, are automatically transformed to the proper coordinate system representation.



Space dimension: Axial symmetry (2D)

Dependent variables: c

Application mode name: chcd

Element: Lagrange - Quadratic

Description:

Convection and diffusion with flux, convective flux, insulation, and concentration boundary conditions.

Transient and steady-state analysis in 2D axial symmetry.

Figure 4-13: You can select Axial symmetry in 2D or 1D from the Space dimension list in the Model Navigator.

Consider an example where a chemical, A, reacts under isothermal conditions while flowing through a tubular reactor. The rate of consumption is given by:

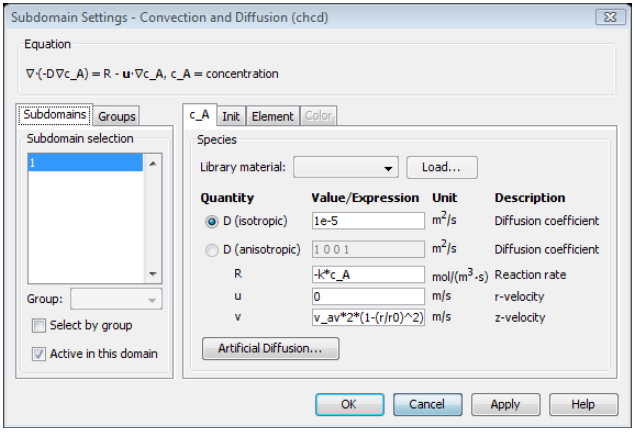
$$R_A = -kc_A$$

The reactor geometry is illustrated in Figure 4-12. Assume further that the flow is laminar and fully developed. Under these conditions, the reactor can be simulated by setting up a mass transport application mode and providing an expression for the fluid flow in the axial direction. The velocity profile for fully developed laminar flow in a pipe can be written as

$$v = 2v_{\mathrm{av}}\left(1-\left(\frac{r}{r_0}\right)^2\right) \qquad (4\text{-}15)$$

where v_{av} is the average velocity in the axial direction and r_0 is the radius of the pipe.

The **Subdomain Settings** for a Convection and Diffusion application mode, applying to the 2D axisymmetric reactor geometry is shown in Figure 4-14.



Subdomain Settings - Convection and Diffusion (chcd)

Equation

∇·(-D∇c_A) = R - **u**·∇c_A, c_A = concentration

Subdomains | Groups

Subdomain selection: 1

Group:

☐ Select by group

☑ Active in this domain

c_A | Init | Element | Color

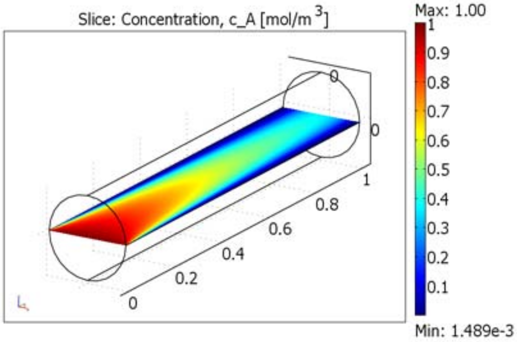
Species

Library material: Load...

Quantity	Value/Expression	Unit	Description
● D (isotropic)	1e-5	m^2/s	Diffusion coefficient
○ D (anisotropic)	1 0 0 1	m^2/s	Diffusion coefficient
R	-k*c_A	mol/(m^3·s)	Reaction rate
u	0	m/s	r-velocity
v	v_av*2*(1-(r/r0)^2)	m/s	z-velocity

Figure 4-14: Type the reaction rate, velocity profile, and diffusivity in the Subdomain Settings dialog box.

Solving the model for both the 3D and the 2D axisymmetric geometries produces the results in Figure 4-15. A slice plot along the axial direction in the 3D reactor shows an equivalent concentration distribution as the surface plot of the axisymmetric model.



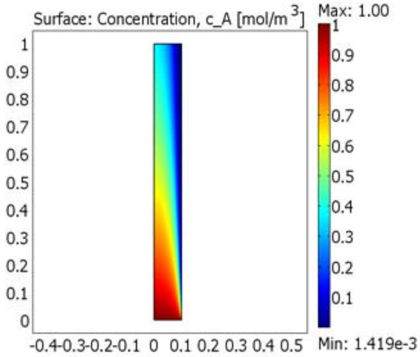
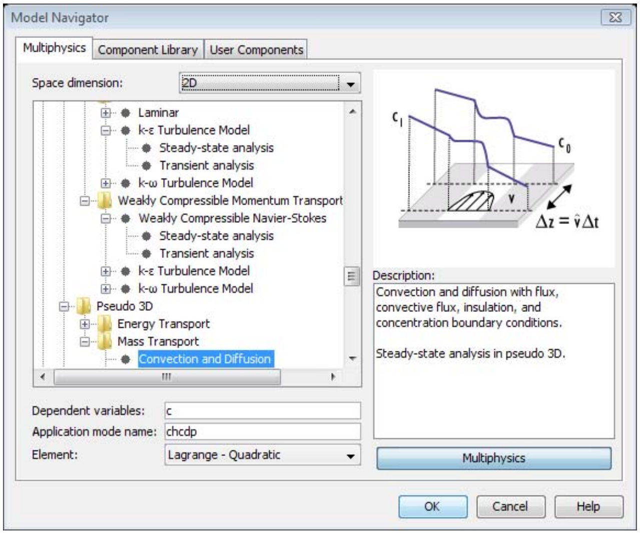


Figure 4-15: Reactant concentration in the 3D reactor geometry (top) and the corresponding 2D axisymmetric geometry (bottom). Concentration distributions in the two representation are equivalent.

Pseudo Modes

The Pseudo application modes available in the Chemical Engineering Module work under assumptions that can further reduce the problem size and computational time for tubular reactor models.

Pseudo modes are suitable for modeling stationary systems where convection is constant and dominant in one direction. As a result, material transport by diffusion is neglected in the main direction of the flow. This makes it is possible to describe the modeled geometry by extrusion of the reactor cross section. Provided that the velocity profile can be expressed in the cross section, you can replace the space variable along the direction of the flow with time, t. This means that a time increment represents a displacement along the direction of the flow. Using this approach, the Pseudo modes allow you to model a 3D system in 2D or a 2D system in 1D. Pseudo application modes exist for mass transport as well as for energy transport.



Space dimension: 2D

Description:
Convection and diffusion with flux, convective flux, insulation, and concentration boundary conditions.

Steady-state analysis in pseudo 3D.

Dependent variables: c

Application mode name: chcdp

Element: Lagrange - Quadratic

Figure 4-16: Pseudo 3D and Pseudo 2D modes are available for both mass and energy transport.

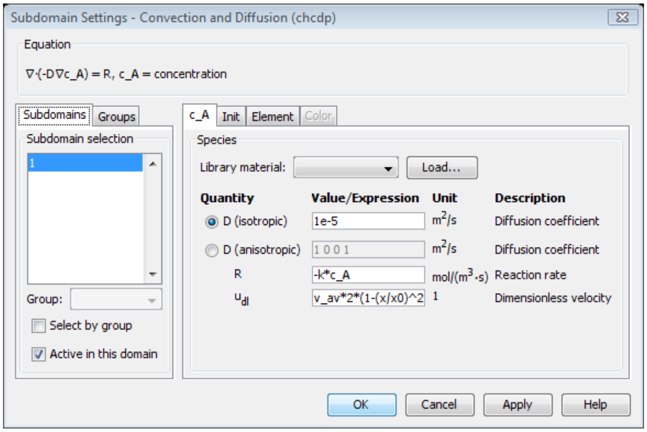
The tubular reactor described in the example above runs at steady state and has a constant cross section. Furthermore, the flow in the reactor is fully developed and laminar, with the velocity in the axial direction expressed by an analytical expression (Equation 4-15). Therefore, an alternative to modeling the full 3D reactor would be to simulate the time-dependent behavior in reactor cross sections, using a Pseudo 3D mode.

In the Convection and Diffusion application mode, the pseudo formulation of the mass transport equation is written as

$$u_{\mathrm{dl}}\frac{\partial c_i}{\partial t} + \nabla\cdot(-D_i\nabla c_i) = R_i \qquad (4\text{-}16)$$

Here, u_{dl} represents the velocity profile in the reactor cross section, as given by Equation 4-15.

Figure 4-19 shows the **Subdomain Settings** dialog of the Convection and Diffusion application mode where the tubular reactor is modeled with a Pseudo 3D mode.



Subdomain Settings - Convection and Diffusion (chcdp)

Equation

$\nabla\cdot(-D\nabla c\_A) = R$, c_A = concentration

Subdomain selection: 1

Group:

☐ Select by group

☑ Active in this domain

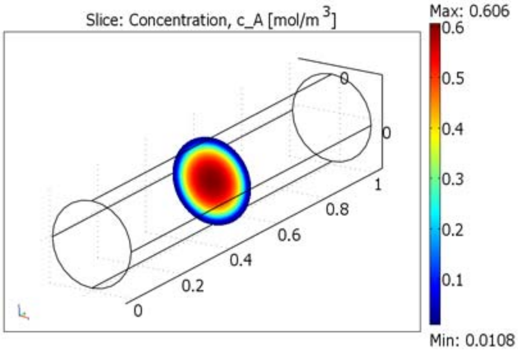
Species

Library material:

Quantity	Value/Expression	Unit	Description
● D (isotropic)	1e-5	m^2/s	Diffusion coefficient
○ D (anisotropic)	1 0 0 1	m^2/s	Diffusion coefficient
R	-k*c_A	$mol/(m^3\cdot s)$	Reaction rate
u_{dl}	v_av*2*(1-(x/x0)^2	1	Dimensionless velocity

Figure 4-17: Type in the reaction rate, velocity profile, and diffusivity into the Subdomain Settings dialog box.

Once solved, the Pseudo 3D model generates results for the reactor cross section as a function of time. Because the fluid velocity and the reactor length are known, the results can readily be compared with the full 3D model. Figure 4-18 shows a concentration distribution of *A* in slice plot half-way down the 3D reactor. The same results are produced by the pseudo 3D model after half the simulated time, 0.5 seconds.



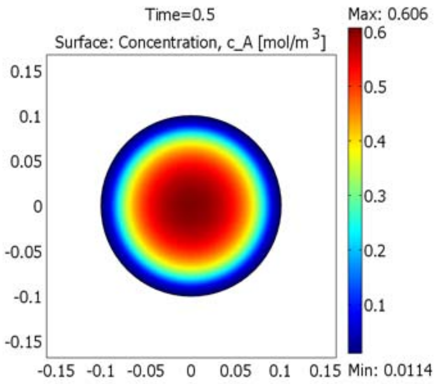
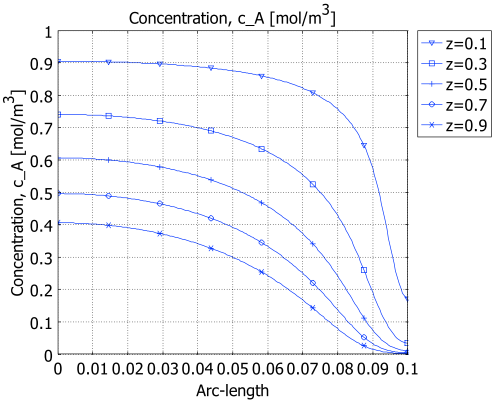


Figure 4-18: Reactant concentration in the 3D reactor model (top) and the corresponding Pseudo 3D model (bottom). Concentration distributions in the two representations are equivalent.

The process of reducing model size can be advanced even further by realizing that the 2D axisymmetric model in turn can be represented in a 2D Pseudo mode. The Pseudo 2D model generates results as a function of the radial position in the reactor, results that change with the simulated time. Because the velocity in the axial direction is known, each time coordinate corresponds to specific position in the reactor.

Figure 4-19 displays radial concentration profiles as function of axial position in the reactor. The top graph originates from the 2D axisymmetric model, and was created by plotting results in the radial direction at positions z = 0.1, 0.3, 0.5, 0.7, and 0.9 m. The bottom graph shows results from the Pseudo 2D model, evaluated at times t = 0.1, 0.3, 0.5, 0.7, and 0.9 s. Again, results are equivalent.



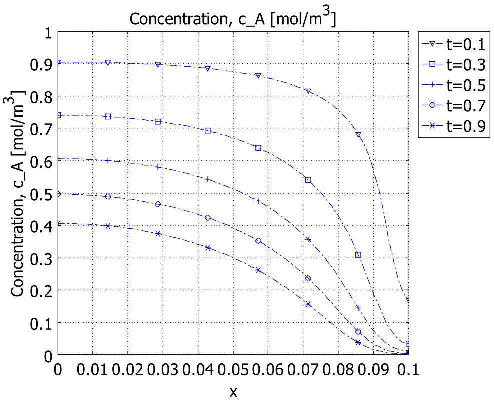


Figure 4-19: Radial concentration profiles of A as a function of axial position in the reactor. Results from the 2D axisymmetric model (top) and the corresponding Pseudo 2D model (bottom) are equivalent.

To summarize, it has been illustrated how you can use symmetry considerations and Pseudo modes to reduce the model size while retaining accurate results. The following table lists the number of elements used in each of the reactor examples discussed, serving as an indicator of model size:

MODEL	ELEMENTS
3D	6498
2D axisymmetric	2484
Pseudo 3D	754
Pseudo 2D	15

From Ideal Reactors to Time- and Space-Dependent Models

This section highlights the link between COMSOL's Reaction Engineering Lab software and COMSOL Multiphysics. Reaction Engineering Lab sets up and solves models of perfectly mixed batch systems and plug-flow reactors, models which later can be exported to application modes of the Chemical Engineering Module. In the COMSOL Multiphysics environment, the models are taken to the next level of detail through simulation of the time- and space-dependent system.

Introduction

The Reaction Engineering Lab is COMSOL's modeling environment for ideal reactor systems, featuring predefined reactor types such as perfectly mixed batch and semi-batch systems, continuous stirred tank reactors (CSTRs), and plug-flow reactors. Working with reaction kinetics and chemical engineering problems is made easy as the software also features automatically generated kinetics, as well as the capability to validate and calibrate reaction models through parameter estimation.

In the Chemical Engineering Module you set up and solve time- and space-dependent models of reacting systems, typically including the effects of flow, heat, and mass transfer. COMSOL Multiphysics furthermore enables you to include any additional physics to your model by adding the corresponding application modes.

The Reaction Engineering Lab integrates directly with the COMSOL Multiphysics environment, so that you can effortlessly move from ideal reactor models to simulating time- and space-dependent systems. The combination of Reaction Engineering Lab and COMSOL Multiphysics forms a powerful tool for chemical engineering simulation, allowing you to cover a wide range of modeling applications:

- Kinetic modeling and parameter estimation
- Ideal reactor analysis and design
- Detailed modeling of time- and space-dependent reacting systems

Ideal Reactors and Systems with Time- and Space-Dependence

This section starts by looking at the equations used to solve time- and space-dependent systems compared to the corresponding equations of ideal reactor models. Mass transport serves to illustrate the link between the equations and the implications when solving them.

Mass transport in a general time- and space-dependent system can be represented by the following equation

$$\frac{\partial c_i}{\partial t} + \nabla \cdot (-D_i \nabla c_i + \mathbf{u} c_i) = R_i \tag{4-17}$$

The first term of Equation 4-17 corresponds to accumulation of the species *i*. The second term signifies the difference of mass entering and leaving a volume element, due to convection and diffusional transport. Finally, the right-hand side of the equation displays the production of species *i*, typically due to chemical reaction.

You can set up Equation 4-17 in the Convection and Diffusion application mode and solve it in COMSOL Multiphysics.

When working with ideal reactors, a number of assumptions are imposed on the general mass balance, given by Equation 4-17. For instance, ideal batch reactors are assumed to be perfectly mixed, and hence the space-dependent part of Equation 4-17 can be disregarded. This simplifies the mass balance to

$$\frac{dc_i}{dt} = R_i \tag{4-18}$$

The plug-flow reactor is another well established model. It describes a reacting fluid in a tubular reactor. This reactor is assumed to run at steady state. The fluid is furthermore considered to be perfectly mixed in the radial as well as angular direction. Finally, the fluid velocity is only a function of the axial position in the reactor. Simplifying Equation 4-17 leads to

$$\frac{d(c_i u_x)}{dx} = R_i \tag{4-19}$$

Equation 4-18 and Equation 4-19 are set up when using the Batch and Plug-flow reactor types in Reaction Engineering Lab, respectively.

As outlined above, the mathematical complexity of mass balance descriptions vary considerably. The assumptions of ideal reactors simplify the model equations from a partial differential equation (Equation 4-17) to ordinary differential equations (Equation 4-18 and Equation 4-19). This means that although the level of detail is reduced in the ideal reactors, so is the demand on computational resources to solve the problems. Furthermore, while convergence issues are commonplace when trying to solve PDEs or systems of coupled PDEs, they are rare for systems of ODEs.

Another observation is that the source term due to reaction, R_i, is the same in both the time- and space-dependent equations and the ideal reactor equations. This suggests that an efficient modeling approach involves investigating reaction kinetics in an ideal reactor environment first, then transferring the kinetic expressions (R_i) to a corresponding time- and space-dependent system. This approach is fully supported through the combination of Reaction Engineering Lab and COMSOL Multiphysics.

The following work flow provides a flexible framework for solving a wide range of problems in chemical engineering:

1. Reaction Engineering Lab
   - Explore reaction kinetics and perform parameter estimation
   - Ideal reactor analysis and design
   - Prepare for detailed modeling in time- and space-dependent systems by exploring time scales and concentration and temperature gradients.
   - Export the reaction model to COMSOL Multiphysics
2. COMSOL Multiphysics
   - Solve the time- and space-dependent problem
   - Extend the model with additional physics couplings

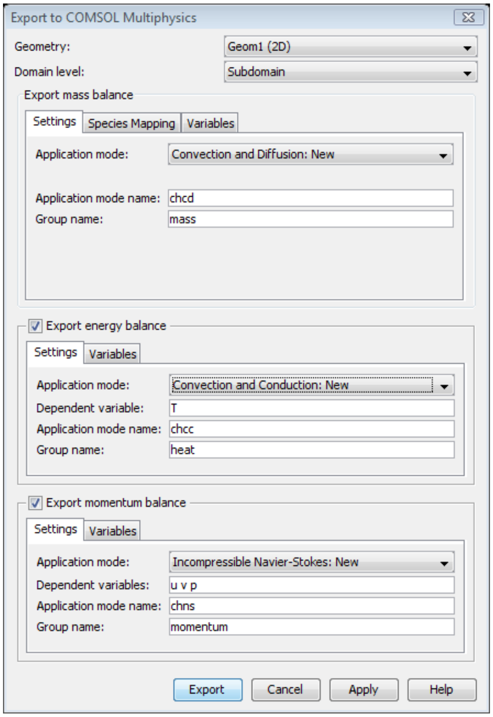
Exporting from Reaction Engineering Lab to COMSOL Multiphysics

The Reaction Engineering Lab lets you to transfer the chemistry and physics of a reaction model to the following application modes of the Chemical Engineering Module:

- Diffusion
- Convection and Diffusion
- Maxwell-Stefan Diffusion and Convection
- Nernst-Planck
- Conduction
- Convection and Conduction

- Incompressible Navier-Stokes
- Weakly Compressible Navier-Stokes

A mass transfer application mode can be exported by itself, together with a heat transfer or a momentum transfer application mode, or with both.



Export to COMSOL Multiphysics

Geometry: Geom1 (2D)

Domain level: Subdomain

Export mass balance

Application mode: Convection and Diffusion: New

Application mode name: chcd

Group name: mass

☑ Export energy balance

Application mode: Convection and Conduction: New

Dependent variable: T

Application mode name: chcc

Group name: heat

☑ Export momentum balance

Application mode: Incompressible Navier-Stokes: New

Dependent variables: u v p

Application mode name: chns

Group name: momentum

Figure 4-20: Select application modes in the Reaction Engineering Lab's Export to COMSOL Multiphysics dialog box.

In the export process you can create a new Multiphysics model, create a new application mode within an existing Multiphysics model, or export into an existing application mode.

Exports create groups that are typically associated at the subdomain level of the application mode. You can also export to boundaries, which is relevant for surface reactions.

REACTION KINETICS

The Reaction Engineering Lab automatically generates kinetics expressions when you type chemical reaction formulas into its graphical user interface. The rates for each individual reaction is formulated according to the mass action law

$$r_j = k_j^f \prod_{i \in \mathrm{react}} c_i^{-\nu_{ij}} - k_j^r \prod_{i \in \mathrm{prod}} c_i^{\nu_{ij}} \qquad (4\text{-}20)$$

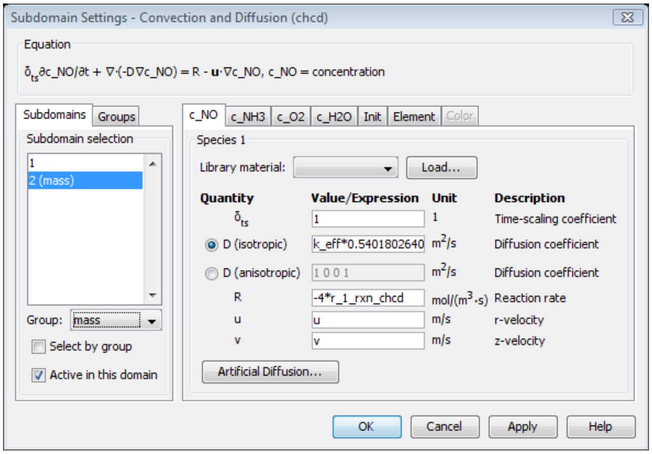
where c_i denotes species concentrations, k_j reaction rate constants, and ν_{ij} the stoichiometric coefficients.

The net rates of each reacting species, affecting mass balance equations, as totalled over all reactions, are also set up automatically

$$R_i = \sum_j \nu_{ij} r_j \qquad (4\text{-}21)$$

As the number of reactions increase, the number and complexity of the R_i terms rapidly increase. Because the net rates, R_i remain the same, no matter if an ideal reactor or time- and space-dependent system is considered, it makes sense to make use of these automatically generated expressions in COMSOL Multiphysics as well.

The export from Reaction Engineering Lab automatically sets up mass balance application modes in COMSOL Multiphysics, corresponding to the reacting species, and enters the reaction rate terms in the **R** edit field.



Subdomain Settings - Convection and Diffusion (chcd)

Equation

$\delta_{ts}\partial c\_NO/\partial t + \nabla\cdot(-D\nabla c\_NO) = R - \mathbf{u}\cdot\nabla c\_NO$, c_NO = concentration

Subdomain selection: 1, 2 (mass)

Group: mass

□ Select by group

☑ Active in this domain

Species 1

Library material:

Quantity	Value/Expression	Unit	Description
δ_{ts}	1	1	Time-scaling coefficient
● D (isotropic)	k_eff*0.5401802640	m^2/s	Diffusion coefficient
○ D (anisotropic)	1 0 0 1	m^2/s	Diffusion coefficient
R	-4*r_1_rxn_chcd	$mol/(m^3\cdot s)$	Reaction rate
u	u	m/s	r-velocity
v	v	m/s	z-velocity

Figure 4-21: Exporting reaction models from Reaction Engineering Lab automatically sets up the appropriate application modes and fills in the rate expressions in the R edit field.

THERMODYNAMIC PROPERTIES

Reaction Engineering Lab also allows you to set up and solve coupled mass and energy balances in ideal reactor modeling. In doing so, you specify a number of thermodynamic properties for each of the reacting species, such as:

- Species heat capacity
- Species enthalpy

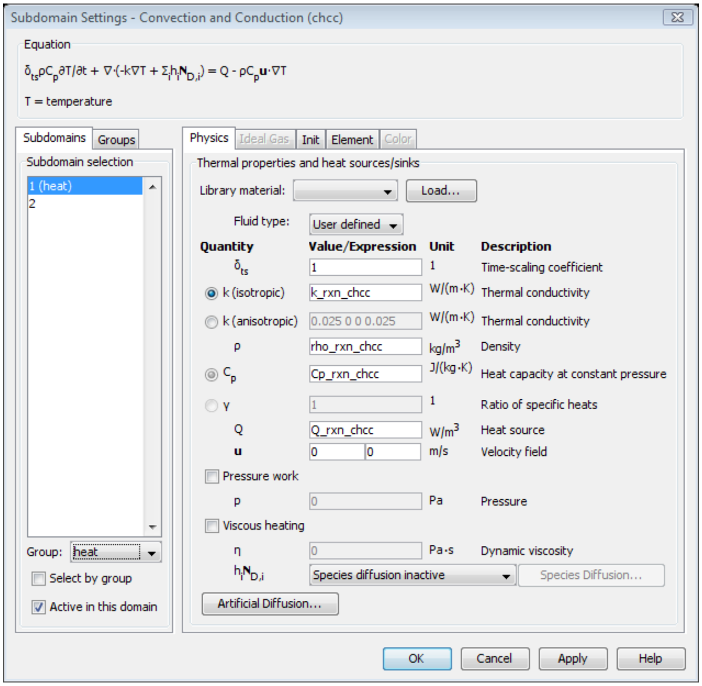
These properties are automatically used in the energy balance equation of the reactor model by defining the reaction thermodynamics. Such automatically generated expressions includes the heat of reaction

$$H_j = \sum_i \nu_{ij} h_i \qquad (4\text{-}22)$$

and the net heat source of reaction

$$Q = -\sum_j H_j r_j \qquad (4\text{-}23)$$

The thermodynamic properties and expressions just mentioned are immediately useful in heat transfer application modes in COMSOL Multiphysics. Once again, exporting from Reaction Engineering Lab to COMSOL Multiphysics sets up the selected application mode and enters the corresponding values of expressions in its **Cp** and **Q** edit fields.



Subdomain Settings - Convection and Conduction (chcc)

Equation

$\delta_{ts}\rho C_p\partial T/\partial t + \nabla\cdot(-k\nabla T + \Sigma_i h_i \mathbf{N}_{D,i}) = Q - \rho C_p \mathbf{u}\cdot\nabla T$

T = temperature

Subdomain selection: 1 (heat) (selected), 2

Group: heat

□ Select by group

☑ Active in this domain

Thermal properties and heat sources/sinks

Library material:

Fluid type: User defined

Quantity	Value/Expression	Unit	Description
δ_{ts}	1	1	Time-scaling coefficient
● k (isotropic)	k_rxn_chcc	W/(m·K)	Thermal conductivity
○ k (anisotropic)	0.025 0 0 0.025	W/(m·K)	Thermal conductivity
ρ	rho_rxn_chcc	kg/m^3	Density
● C_p	Cp_rxn_chcc	J/(kg·K)	Heat capacity at constant pressure
○ γ	1	1	Ratio of specific heats
Q	Q_rxn_chcc	W/m^3	Heat source
u	0 0	m/s	Velocity field
□ Pressure work			
p	0	Pa	Pressure
□ Viscous heating			
η	0	Pa·s	Dynamic viscosity
$h_i\mathbf{N}_{D,i}$	Species diffusion inactive		

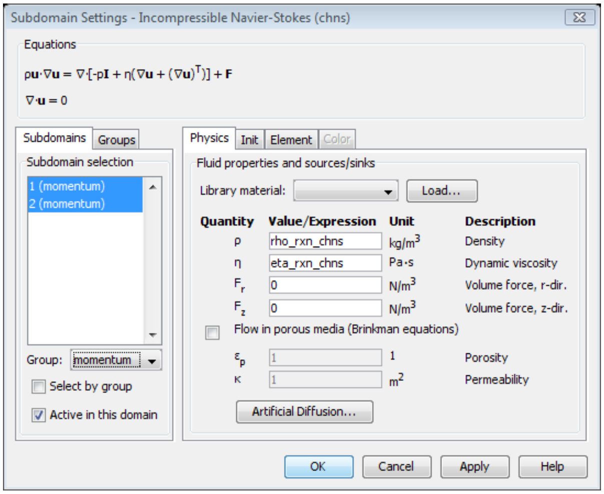
Figure 4-22: Exports including an energy balance automatically sets up the application mode and fills in the k, ρ, C_p, and Q edit fields.

TRANSPORT PROPERTIES

Reaction Engineering Lab can furthermore set up expressions for the transport properties of gases and liquids, expressed as functions of pressure, composition and temperature. Such properties include:

- Diffusivity
- Viscosity
- Thermal conductivity
- Density

In contrast to the reaction kinetics and thermodynamic properties, the transport properties listed above do not enter equations solved in Reaction Engineering Lab. However, with respect to the export functionality, they work in a similar fashion; expressions for diffusivities end up in the **Diffusivity** edit fields of mass transfer application modes; thermal conductivity expressions are entered into the corresponding edit field of heat transfer application modes; and viscosity and density expressions are exported to the appropriate edit fields in the momentum transfer application modes.



Subdomain Settings - Incompressible Navier-Stokes (chns)

Equations

$\rho\mathbf{u}\cdot\nabla\mathbf{u} = \nabla\cdot[-p\mathbf{I} + \eta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)] + \mathbf{F}$

$\nabla\cdot\mathbf{u} = 0$

Subdomains | Groups

Subdomain selection

1 (momentum)
2 (momentum)

Group: momentum

☐ Select by group

☑ Active in this domain

Physics | Init | Element | Color

Fluid properties and sources/sinks

Library material: Load...

Quantity	Value/Expression	Unit	Description
ρ	rho_rxn_chns	kg/m^3	Density
η	eta_rxn_chns	Pa·s	Dynamic viscosity
F_r	0	N/m^3	Volume force, r-dir.
F_z	0	N/m^3	Volume force, z-dir.

☐ Flow in porous media (Brinkman equations)

ε_p	1	1	Porosity
κ	1	m^2	Permeability

Artificial Diffusion...

OK | Cancel | Apply | Help

Figure 4-23: Exports including a momentum balance automatically sets up the application mode and fills in the ρ *and* η *edit fields.*

You can read more about the features of Reaction Engineering Lab and its coupling to COMSOL Multiphysics in the *COMSOL Reaction Engineering Lab User's Guide*.

Multiphase Flow

The Chemical Engineering Module includes several multiphase flow application modes. The Level Set Two-Phase Flow application modes are suitable when you want to study the flow of two immiscible fluids in detail, that is, when you want to keep track of the exact position of the moving fluid-fluid interface.

When you are interested in the macroscopic behavior of a flow, consisting of many bubbles, droplets, or solid particles moving through a liquid, you can use either the Bubbly Flow or the Mixture Model application modes.

All multiphase flow application modes are available either with or without a turbulence model.

The following sections describe how you can use these application modes.

The Level Set Two-Phase Flow Application Modes

With the Level Set Two-Phase Flow application modes, you solve for the velocity, pressure, and the level set variable. The fluid interface is given by the 0.5 contour of the level set variable.

SUBDOMAIN SETTINGS

In the **Subdomain Settings** dialog box, you define two fluids, Fluid 1 and Fluid 2. For each of the fluids, you give its density and viscosity. Here, you can also set the surface tension coefficient and the gravity vector. On the **Init** tab you specify which fluid each subdomain initially is filled with.

Appropriate Values for the Level Set Parameters γ *and* ε

In addition to the physical parameters, you also need to assign values to the two level set parameters, ε and γ. A suitable default value of the parameter, ε, is the maximum mesh element size in each subdomain. You can also use the same value in all subdomains. ε should then be equal to the maximum mesh element size in all subdomains. For γ, you can use an approximate value for the maximum speed of the flow. Because the speed of the flow is not always known in advance, you might have to perform several calculations to find the optimal γ.

BOUNDARY CONDITIONS

All boundary conditions available for single-phase flow, except for the boundary types **Open boundary** and **Stress**, are also available for the Level Set Two-Phase Flow application modes. In some instances, you might have to assign boundary conditions to the level set variable. There are also two boundary conditions that are only available for the Level Set Two-Phase Flow application mode, namely the conditions **Wetted wall** and **Initial fluid interface** for interior boundaries.

Inlets

For inlet boundaries, you need to specify the value of the level set function. If Fluid 1 flows in through the boundary, set the level set function to zero, otherwise, set it to one.

Walls, Outlets, and Symmetry Boundaries

For walls, outlets, and symmetry boundaries, you do not have to set any boundary condition on the level set function. For walls in contact with the fluid interface, you can use the **Wetted wall** boundary condition. You then need to specify the contact angle θ and the slip length β. Figure 4-24 illustrates the definition of these two quantities.

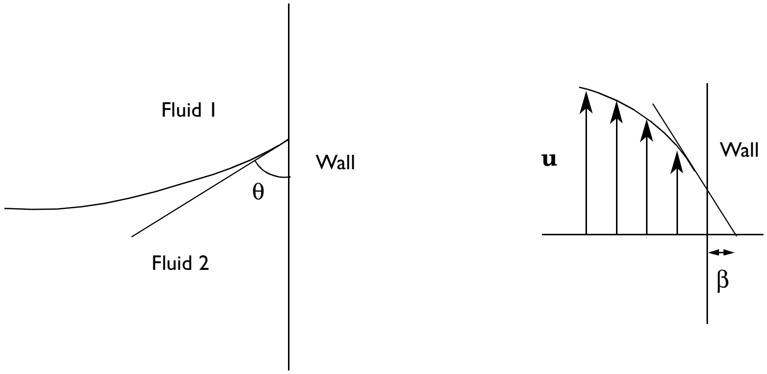


Figure 4-24: Definition of the contact angle θ *and the slip length* β *for the Wetted wall boundary condition.*

Interior Boundaries

For interior boundaries that initially coincide with the fluid interface, use the **Initial fluid interface** boundary condition.

SOLUTION PROCEDURE

Before you can start the fluid-flow calculation, you need to initialize the level set function. This section describes in detail how to proceed to solve a model, including the initialization of the level set function and the full multiphase flow calculation.

After you have drawn the geometry, created the mesh, and applied suitable subdomain settings and boundary conditions, perform the following steps.

First, determine a suitable time scale for the initialization of the level set function:

1 From the **Solve** menu, select **Get Initial Value**.

2 Click the **Plot Parameters** button on the Main toolbar. Type `5*epsilon_chns/gamma_chns` in the **Expression** edit field and click **OK**.

The largest value shown in the plot is a suitable time for the initialization:

1 Click the **Solver Parameters** button on the Main toolbar. In the **Times** edit field, replace the expression by the largest value shown in the plot. Click **OK**.

2 Click the **Solve** button on the Main toolbar.

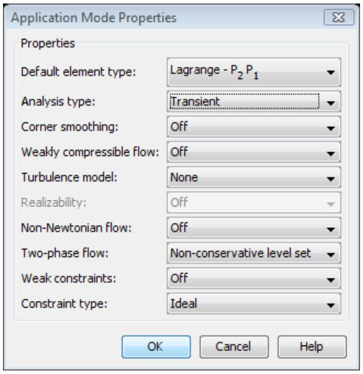
3 To plot the initial level set function, click the **Plot Parameters** button on the Main toolbar. Type `phi` in the **Expression** edit field and click **OK**.

4 Click the **Solver Manager** button on the Main toolbar.

5 Click the **Store Solution** button at the bottom of the dialog box. Select the last time only and click **OK**.

6 In the **Initial value** area, click the **Stored solution** option button and select the stored time in the corresponding list. Click **OK**.

7 From the **Physics** menu, select **Properties** to open the **Application Mode Properties** dialog box. Select **Transient** from the **Analysis type** list. Click **OK**.



Application Mode Properties

Properties

Default element type: Lagrange - P_2 P_1

Analysis type: Transient

Corner smoothing: Off

Weakly compressible flow: Off

Turbulence model: None

Realizability: Off

Non-Newtonian flow: Off

Two-phase flow: Non-conservative level set

Weak constraints: Off

Constraint type: Ideal

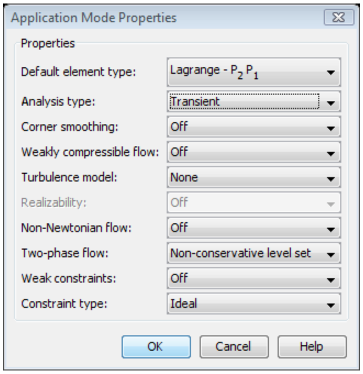
Figure 4-25: Note that after you have initialized the level set function, you must switch to the transient analysis type.

8 Open the **Solver Parameters** dialog box and type in a suitable time range in the **Times** edit field. Click **OK**.

9 Click the **Solve** button to start the fluid-flow calculation.

TURBULENCE MODELING AND ALTERNATIVE FLUID FLOW MODELS

By right-clicking the Level Set Two-Phase Flow application mode in the Model tree and selecting **Properties**, you can open the **Application Mode Properties** dialog box (see Figure 4-26). Here, you can switch on or off a turbulence model, include weak compressibility effects, or switch to a single-phase flow model. For axisymmetric models, you can also switch on Swirl flow.


Application Mode Properties

Properties

Default element type: Lagrange - $P_2 P_1$

Analysis type: Transient

Corner smoothing: Off

Weakly compressible flow: Off

Turbulence model: None

Realizability: Off

Non-Newtonian flow: Off

Two-phase flow: Non-conservative level set

Weak constraints: Off

Constraint type: Ideal

Figure 4-26: In the Application Mode Properties dialog box, you can switch on or off a turbulence model and include weak compressibility effects.

CONVERGENCE PROBLEMS

If you run into convergence problems, check the following:

- Is γ approximately as large as the maximum speed of the flow?
- Is ε at least as large as the maximum value of the mesh size h in each subdomain?
- Is the Cell Reynolds number smaller than 2? If not, you must add artificial diffusion. Open the **Subdomain Settings** dialog box, go to the **Sources/Sinks** tab and click the **Artificial Diffusion** button. For the Navier-Stokes equations, select the **Streamline diffusion** check box and select **Anisotropic Diffusion** from the list. Click **OK** twice.
- If the effect of surface tension is small, set the surface tension coefficient σ to zero.
- Make sure you have initialized the level set function as described in the section “Solution Procedure” on page 92.
- If you are unable to start a transient turbulent simulation, set the initial value for the turbulence variables `logd` and `logk` to −16.
- Verify that there is no inflow through any boundaries that has been specified as outlet boundaries. For boundaries that are both inlets and outlets, specify it as an inlet.

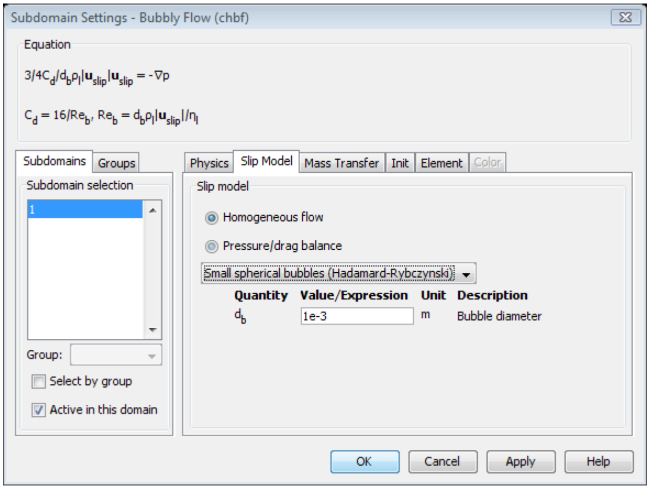
- If the pressure is not specified on any boundary, make sure that there is either a condition on the pressure in one point, or a condition on the integral of the pressure over the entire fluid domain.
- If you have a velocity boundary condition, increase it gradually from zero to its actual value during a short initial time interval using the built-in Heaviside smoothing function `flc1hs(t-deltat,deltat)`.

The Bubbly Flow Application Modes

The Bubbly Flow application modes are suitable for modeling the macroscopic behavior of many gas bubbles rising through a liquid. The application modes solve for the liquid velocity, the pressure, and the volume fraction of the gas phase.

SPECIFYING THE SLIP VELOCITY

In the **Subdomain Settings** dialog box, you need to specify how the slip velocity is to be calculated. The slip velocity is the relative velocity between the bubbles and the liquid.



Subdomain Settings - Bubbly Flow (chbf)

Equation

$3/4C_d/d_b\rho_l|\mathbf{u}_{slip}|\mathbf{u}_{slip} = -\nabla p$

$C_d = 16/Re_b,\ Re_b = d_b\rho_l|\mathbf{u}_{slip}|/\eta_l$

Subdomain selection

1

Group:

□ Select by group

☑ Active in this domain

Slip model

● Homogeneous flow

○ Pressure/drag balance

Small spherical bubbles (Hadamard-Rybczynski)

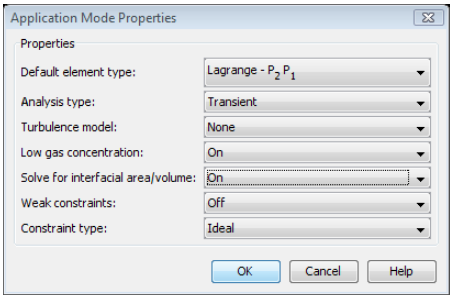
Quantity	Value/Expression	Unit	Description
d_b	1e-3	m	Bubble diameter

Figure 4-27: In the Subdomain Settings dialog box, you specify which slip model to use.

In general, choose **Pressure/drag balance**. For small bubbles (with a diameter less than 1mm), select **Small spherical bubbles (Hadamard-Rybczynski)** and for larger bubbles, select **Large bubbles**. Alternatively, you can specify a user-defined drag coefficient.

INCLUDING MASS TRANSFER

If desired, you can account for mass transfer between the gas and the liquid. For the dissolution of a gas, you can select the **Two-film theory model**. If the mass transfer rate depends on the interfacial area between the phases, you need to solve for an additional variable, the bubble number density. To do so, open the **Application Mode Properties** dialog box and select **On** from the **Solve for interfacial area/volume** list (see Figure 4-28). If you select the Two-film theory model, the application mode automatically switches on this property.



Default element type: Lagrange - $P_2 P_1$
Analysis type: Transient
Turbulence model: None
Low gas concentration: On
Solve for interfacial area/volume: On
Weak constraints: Off
Constraint type: Ideal

Figure 4-28: Set Solve for interfacial area/volume to On if the mass transfer rate depends on the interfacial area.

In the **Application Mode Properties** dialog box you can also turn on or off a turbulence model and change the equation form such that it corresponds to low or high gas concentration. The property **Low gas concentration** is set to **On** by default because this form is much easier to solve numerically and often gives satisfying results also for fairly high gas volume fractions.

TURBULENCE MODELING

You can choose between no turbulence model, the standard k-ε turbulence model and a k-ε turbulence model that includes bubble-induced turbulence. For the k-ε turbulence model with bubble-induced turbulence, you account for the extra production of turbulence due to the relative velocity between the bubbles and the liquid.

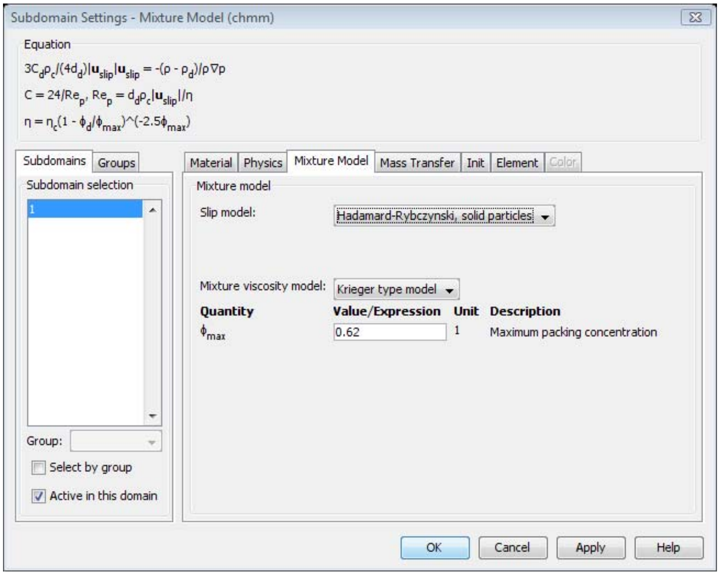
It can sometimes be difficult to solve a stationary multiphase flow model with a turbulence model, especially if the solver is provided with a poor initial guess. To find a suitable initial solution, you can make a transient, either laminar or turbulent, calculation. Use the result at the last time step as the initial condition to the stationary problem. You can also solve the transient problem until it reaches a steady-state solution.

The Mixture Model Application Modes

The Mixture Model application modes are suitable for macroscopic simulation of droplets or solid particles immersed in a liquid. The application modes solve for the mass averaged velocity, the pressure, and the volume fraction of the dispersed phase, which are the liquid droplets or the solid particles.

SLIP VELOCITY

In the **Subdomain Settings** dialog box you need to specify which model the application mode uses to determine the relative velocity between the two phases.



Subdomain Settings - Mixture Model (chmm)

Equation

$3C_d\rho_c/(4d_d)|\mathbf{u}_{slip}|\mathbf{u}_{slip} = -(\rho - \rho_d)/\rho\nabla p$

$C = 24/Re_p,\ Re_p = d_d\rho_c|\mathbf{u}_{slip}|/\eta$

$\eta = \eta_c(1 - \phi_d/\phi_{max})\hat{}(-2.5\phi_{max})$

Subdomain selection: 1

Group:

☐ Select by group

☑ Active in this domain

Mixture model

Slip model: Hadamard-Rybczynski, solid particles

Mixture viscosity model: Krieger type model

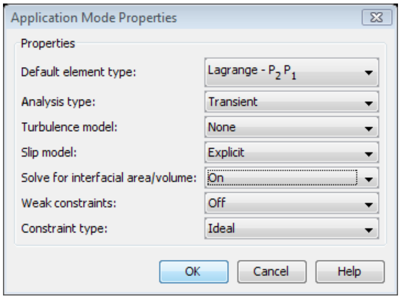
Quantity	Value/Expression	Unit	Description
ϕ_{max}	0.62	1	Maximum packing concentration

Figure 4-29: Specify which slip model to use in the Subdomain Settings dialog box.

For small particles or droplets, you can use the **Hadamard-Rybczynski** drag law. The **Schiller-Naumann** drag law is more general, but requires the solution of an additional variable. As a consequence, the application mode uses slightly more memory when you use the Schiller-Naumann model.

INCLUDING MASS TRANSFER

You can choose between **No mass transfer**, dissolution according to the **Two-film theory model**, or specifying a **User-defined mass transfer** expression for the mass transfer between the two phases. If you specify a user-defined expression that depends on the interfacial area/volume between the two phases, you need to solve for an additional variable: the number density of the dispersed phase. To do so, open the **Application Mode Properties** dialog box and select **On** from the **Solve for interfacial area/volume** list. If you select the Two-film theory model, the application mode automatically switches on this property.


Application Mode Properties

Properties

Default element type: Lagrange - P_2 P_1

Analysis type: Transient

Turbulence model: None

Slip model: Explicit

Solve for interfacial area/volume: On

Weak constraints: Off

Constraint type: Ideal

Figure 4-30: Switch on the property Solve for interfacial area/volume if the mass transfer rate depends on the interfacial area between the two phases.

TURBULENCE MODELING

For the Mixture Model you can choose between no turbulence model and the standard k-ε turbulence model. It can sometimes be difficult to solve a stationary multiphase flow model with a turbulence model, especially if the solver is provided with a poor initial guess. To find a suitable initial solution, you can make a transient, either laminar or turbulent, calculation. Use the result at the last time step as the initial condition to the stationary problem. You can also solve the transient problem until it reaches a steady-state solution.

Solving Large 3D Problems

Eventually you probably need to solve a model with a large number of degrees of freedom (DOFs). When direct solvers require too much memory or become relatively slow, consider using iterative solvers instead. This section contains a survey of the iterative solvers suitable for solving transport problems.

A problem can be considered to be large when iterative solvers are faster than any direct solver or when the direct solvers consume all of the computer's memory. For the PARDISO solver (which is the least memory consuming direct solver), the "break-even point" occurs at approximately 100,000 DOFs, depending on the capacity of your computer.

A typical application that has a tendency to produce large problems is three-dimensional fluid flow. Hence, when you open a new model using the Incompressible Navier-Stokes application mode, the default settings are designed to handle large problems. This section explains the logic behind these settings and provides guidelines for modifying the settings if necessary.

The Navier-Stokes equations form a nonlinear system of algebraic equations when they are discretized. This system of equations is solved using the Newton method, which solves a linearized system of equations in each Newton step until the solution satisfies the nonlinear system.

A direct solver performs Gauss elimination on the whole system to find the solution of the linear system. This elimination is memory consuming and suitable for problems that are considered small or medium sized. Iterative solvers try to find the solution to the linear system using several iterative steps, which is less memory consuming than the direct elimination. For this scheme to be efficient and robust, the linear system has to be reordered and rescaled, a procedure known as preconditioning.

For details about the linear system solvers, see "The Linear System Solvers" on page 392 of the *COMSOL Multiphysics User's Guide*.

Selecting a Linear System Solver

Regardless of if you are solving a stationary or a transient problem, you need to choose a linear system solver. The key to solving large transport problems is to make use of the geometric multigrid (GMG) algorithm. You can use it as a linear system solver, but, in most cases, it is preferable to use either GMRES of FGMRES as the linear system solver. The latter two are more stable than GMG, meaning that there are models where GMRES and FGMRES converge while GMG diverges.

GMRES is more memory efficient than FGMRES, using about half as much memory. The drawback is that GMRES can only use stationary methods as preconditioners while this limitation does not apply to FGMRES. The next section contains a discussion about the implication of this fact.

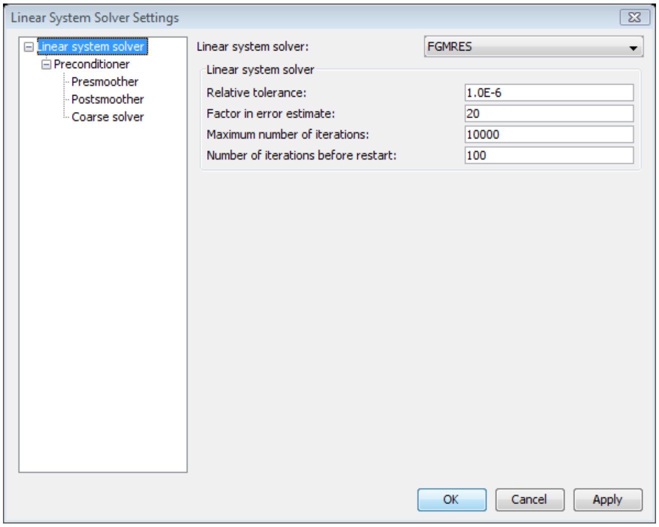
Figure 4-31 shows the default setting for the Incompressible Navier-Stokes application mode, which uses GMRES as the linear system solver. To open the dialog box in Figure 4-31, follow these steps:

1 In the **Model Navigator**, select **3D** from the **Space dimension** list.

2 From the **Application mode** tree, select **Chemical Engineering Module> Momentum Transport>Laminar Flow>Incompressible Navier-Stokes** and click **OK**.

3 Click the **Solver Parameters** button on the Main menu.

4 In the dialog box that appears, go to the **General** page and click the **Settings** button.



Linear System Solver Settings

Linear system solver: FGMRES

Linear system solver

Relative tolerance: 1.0E-6

Factor in error estimate: 20

Maximum number of iterations: 10000

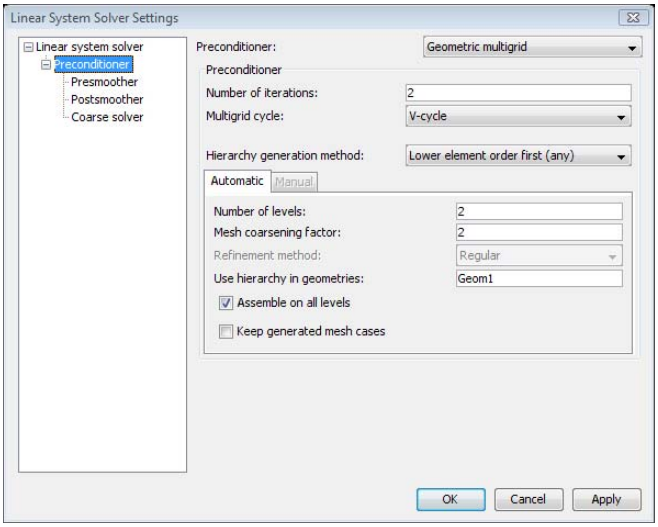
Number of iterations before restart: 100

Figure 4-31: Default Linear System Solver Settings for the 3D Incompressible Navier-Stokes application mode.

Preconditioners

Geometric multigrid (GMG) has its primary use as a preconditioner to the linear system solvers GMRES and FGMRES. However, the settings outlined below also apply when using GMG as a linear system solver.

The default settings for the Incompressible Navier-Stokes application mode in 3D is shown in Figure 4-32.



Linear System Solver Settings

Preconditioner: Geometric multigrid

Preconditioner

Number of iterations: 2

Multigrid cycle: V-cycle

Hierarchy generation method: Lower element order first (any)

Number of levels: 2

Mesh coarsening factor: 2

Refinement method: Regular

Use hierarchy in geometries: Geom1

☑ Assemble on all levels

□ Keep generated mesh cases

Figure 4-32: The Incompressible Navier-Stokes application mode in 3D uses geometric multigrid as the default preconditioner.

GENERATION OF MESH CASES

Carefully designed *mesh cases* are important in order for GMG to work efficiently. As can be seen in Figure 4-32, the default method for the Incompressible Navier-Stokes application mode in 3D is **Lower element order first (any)**. That method works using the following scheme:

1. Start from the current mesh.
2. If there is at least one shape function that can have a lower order, create a new mesh case with the same cells as the current mesh, but with the lower order shape functions. Leave all shape functions that cannot be lower at the same order. Go to Step 4.
3. Create a new mesh case that has the same shape functions as the current mesh but with fewer cells.

4 If the number of mesh cases is now equal to the number of requested levels then quit, else go to Step 1.

When using GMG, there are a number of things that you have to consider. The first thing is that the finest mesh case (Mesh Case 0) must have a sufficient resolution for your problem. Quadratic elements are the default elements for most application modes in the Chemical Engineering Module. The Incompressible Navier-Stokes application mode uses P2-P1 elements, which are quadratic Lagrange elements for the velocities and linear Lagrange elements for pressure.

If mesh case 0 has quadratic elements, then the automatic mesh case generation will generate a new, coarser mesh case (Mesh Case 1) with linear elements (Step 2 in the scheme above). The **Lower element order first (any)** method is not applicable, however, if you use special elements, for example, P1+P1 (mini) elements.

The default settings sets up two mesh cases. If you request more mesh cases, the **Lower element order first (any)** method will try to remesh the domain with fewer cells compared to Mesh Case 0 and 1 (Step 3). Mesh Case 2 uses the same element order as Mesh Case 1. For simple geometries, the automatic mesh case generation method is likely to give Mesh Case 2 good properties. Complex geometries must be meshed in several steps, however, and you then have to resort to manual construction of the mesh cases (see "Mesh Cases" on page 355 in the *COMSOL Multiphysics User's Guide*).

The GMG method solves an equation system on the coarsest mesh case, so make the coarsest mesh case with as few DOFs as possible. The mesh in the coarsest mesh case has to be dense enough to resolve some basic physical and numerical properties of the problem.

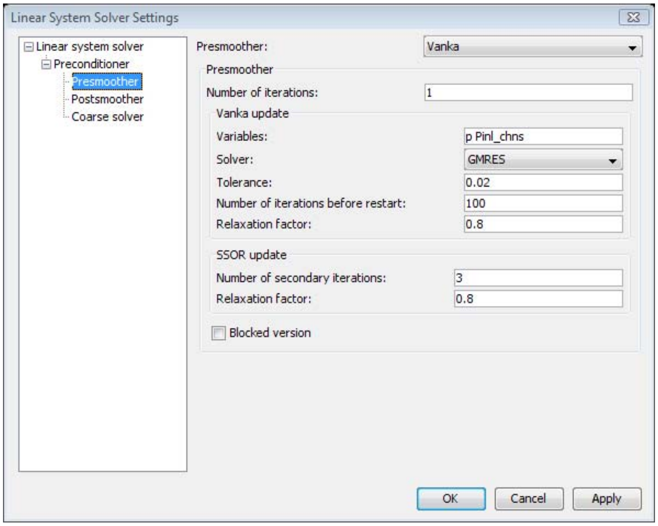
SMOOTHERS

When data is transferred from one multigrid level to another, it must be smoothed in order to prevent oscillations. This operation is done by presmoothers and postsmoothers.

The SSOR smoother/preconditioner is computationally inexpensive and efficient and is recommended for scalar transport equations both as pre- and postsmoother. GMG with SSOR as pre- and postsmoother is a stationary preconditioner and can hence be used as pre- and postsmoother for both GMRES and FGMRES.

SSOR is unstable for so-called saddle-point problems (the discretized Navier-Stokes equations, for example). In such cases, use the Vanka smoother/preconditioner instead. In the Incompressible Navier-Stokes application mode, treat the pressure as a Vanka variable. Figure 4-33 shows the default presmoother settings for the 3D

Incompressible Navier-Stokes application mode are displayed. The variable `Pinl_chns` is used by the laminar inflow/outflow boundary condition. Furthermore, if the model includes boundary conditions on weak form, any Lagrange multipliers also require the Vanka smoother.



Presmoother: Vanka

Presmoother

Number of iterations: 1

Vanka update

Variables: p Pinl_chns

Solver: GMRES

Tolerance: 0.02

Number of iterations before restart: 100

Relaxation factor: 0.8

SSOR update

Number of secondary iterations: 3

Relaxation factor: 0.8

☐ Blocked version

Figure 4-33: Default Presmoother settings for the 3D Incompressible Navier-Stokes application mode.

GMG with Vanka as the pre- and postsmoother is a stationary preconditioner if the Vanka system is solved by the direct solver, but it is a nonstationary preconditioner if the Vanka system is solved by GMRES. Because the Incompressible Navier-Stokes application mode uses GMRES as the linear system solver, the Vanka system is solved by a direct solver (see Figure 4-33).

COARSE SOLVER

The PARDISO direct solver can be used as a coarse solver. It is not necessary to solve the equation system for the coarse mesh very accurately. That is, you can set both the **Tolerance** and the **Factor in error estimate** properties to relatively high values: 0.1 and 1.0, respectively, for example.

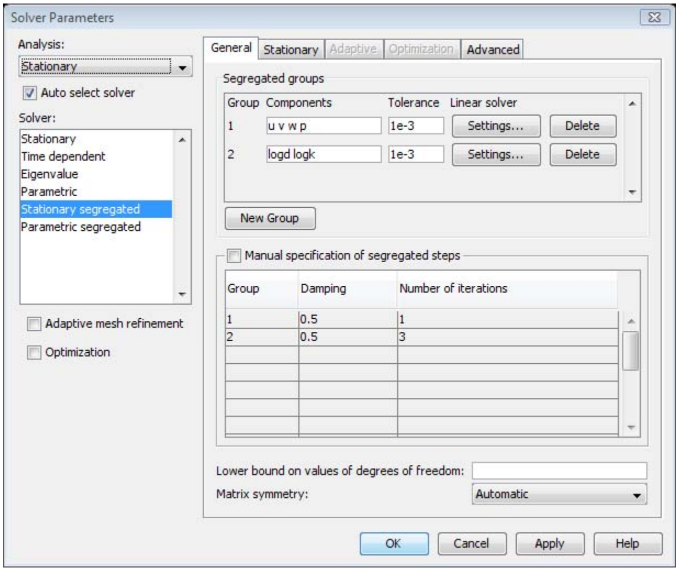
Using the Stationary Segregated Solver

You can use the stationary segregated solver for turbulence models as well as when your model contains several application modes (multiphysics models, for example).

Because this method solves for only a subset of the dependent variables at a time, it requires less memory than a fully coupled approach. Smaller equation systems also have the advantage that they are better conditioned than large equation systems. A segregated approach can therefore also be more stable than a fully coupled approach.

A group has to contain variables that, loosely speaking, "belong together." For example, if you have a model describing the mass transport of three species, group the velocities and the pressure in one group and the concentration variables in another group. The k-ε Turbulence Model application model uses the segregated solver by default with the settings as shown in Figure 4-34. The velocities `u`, `v`, and `w` and the pressure `p` constitute Group 1, and the turbulence variables `logk` and `logd` constitute Group 2.

Note: The velocity components and the pressure must always belong to the same group, because the pressure does not appear in the continuity equation and thus cannot be solved for using that equation only.



Solver Parameters

Analysis: Stationary

☑ Auto select solver

Solver: Stationary, Time dependent, Eigenvalue, Parametric, **Stationary segregated** (selected), Parametric segregated

□ Adaptive mesh refinement

□ Optimization

Segregated groups

Group	Components	Tolerance	Linear solver	
1	u v w p	1e-3	Settings...	Delete
2	logd logk	1e-3	Settings...	Delete

□ Manual specification of segregated steps

Group	Damping	Number of iterations
1	0.5	1
2	0.5	3

Lower bound on values of degrees of freedom:

Matrix symmetry: Automatic

Figure 4-34: The k-ε Turbulence Model application mode uses the Stationary Segregated solver as default solver.

If a group contains few DOFs, it is possible to use a direct solver. If the group contains a large number of DOFs, you can solve it with iterative solvers as described earlier in this section. Note though that if several groups are using a geometric multigrid (GMG) approach, they must all use the same mesh cases and the same multigrid cycle. However, the groups can use different pre- and postsmoothers as well as different coarse solvers.

5

Momentum Transport

This chapter describes the various types of momentum transport that you can simulate using the Chemical Engineering Module: laminar and turbulent flow, Newtonian and non-Newtonian flow, multiphase flow, and flow in porous media.

Introduction

All application modes described in this chapter are based on the Navier-Stokes equations, which in their most general form read

$$\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\mathbf{u}) = 0 \qquad (5\text{-}1)$$

$$\rho\frac{\partial\mathbf{u}}{\partial t} + (\rho\mathbf{u}\cdot\nabla)\mathbf{u} = -\nabla p + \nabla\cdot\tau + \mathbf{F} \qquad (5\text{-}2)$$

$$\rho C_p\left(\frac{\partial T}{\partial t} + (\mathbf{u}\cdot\nabla)T\right) = -(\nabla\cdot\mathbf{q}) + \tau{:}\mathbf{S} - \frac{T}{\rho}\frac{\partial\rho}{\partial T}\bigg|_p\left(\frac{\partial p}{\partial t} + (\mathbf{u}\cdot\nabla)p\right) + Q \qquad (5\text{-}3)$$

where

- ρ is the density ($\mathrm{kg/m^3}$)
- $\mathbf{u}$ is the velocity vector (m/s)
- p is pressure (Pa)
- τ is the viscous stress tensor (Pa)
- $\mathbf{F}$ is the body force vector ($\mathrm{N/m^3}$)
- C_p is the specific heat capacity at constant pressure (J/(kg·K))
- T is absolute temperature (K)
- $\mathbf{q}$ is the heat flux vector ($\mathrm{W/m^2}$)
- Q contains the heat sources ($\mathrm{W/m^3}$)

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

$$\mathbf{a}{:}\mathbf{b} = \sum_n\sum_m a_{nm}b_{nm} \qquad (5\text{-}4)$$

Equation 5-1 is the continuity equation and represents the conservation of mass. Equation 5-2 is a vector equation and represent the conservation of momentum. Equation 5-3 describes the conservation of energy, formulated in terms of temperature. This is an intuitive formulation that facilities specification of boundary conditions.

To close the equation system 5-1 through 5-3, some constitutive relations are needed. A common relation is derived by assuming that the fluid is Newtonian. Together with Stokes' assumption, the viscous stress tensor becomes:

$$\tau = 2\eta\mathbf{S} - \frac{2}{3}\eta(\nabla \cdot \mathbf{u})\mathbf{I} \qquad (5\text{-}5)$$

The dynamic viscosity η (Pa·s) is allowed to depend on the thermodynamic state, but not on the velocity field. All gases and many liquids can be considered Newtonian. Examples of non-Newtonian fluids are honey, mud, blood, liquid metals, and most polymer solutions. For modeling of flows of non-Newtonian fluids use the Non-Newtonian Flow application mode, see section Non-Newtonian Flow on page 132. All other application modes use stress tensors based on Equation 5-5. Other commonly used constitutive relations are Fourier's law of conduction and the ideal gas law.

There are several books where the Navier-Stokes equations are derived and concepts such as Newtonian fluids and the Stokes assumption are explained in detail. See, for example, the classical text by Batchelor (Ref. 1) and the more recent work by Panton (Ref. 2).

Many applications describe isothermal flows where Equation 5-3 is decoupled from Equation 5-1 and Equation 5-2. The temperature equation is described in the section "Convection and Conduction" on page 196, and the fully-coupled system is treated in the section "Non-Isothermal Flow and Flow with Variable Density" on page 48.

Each section discusses a specific form of the continuity and momentum equations. The properties of the equations are then used as a foundation to clarify their range of applicability.

References

1. G.K. Batchelor, "*An Introduction To Fluid Dynamics*," Cambridge University Press, 1967.

2. R.L. Panton, "*Incompressible Flow*," second edition, John Wiley & sons, inc., 1996.

The Incompressible Navier-Stokes Equations

The Incompressible Navier-Stokes application mode is based directly on the incompressible form of the Navier-Stokes equations. In its basic configuration, the application mode is useful for simulating flows of Newtonian fluids where the density can be assumed to be constant or nearly constant. It also acts as the common platform for most incompressible flow simulations. Functionality which is shared between the different flow application modes is described in this section.

In theory, the same equations describe laminar as well as turbulent flows. In practice however, the mesh resolution required to simulate turbulence with this application mode makes such an approach impractical. This is discussed further in the section "Numerical Stability—Artificial Diffusion" on page 121.

Buoyant flows in general should be treated by the Non-Isothermal Flow predefined Multiphysics coupling (see "Laminar Non-Isothermal Flow" on page 237). It is possible, however, to treat some simple cases of buoyant flows with the Incompressible Navier-Stokes application mode by invoking the Boussinesq approximation. "The Boussinesq Approximation" on page 126 describes this in more detail.

Subdomain Equations

The Incompressible Navier-Stokes application mode assumes that the fluid is incompressible; that is, that ρ is constant or nearly constant. This is the case for all fluids under normal conditions and also for gases at low velocities. For constant ρ, Equation 5-1 reduces to

$$\nabla \cdot \mathbf{u} = 0 \tag{5-6}$$

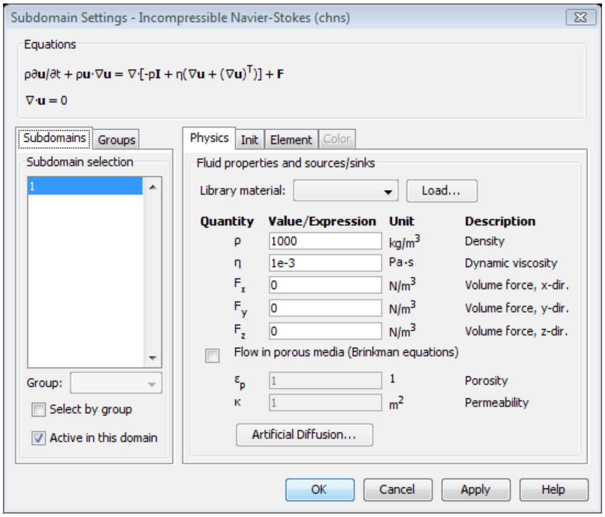
and the stress tensor τ in Equation 5-5 becomes

$$\tau = \eta(\nabla \mathbf{u} + (\nabla \mathbf{u})^{\mathrm{T}})$$

These assumptions put together give the following momentum equations:

$$\rho\frac{\partial \mathbf{u}}{\partial t} + (\rho \mathbf{u} \cdot \nabla)\mathbf{u} = -\nabla p + \nabla \cdot (\eta(\nabla \mathbf{u} + (\nabla \mathbf{u})^{\mathrm{T}})) + \mathbf{F} \tag{5-7}$$

The equations used by the application mode are displayed in the **Subdomain Settings** dialog box. Here, you also specify the physical properties of the fluid (see Figure 5-1).



Subdomain Settings - Incompressible Navier-Stokes (chns)

Equations

$$\rho\partial\mathbf{u}/\partial t + \rho\mathbf{u}\cdot\nabla\mathbf{u} = \nabla\cdot[-p\mathbf{I} + \eta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)] + \mathbf{F}$$

$$\nabla\cdot\mathbf{u} = 0$$

Subdomain selection: 1

Group:

☐ Select by group

☑ Active in this domain

Fluid properties and sources/sinks

Library material:

Quantity	Value/Expression	Unit	Description
ρ	1000	kg/m^3	Density
η	1e-3	Pa·s	Dynamic viscosity
F_x	0	N/m^3	Volume force, x-dir.
F_y	0	N/m^3	Volume force, y-dir.
F_z	0	N/m^3	Volume force, z-dir.

☐ Flow in porous media (Brinkman equations)

ε_p	1	1	Porosity
κ	1	m^2	Permeability

Figure 5-1: The Subdomain Settings dialog box for 3D incompressible Navier-Stokes equations.

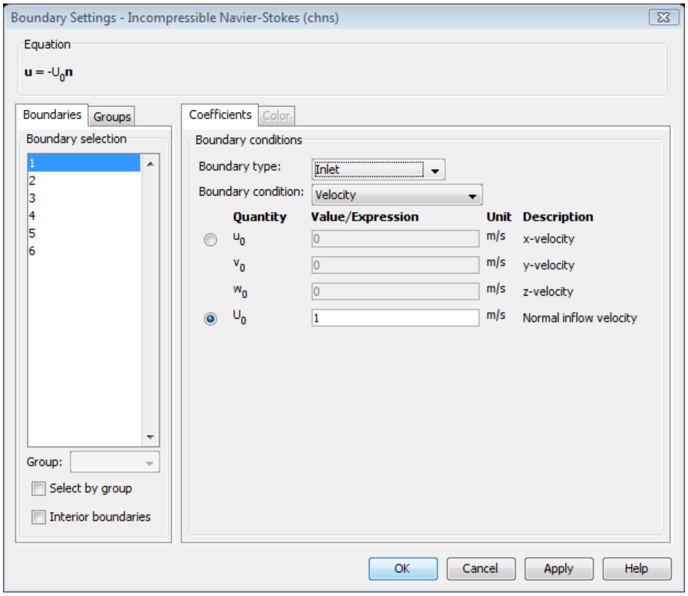
Boundary Conditions

The boundary conditions for the Incompressible Navier-Stokes application mode are grouped into the following types:

- Wall
  - No slip (Default)
  - Slip
  - Sliding wall
  - Moving/leaking wall
- Inlet
  - Velocity (Default)
  - Pressure, no viscous stress
  - Laminar inflow

- Outlet
  - Velocity
  - Pressure
  - Pressure, no viscous stress (Default)
  - No viscous stress
  - Normal stress
  - Laminar outflow
- Symmetry boundary
  - Symmetry (Default)
  - Axial symmetry
- Open boundary
  - Normal stress (Default)
  - No viscous stress
- Stress
  - General stress (Default)
  - Normal stress
  - Normal stress, normal flow

You specify a boundary condition in the **Boundary Settings** dialog where you first select the appropriate **Boundary type** and then a **Boundary Condition**.



Boundary Settings - Incompressible Navier-Stokes (chns)

Equation

$\mathbf{u} = -U_0\mathbf{n}$

Boundaries | Groups

Boundary selection

1
2
3
4
5
6

Group:

□ Select by group

□ Interior boundaries

Coefficients | Color

Boundary conditions

Boundary type: Inlet

Boundary condition: Velocity

	Quantity	Value/Expression	Unit	Description
○	u_0	0	m/s	x-velocity
	v_0	0	m/s	y-velocity
	w_0	0	m/s	z-velocity
●	U_0	1	m/s	Normal inflow velocity

Figure 5-2: Boundary Settings dialog box for the Incompressible Navier-Stokes application mode.

If a mathematical formulation describes more than one type of physical boundary condition, it can appear in more than one boundary type. However, every possible use of a single mathematical formulation cannot be covered. Hence, the boundary types should be regarded as guidelines, not as restrictions on the applicability of the formulations.

The theory of most boundary conditions can be found in Ref. 1.

WALL

These boundary conditions describe the existence of a solid wall.

No Slip

This is the standard and default boundary condition for a stationary solid wall. The condition prescribes

$$\mathbf{u} = \mathbf{0}$$

that is, that the fluid at the wall is not moving.

Moving/Perforated Wall

If the wall moves, so must the fluid. Hence, this boundary condition prescribes

$$\mathbf{u} = \mathbf{u}_{\mathrm{w}}$$

Note that setting this boundary condition does not automatically cause the associated wall to move. The section "The Moving Mesh Application Mode" on page 401 of the *COMSOL Multiphysics Modeling Guide* describes how to set up a model with moving boundaries.

You can also use the Moving/perforated wall boundary condition to simulate a wall where fluid is leaking into or leaving through a perforated wall.

Sliding Wall

If you use this boundary condition, the wall is assumed to behave like a conveyor belt, that is, that the surface is sliding in its tangential direction. The wall does not have to actually move in the coordinate system.

In two space dimensions (2D), the tangential direction is unambiguously defined by the direction of the boundary. However, the situation becomes more complicated in 3D. For this reason, this boundary condition has slightly different definitions in the different space dimensions.

2D and Axial Symmetry The velocity is given as a scalar U_{w} and the condition prescribes

$$\mathbf{u} \cdot \mathbf{n} = 0, \quad \mathbf{u} \cdot \mathbf{t} = U_{\mathrm{w}}$$

where $\mathbf{t} = (-n_y, n_x)$ for 2D and $\mathbf{t} = (-n_z, n_r)$ for axial symmetry. When the Swirl flow property is activated (see section "Swirl Flow" on page 129), the swirl wall component is specified separately.

3D The velocity is set equal to a given vector $\mathbf{u}_{\mathrm{w}}$ projected onto the boundary plane:

$$\mathbf{u} = \mathbf{u}_{\mathrm{w}} - (\mathbf{n} \cdot \mathbf{u}_{\mathrm{w}})\mathbf{n}$$

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 may be 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:

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

where $\mathbf{t}$ is a tangential vector to the boundary.

INLET

This boundary type contains different ways to specify conditions on a boundary where the fluid is supposed to enter the domain. Notice that the formulations contained in this boundary type all appear, some of them slightly modified, in the Outflow boundary type as well. Hence, there is nothing in the mathematical formulations that prevents a fluid from leaving the domain through boundaries where you have specified the Inlet boundary type.

Velocity

This boundary condition offers two ways to specify an inlet velocity. The first is to set the velocity equal to a given vector $\mathbf{u}_0$:

$$\mathbf{u} = \mathbf{u}_0$$

The other is to specify a normal inflow velocity:

$$\mathbf{u} = -\mathbf{n}\mathrm{U}_0$$

Note that the boundary normal, $\mathbf{n}$, is pointing out of the domain.

Pressure, No Viscous Stress

This boundary condition specifies vanishing viscous stress along with a Dirichlet condition on the pressure:

$$\eta(\nabla\mathbf{u}+(\nabla\mathbf{u})^{T})\mathbf{n} = \mathbf{0}, \qquad p = p_0$$

It is a numerically stable boundary condition that admits total control of the pressure level along the entire boundary. However, if the inflow is not normal to the boundary, this condition yields an overspecification of the problem. In the case that your solution turns out to have a non-normal inflow velocity, there are two choices. Either, move the boundary farther away to a location where the inflow is normal to the boundary or, use a stress type boundary condition described on page 120.

Note that this condition is identical to the Pressure, no viscous stress condition for Outflow boundaries. Hence, depending on the pressure field in the rest of the subdomain, a boundary with this condition can very well become an outflow boundary.

Laminar Inflow

This boundary conditions adds a weak contribution of the form

$$-L_{\mathrm{entr}}((-p\mathbf{I} + \eta(\nabla_{\mathrm{t}}\mathbf{u} + (\nabla_{\mathrm{t}}\mathbf{u})^T)),\nabla_{\mathrm{t}}\hat{\mathbf{u}})_{\Gamma} - p_{\mathrm{entr}}(\mathbf{n},\hat{\mathbf{u}})_{\Gamma} - L_{\mathrm{entr}}((\nabla_{\mathrm{t}}\cdot\mathbf{u}),\hat{p})_{\Gamma} \quad (5\text{-}8)$$

Where $(.,.)_{\Gamma}$ is the L_2 inner product on the boundary, ∇_{t} is the gradient in the tangential direction of the boundary, and $\hat{\mathbf{u}}$ and $\hat{p}$ are the test functions for the velocities and pressure, respectively. The addition of Equation 5-8 corresponds to the situation shown in Figure 5-3: a fictitious domain of length L_{entr} is assumed to be attached to the inlet of the computational domain. The flow in this fictitious domain is assumed to be laminar plug flow. If you select the option **Constrain outer edges to zero**, the flow in the fictitious domain is instead assumed to be fully developed laminar channel flow (in 2D) or fully developed laminar internal flow (in 3D). This does not affect the boundary condition in the real domain, Ω, where the boundary conditions are always fulfilled.

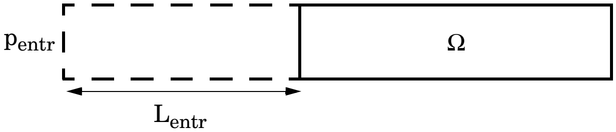


Figure 5-3: Sketch of the physical situation simulated when using the Laminar inflow boundary condition. Ω is the actual computational domain while the dashed domain is a fictitious domain.

If you specify an average inlet velocity or inlet mass flow instead of the pressure, COMSOL Multiphysics adds an ODE that calculates a pressure, p_{entr}, such that the desired inlet velocity or mass flow is obtained.

OUTFLOW

This boundary type contains different ways to specify conditions on a boundary where the fluid exits the domain. Note that all of the formulations in this type can be found, possibly slightly modified, in other boundary types as well. Hence, there is nothing in the mathematical formulations that prevent a fluid from entering the domain through boundaries where you have set the Outflow boundary type.

Setting outlet conditions for the Navier-Stokes equations is not a trivial task. Some practical guidelines can be found in the section "Selecting Boundary Conditions" on page 44. A general rule of thumb, however, is that if there is something interesting happening at an outflow boundary, extend the computational domain to include this phenomenon.

Velocity

This boundary condition offers two ways to specify an outlet velocity. The first is to set the velocity equal to a given vector $\mathbf{u}_0$:

$$\mathbf{u} = \mathbf{u}_0$$

The other is to specify a normal outlet velocity:

$$\mathbf{u} = \mathbf{n}\mathrm{U}_0$$

Observe that the boundary normal, $\mathbf{n}$, is pointing out of the domain.

Pressure, No Viscous Stress

This boundary condition specifies vanishing viscous stress along with a Dirichlet condition on the pressure:

$$\eta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)\mathbf{n} = \mathbf{0}, \qquad p = p_0$$

It is a numerically stable boundary condition that admits total control of the pressure level at the whole boundary. However, if the outflow is not normal to the boundary, this condition is an overspecification. In the case that your solution turns out to have a non-normal outflow velocity, there are two choices. Either move the boundary farther away to a location where the outflow is normal to the boundary or use a stress type boundary condition described on page 120.

Observe that this condition is identical to the Pressure, no viscous stress condition for Inflow boundaries. Hence, depending on the pressure field in the rest of the subdomain, a boundary with this condition can very well become an inflow boundary.

Pressure

This boundary condition prescribes only a Dirichlet condition for the pressure:

$$p = p_0$$

Use this boundary condition only for high Reynolds number outflow boundaries, that is $\mathrm{Re}^c = \rho|\mathbf{u}|h/(2\eta) >> 1$. It is far less stable than the Pressure, no viscous stress boundary condition, but it is consistent with a non-normal outflow velocity.

No Viscous Stress

Prescribes vanishing viscous stress:

$$\eta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)\mathbf{n} = \mathbf{0}$$

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. It should however be combined with a point constraint on the pressure to be numerically stable (see "Point Settings" on page 121).

Normal Stress

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

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

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

$$p = 2\eta\frac{\partial u_n}{\partial n} + f_0 \tag{5-9}$$

If $\partial u_n/\partial n$ is small, Equation 5-9 can be interpreted as $p \approx f_0$.

Laminar Outflow

This boundary conditions adds a weak contribution of the form

$$-L_{exit}((-p\mathbf{I} + \eta(\nabla_t\mathbf{u} + (\nabla_t\mathbf{u})^T)), \nabla_t\hat{\mathbf{u}})_\Gamma - p_{exit}(\mathbf{n}, \hat{\mathbf{u}})_\Gamma - L_{exit}((\nabla_t \cdot \mathbf{u}), \hat{p})_\Gamma \tag{5-10}$$

Where $(.,.)_\Gamma$ is the L_2 inner product on the boundary, ∇_t is the gradient in the tangential direction of the boundary, and $\hat{\mathbf{u}}$ and $\hat{p}$ are the test functions for the velocities and pressure, respectively. The addition of Equation 5-10 corresponds to the situation shown in Figure 5-4: a fictitious domain of length L_{exit} is assumed to be attached to the outlet of the computational domain. The flow in this fictitious domain is assumed to be laminar plug flow. If you select the **Constrain outer edges to zero** check box, the flow in the fictitious domain is instead assumed to be fully developed laminar channel flow (in 2D) or fully developed laminar internal flow (in 3D). This does not affect the boundary condition in the real domain, Ω, where the boundary conditions are always fulfilled.

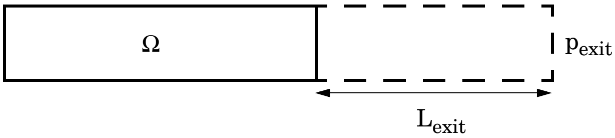


Figure 5-4: Sketch of the physical situation simulated when using Laminar outflow boundary condition. Ω is the actual computational domain while the dashed domain is a fictitious domain.

If you specify an average outlet velocity or inlet mass flow instead of the pressure, the software adds an ODE that calculates p_{exit} such that the desired outlet velocity or mass flow is obtained.

SYMMETRY BOUNDARY

Prescribes no penetration and vanishing shear stresses:

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

In 2D Axial Symmetry, the above formulation is called **Symmetry**.

Axial Symmetry

This boundary condition is only available in 2D Axial Symmetry. Use it on all boundaries with coordinate $r = 0$. It prescribes $u_r = 0$ and vanishing stresses in the z direction.

OPEN BOUNDARY

You can use this boundary type on boundaries that are open to large volumes of fluid. Fluid can both enter and leave the domain on boundaries with this type of condition.

No Viscous Stress

Prescribes vanishing viscous stress:

$$\eta(\nabla\mathbf{u}+(\nabla\mathbf{u})^T)\mathbf{n} = \mathbf{0}$$

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. It should however be combined with a point constraint on the pressure to be numerically stable (see “Point Settings” on page 121).

Normal Stress

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

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

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

$$p = 2\eta\frac{\partial u_n}{\partial n} + f_0 \tag{5-11}$$

If $\partial u_n / \partial n$ is small, Equation 5-11 can be interpreted as $p \approx f_0$.

STRESS

This type of boundary condition represents a very general class of conditions also known as traction boundary conditions.

General Stress

The total stress on the boundary is set equal to a given stress **F**:

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

This boundary condition implicitly sets a constraint on the pressure that for 2D flows can be written

$$p = 2\eta\frac{\partial u_n}{\partial n} - \mathbf{n}\cdot\mathbf{F} \tag{5-12}$$

If $\partial u_n / \partial n$ is small, Equation 5-12 can be interpreted as $p \approx -\mathbf{n}\cdot\mathbf{F}$.

Normal Stress

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

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

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

$$p = 2\eta\frac{\partial u_n}{\partial n} + f_0 \tag{5-13}$$

If $\partial u_n / \partial n$ is small, Equation 5-13 can be interpreted as $p \approx f_0$.

Normal Stress, Normal Flow

In addition to the stress condition set in the Normal stress condition, this condition also prescribes that there must be no tangential velocities on the boundary:

$$(-p\mathbf{I} + \eta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T))\mathbf{n} = -\mathrm{f}_0\mathbf{n}, \quad \mathbf{t}\cdot\mathbf{u} = \mathbf{0}$$

Also this boundary condition implicitly sets a constraint on the pressure that for 2D flows can be written

$$p = 2\eta\frac{\partial u_n}{\partial n} + \mathrm{f}_0 \qquad (5\text{-}14)$$

If $\partial u_n / \partial n$ is small, Equation 5-14 can be interpreted as $p \approx \mathrm{f}_0$.

Point Settings

If it is not possible to specify the pressure level using a boundary condition, the pressure must be set in some other way, for example, by specifying a fixed pressure at a point. You find a dialog box for **Point Settings** on the **Physics** menu.

Numerical Stability—Artificial Diffusion

The momentum equations (Equation 5-7) are of convection-diffusion type. It is well known that if standard Galerkin discretization is used, such equations become unstable for an element Peclet number (Pe) larger than one (Ref. 2):

$$\mathrm{Pe} = \frac{\rho\|\mathbf{u}\|h}{2\eta} > 1 \qquad (5\text{-}15)$$

where h is the element size. The instabilities might cause the simulation to diverge. If a solution where Pe is larger than one is obtained, it can often have spurious oscillations even if the exact solution is smooth. Figure 5-5 shows an example of such ocscillations.

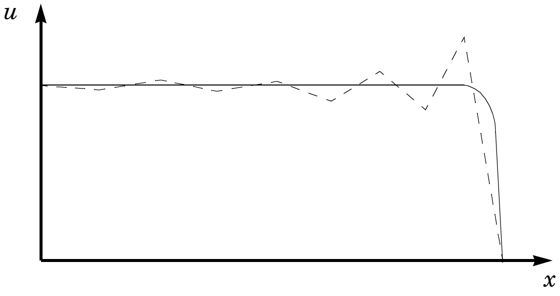
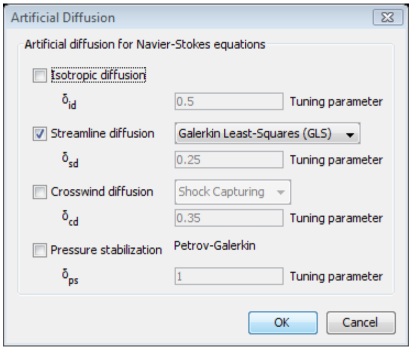


Figure 5-5: A 1D convection diffusion example with $\mathrm{Pe} > 1$. *The exact solution (solid line) is smooth while the solution obtained by standard Galerkin discretization (dashed line) contains spurious oscillations.*

The element Peclet number can always be made less than one by refining the mesh, that is, by making the mesh element size h small enough. This method is not always feasible because it can require a very dense mesh. Instead, so-called stabilization methods, or artificial diffusion, are used. COMSOL Multiphysics provides several of these methods. The dialog box for selecting artificial diffusion (Figure 5-6) is opened from the **Subdomain Settings** dialog box by clicking the **Artificial Diffusion** button (see Figure 5-1 on page 111).



Artificial Diffusion

Artificial diffusion for Navier-Stokes equations

□ Isotropic diffusion

δ_{id} 0.5 Tuning parameter

☑ Streamline diffusion Galerkin Least-Squares (GLS)

δ_{sd} 0.25 Tuning parameter

□ Crosswind diffusion Shock Capturing

δ_{cd} 0.35 Tuning parameter

□ Pressure stabilization Petrov-Galerkin

δ_{ps} 1 Tuning parameter

Figure 5-6: The Artificial Diffusion dialog box showing the default setting: Galerkin least-squares (GLS) streamline diffusion.

None of the artificial diffusion methods allows arbitrarily high Peclet numbers. As the Peclet number goes to infinity, they either lose their stabilizing effect or completely destroy the solution.

To use no artificial diffusion at all corresponds to the standard Galerkin discretization. A requirement for stability is then that higher-order basis functions must be used for the velocities than for the pressure. This condition is know as the Babuska-Brezzi condition or the inf-sup stability condition (Ref. 1 and 3). Two of the artificial diffusion methods provided in COMSOL Multiphysics can relax the Babuska-Brezzi condition: Galerkin least-squares (GLS) and pressure stabilization.

The mathematical description of the methods is given in Chapter 15, "Stabilization Techniques," in the *COMSOL Multiphysics Modeling Guide*.

ISOTROPIC DIFFUSION

Isotropic diffusion increases the viscosity in regions where the Peclet number is large. Hence, the original equation is perturbed. This method can force the simulation to be stable, but there is no guarantee that the solution is the correct physical solution.

STREAMLINE DIFFUSION

In most cases, spurious oscillations arise in the streamline direction only. Streamline diffusion methods introduce diffusion in the streamline direction by weighting the discretization scheme in the upstream direction. This means that more information is taken from the direction from which information is convected and less from the direction in which the information is traveling. Streamline-diffusion methods are closely related to the upwind methods in finite difference and finite volume methods.

Anisotropic Streamline Diffusion

This method projects the extra diffusion onto the streamlines and is therefore much less brutal than the isotropic diffusion. Still, the method is not consistent, meaning that the exact solution does not solve the discrete system. Therefore, the accuracy of the solution cannot be guaranteed.

Streamline Upwind Petrov-Galerkin (SUPG)

The Streamline Upwind Petrov-Galerkin method is a consistent method, which means that it does not perturb the equations. A model that converges with this method can be considered to be a solution to the discrete counterpart of Equation 5-6 and Equation 5-7. This method is somewhat less stabilizing than the anisotropic streamline diffusion because the terms that differ between the methods are mainly destabilizing.

Streamline Upwind Petrov-Galerkin (SUPG), Compensated

This version of SUPG turns off the diffusion in regions with element Peclet number less than one.

Galerkin Least-Squares (GLS)

The Galerkin least-squares (GLS) method is a consistent method that circumvents the Babuska-Brezzi condition, that is, it is possible to use equal-order elements for the velocities and the pressure.

For the momentum equations, GLS has many of its stability properties in common with SUPG. The superior stability of GLS can often be explained by the stabilizing terms on the continuum equation, terms that SUPG lacks.

GLS is activated per default even though it is more expensive to assemble than SUPG. The extra cost is often more than compensated for by the superior stability properties of GLS.

CROSSWIND DIFFUSION

In some cases, diffusion in the streamline direction is not enough to enforce stability. This is the situation in thin boundary layers and shear layers where there are large gradients in directions orthogonal to the streamline direction.

Ordo $h^{3/2}$

The Ordo $h^{3/2}$ method is an efficient and inexpensive method with good convergence properties. It is not consistent but has good accuracy compared to methods of similar complexity. However, the method is constructed exclusively for linear elements and it is dimensionally inconsistent. The latter is important to note if the model uses other units than SI units.

Shock Capturing

This method is more general and often better than the Ordo $h^{3/2}$ method. It is also consistent and independent of which units that are used. It is, however, considerably more computationally expensive than the Ordo $h^{3/2}$ method.

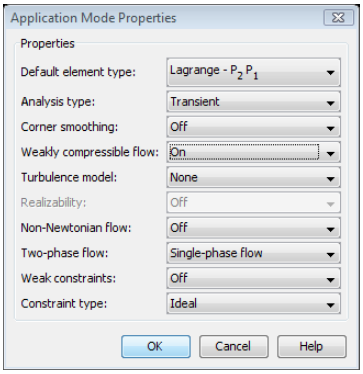
PRESSURE STABILIZATION

The main feature of this method is that it circumvents the Babuska-Brezzi condition. It can also add stability essential to iterative solvers.

Pressure stabilization together with SUPG can be regarded as a less expensive and less stable version of GLS. Pressure stabilization should not be used together with GLS.

Corner Smoothing

You find the **Corner smoothing** property in the **Application Mode Properties** dialog box. It can be a useful property when the model contains walls with slip conditions, as described below.



Default element type: Lagrange - $P_2 P_1$
Analysis type: Transient
Corner smoothing: Off
Weakly compressible flow: On
Turbulence model: None
Realizability: Off
Non-Newtonian flow: Off
Two-phase flow: Single-phase flow
Weak constraints: Off
Constraint type: Ideal

Figure 5-7: The Application Mode Properties dialog box where Corner smoothing can be turned on and off.

Consider the situation sketched in Figure 5-8. At the point where the boundaries Γ_1 and Γ_2 intersect, there will be two boundary normals, one for Γ_1 and one for Γ_2. These two normals are denoted $\mathbf{n}_{\Gamma_1}$ and $\mathbf{n}_{\Gamma_2}$ in Figure 5-8. If the boundaries now both have no-penetration condition, there will be two Dirichlet conditions at the point of intersection, namely

$$\mathbf{n}_{\Gamma_1} \cdot \mathbf{u} = 0 \qquad (5\text{-}16)$$

and

$$\mathbf{n}_{\Gamma_2} \cdot \mathbf{u} = 0 \qquad (5\text{-}17)$$

The only way that both Equation 5-16 and Equation 5-17 can be fulfilled is if $\mathbf{u} \equiv \mathbf{0}$ at the point of intersection. This is not always the expected solution, however.

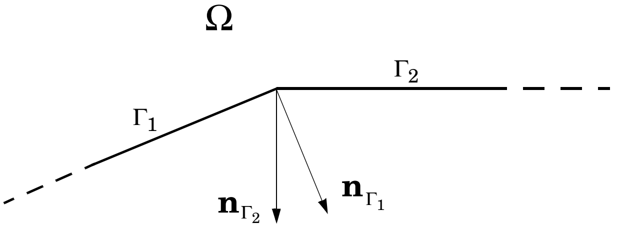


Figure 5-8: Intersection between the boundaries Γ_1 *and* Γ_2. $\mathbf{n}_{\Gamma_1}$ *and* $\mathbf{n}_{\Gamma_2}$ *are the boundary normals prescribed by* Γ_1 *and* Γ_2 *respectively.* Ω *is the computational domain.*

When corner smoothing is activated, any Dirichlet condition $d(\mathbf{n}) = 0$ is replaced by $d(\mathbf{n}_\mathrm{w}) = 0$, where $\mathbf{n}_\mathrm{w}$ is a vector of dependent boundary variables whose solution in each point is the average of all normals in that point. In the current example, equations 5-16 and 5-17 are replaced with

$$\mathbf{n}_\mathrm{w} \cdot \mathbf{u} = \mathbf{0} \qquad (5\text{-}18)$$

and $\mathbf{n}_\mathrm{w}$ has the solution $\mathbf{n}_\mathrm{w} = 1/2(\mathbf{n}_{\Gamma_1} + \mathbf{n}_{\Gamma_2})$. Equation 5-18 can then be satisfied for $\mathbf{u} \neq \mathbf{0}$.

The Boussinesq Approximation

The Boussinesq approximation is a way to treat some simple cases of buoyant flows without having to use a compressible formulation of the Navier-Stokes equations.

The Boussinesq approximation assumes that variations in density have no effect on the flow field except that they give rise to buoyant forces. The density is taken to be a reference value, ρ_0, except in the body force term, which is set to

$$\mathbf{F} = (\rho_0 + \Delta\rho)\mathbf{g} \qquad (5\text{-}19)$$

where $\mathbf{g}$ is the gravity vector. You can enter an expression for Equation 5-19 in the **Volume force** edit fields in the **Subdomain Settings** dialog box (see Figure 5-1 on page 111). However, further simplifications are often possible. Because $\mathbf{g}$ can be written in terms of a potential, Φ, it is possible to write Equation 5-19 as:

$$\mathbf{F} = -\nabla(\rho_0 \Phi) + \Delta\rho\mathbf{g}$$

The first part can be canceled out by splitting the true pressure, p, as a sum of a hydrodynamic component, P, and a hydrostatic component, $-\rho_0\Phi$. Then you can write equations 5-6 and 5-7 in terms of the hydrodynamic pressure $P = p + \rho_0\Phi$:

$$\nabla \cdot \mathbf{u} = 0 \tag{5-20}$$

$$\rho_0 \frac{\partial \mathbf{u}}{\partial t} + (\rho_0 \mathbf{u} \cdot \nabla)\mathbf{u} = -\nabla P + \nabla \cdot (\eta(\nabla \mathbf{u} + (\nabla \mathbf{u})^{\mathrm{T}})) + \mathbf{g}\Delta\rho \tag{5-21}$$

Hence, to obtain the Boussinesq approximation on this form, you only have to enter the expression for $\mathbf{g}\Delta\rho$ in the **Volume Force** edit fields in the **Subdomain Settings** dialog box.

In practice, the shift from p to P can be ignored except where the pressure appears in boundary conditions. The pressure that you specify at boundaries is the hydrodynamic pressure in this case. For example, on a vertical outflow or inflow boundary, the hydrodynamic pressure is typically a constant, while the true pressure is a function of the vertical coordinate.

The system that Equation 5-20 and Equation 5-21 form has its limitations. The main assumption is that the density fluctuations must be small, that is, $\Delta\rho/\rho_0 \ll 1$. There are also some more subtle constraints that, for example, makes the Boussinesq approximation unsuitable for systems of very large dimensions. An excellent discussion of the Boussinesq approximation and its limitations appears in Chapter 14 of Ref. 4.

Example Models

There are several models in the *Chemical Engineering Module Model Library* that make use of the Incompressible Navier-Stokes application mode. The model “Flow Between Two Parallel Plates” on page 39 demonstrates a 2D formulation, while “Pressure Recovery in a Diverging Duct” on page 10 demonstrates the 2D axisymmetric formulation. An example in 3D is given in the model “Flow in a Fuel Cell Stack” on page 18.

References

1. P.M. Gresho and R.L. Sani, *Incompressible Flow and the Finite Element Method, Volume 2: Isothermal Laminar Flow*, John Wiley and Sons, LTD, 2000.

2. O.C. Zienkiewicz, R.L. Taylor, and P. Nithiarasu, *The Finite Element Method for Fluid Dynamics*, Sixth edition, Elsevier, 2005.

3. C. Johnson, *Numerical solution of partial differential equations by the finite element method*, Studentlitteratur, 1987.

4. D.J. Tritton, *Physical Fluid Dynamics*, second edition, Oxford University Press, 1988.

Swirl Flow

The Swirl Flow application mode is an extension of the Incompressible Navier-Stokes application mode for axially symmetric geometries. The basic Navier-Stokes application mode assumes that the azimuthal velocity u_φ in a 2D axisymmetric model is zero, while the Swirl Flow application mode only assumes that there are no variations in any model variable in the azimuthal direction.

For a cylindrical coordinate system, under the assumption that $\partial/\partial\varphi = 0$, Equation 5-6 and Equation 5-7 can be written

$$\frac{\partial u_r}{\partial r} + \frac{u_r}{r} + \frac{\partial u_z}{\partial z} = 0 \tag{5-22}$$

$$\begin{aligned}
\rho\left(\frac{\partial u_r}{\partial t} + u_r\frac{\partial u_r}{\partial r} + u_z\frac{\partial u_r}{\partial z} - \frac{u_\varphi^2}{r}\right) &= -\frac{\partial p}{\partial r} + \frac{1}{r}\frac{\partial}{\partial r}(r\tau_{rr}) + \frac{\partial \tau_{zr}}{\partial z} - \frac{\tau_{\varphi\varphi}}{r} + \boldsymbol{F}_r \\
\rho\left(\frac{\partial u_z}{\partial t} + u_r\frac{\partial u_z}{\partial r} + u_z\frac{\partial u_z}{\partial z}\right) &= -\frac{\partial p}{\partial z} + \frac{1}{r}\frac{\partial}{\partial r}(r\tau_{rz}) + \frac{\partial \tau_{zz}}{\partial z} + \boldsymbol{F}_z \\
\rho\left(\frac{\partial u_\varphi}{\partial t} + u_r\frac{\partial u_\varphi}{\partial r} + \frac{u_r u_\varphi}{r} + u_r\frac{\partial u_\varphi}{\partial z}\right) &= \frac{1}{r^2}\frac{\partial}{\partial r}(r^2\tau_{r\varphi}) + \frac{\partial \tau_{z\varphi}}{\partial z} + \boldsymbol{F}_\varphi
\end{aligned} \tag{5-23}$$

The Newtonian viscous stress tensor (Equation 5-5) in cylindrical coordinates assuming $\partial/\partial\varphi = 0$ is given by

$$\begin{aligned}
\tau_{rr} &= 2\eta\frac{\partial u_r}{\partial r}, & \tau_{zr} &= \tau_{rz} = \eta\left(\frac{\partial u_z}{\partial r} + \frac{\partial u_r}{\partial z}\right) \\
\tau_{zz} &= 2\eta\frac{\partial u_z}{\partial z}, & \tau_{r\varphi} &= \tau_{\varphi r} = \eta r\frac{\partial}{\partial r}\left(\frac{u_\varphi}{r}\right) \\
\tau_{\varphi\varphi} &= 2\eta\frac{u_r}{r}, & \tau_{z\varphi} &= \tau_{\varphi z} = \eta\frac{\partial u_\varphi}{\partial z}
\end{aligned}$$

When the azimuthal velocity component u_φ is included in the equations, the application mode by default denotes it w. The remaining components are then $u = u_r$ and $v = u_z$ just as when the swirl velocity is not present.

In an axisymmetric geometry, COMSOL Multiphysics assumes the symmetry axis to be at $r = 0$. This means that you have to build all models using the axisymmetry modes to the right of the axis $r = 0$. A correctly drawn example is shown in Figure 5-9.

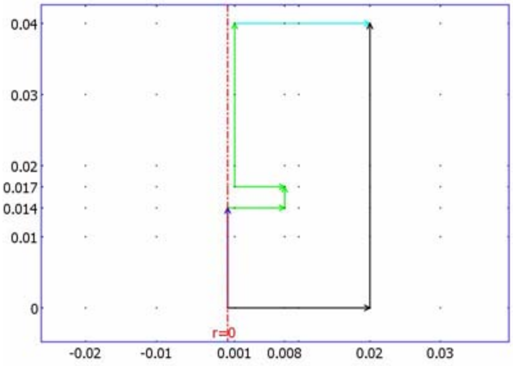


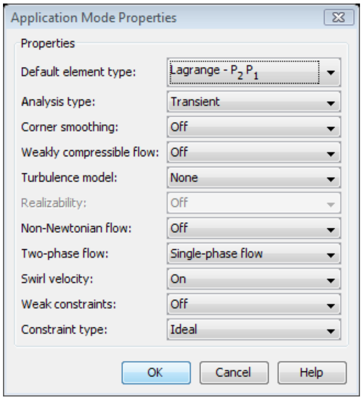
Figure 5-9: A geometry drawn in the 2D Axial Symmetry space dimension. Notice that the whole domain must be contained to the right of the symmetry line $r = 0$.

The Swirl Flow application mode shares boundary conditions, point settings, and artificial diffusion functionality with the Incompressible Navier-Stokes application mode. See "Boundary Conditions" on page 111, "Point Settings" on page 121, and "Numerical Stability—Artificial Diffusion" on page 121. It is also possible to use corner smoothing with swirl flow. See "Corner Smoothing" on page 125.

The Swirl Velocity Property

Most application modes for momentum transport have a swirl flow formulation that can be activated through the **Swirl velocity** property. However, only the swirl flow formulation of the Incompressible Navier-Stokes has its own entry in the **Model**

Navigator. Figure 5-10 shows the **Application Mode Properties** dialog box, which you open by choosing **Properties** from the **Physics** menu.


Application Mode Properties

Properties

Default element type: Lagrange - $P_2 P_1$

Analysis type: Transient

Corner smoothing: Off

Weakly compressible flow: Off

Turbulence model: None

Realizability: Off

Non-Newtonian flow: Off

Two-phase flow: Single-phase flow

Swirl velocity: On

Weak constraints: Off

Constraint type: Ideal

Figure 5-10: The Application Mode Properties dialog box for the Swirl Flow application mode. The Swirl velocity property is set to On.

The **Swirl Velocity** property is **On** by default for the Swirl Flow application mode. Setting this property to **Off** turns off the swirl flow component, and the application mode becomes the Incompressible Navier-Stokes for 2D Axial Symmetry.

Example Model

The model “Swirl Flow Around a Rotating Disk” on page 216 in the *Chemical Engineering Module Model Library* illustrates how to use the 2D axisymmetric Swirl Flow application mode. It shows how to model a rotating disk confined in a cylindrical tank.

Non-Newtonian Flow

The Non-Newtonian Flow application mode assumes that the fluid is incompressible. It differs from the Incompressible Navier-Stokes application mode in that it extends the formulation of Equation 5-5 on page 109 by allowing the dynamic viscosity to be a function of the velocity field. The application mode provides a specific dialog box for the Power law and Carreau models for describing viscosity. Both of these models are suitable for flows where viscosity variations arise due to a fluid's shear rate. In addition, there is an option to enter your own expression for the dynamic viscosity.

Subdomain Equations

The application mode assumes that the fluid is incompressible, that is, that ρ is constant or very nearly so. This is the case for all liquids under normal conditions and also for gases at low velocities. For constant ρ, Equation 5-1 on page 108 reduces to

$$\nabla\cdot\mathbf{u} = 0 \qquad (5\text{-}24)$$

The stress tensor, τ, is an extension of Equation 5-5, including the shear-rate tensor, also known as the engineering strain-rate tensor. The tensor is denoted by $\dot{\gamma}$ and is defined by

$$\dot{\gamma} = (\nabla\mathbf{u} + (\nabla\mathbf{u})^T) \qquad (5\text{-}25)$$

Its magnitude, the shear rate, is defined by

$$\dot{\gamma} = |\dot{\gamma}| = \sqrt{\frac{1}{2}\dot{\gamma}:\dot{\gamma}} \qquad (5\text{-}26)$$

where the contraction operator ":" is defined by Equation 5-4 on page 108. Using the expressions in Equation 5-25 and Equation 5-26, the stress tensor used by this application mode can be written

$$\tau = \eta(\dot{\gamma})\dot{\gamma} = \eta(\dot{\gamma})(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) \qquad (5\text{-}27)$$

It can be seen that the only difference between this expression and the corresponding expression for the incompressible Navier-Stokes (Equation 5-7) is that in Equation 5-27, the dynamics viscosity, η, is a function of the shear rate, $\dot{\gamma}$.

With the incompressibility assumption and Equation 5-27, the momentum transport equations can be written:

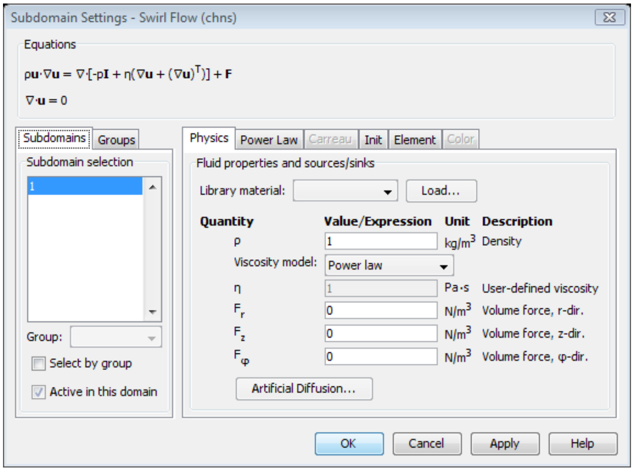
$$\rho\frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u}\cdot\nabla)\mathbf{u} = -\nabla p + \nabla\cdot\eta(\nabla\mathbf{u}+(\nabla\mathbf{u})^T) + \mathbf{F} \qquad (5\text{-}28)$$

where $\eta = \eta(\dot{\gamma})$.

Predefined Viscosity Models

The **Subdomain Settings** dialog box for the Non-Newtonian Flow application mode contains a list where you can select the viscosity model. As Figure 5-11 shows, the choices are

- Power law
- Carreau model
- User-defined model



Subdomain Settings - Swirl Flow (chns)

Equations

$\rho\mathbf{u}\cdot\nabla\mathbf{u} = \nabla\cdot[-p\mathbf{I} + \eta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)] + \mathbf{F}$

$\nabla\cdot\mathbf{u} = 0$

Subdomain selection: 1

Group:

☐ Select by group

☑ Active in this domain

Fluid properties and sources/sinks

Library material:

Quantity	Value/Expression	Unit	Description
ρ	1	kg/m^3	Density
Viscosity model:	Power law		
η	1	Pa·s	User-defined viscosity
F_r	0	N/m^3	Volume force, r-dir.
F_z	0	N/m^3	Volume force, z-dir.
F_φ	0	N/m^3	Volume force, φ-dir.

Figure 5-11: The Subdomain Settings dialog box for the Non-Newtonian Flow application mode.

The edit field for a user-defined viscosity is active only when you select **User-defined model**. Selecting **Power law** or **Carreau model** activates the corresponding page in the **Subdomain Settings** dialog box. These pages contain edit fields for the model parameters to the power law and the Carreau model, respectively.

POWER LAW

The power law is a so-called generalized Newtonian model. It prescribes

$$\eta = m\dot{\gamma}^{n-1} \tag{5-29}$$

where m and n are scalars that you can set to arbitrary values.

For $n > 1$, the power law describes a shear thickening (pseudoplastic) fluid. For $n < 1$, it describes a shear thinning (dilatant) fluid. A value of n equal to one gives the expression for a Newtonian fluid.

CARREAU MODEL

The Carreau expression gives the viscosity by the following four-parameter equation

$$\eta = \eta_\infty + (\eta_0 - \eta_\infty)[1 + (\lambda\dot{\gamma})^2]^{\frac{(n-1)}{2}} \tag{5-30}$$

where λ is a parameter with units of time, η_0 is the zero shear rate viscosity, η_∞ is the infinite shear rate viscosity and n is a dimensionless parameter. This expression is able to describe viscosity for mostly stationary polymer flow.

USER-DEFINED MODEL

The last variation of the Non-Newtonian application mode is based upon general expressions for the dynamic viscosity. A built-in expression variable or variable name for the shear rate, `sr_chnn`, makes it simple to quickly define arbitrary expressions of viscosity as a function of the shear rate.

Other Application-Mode Settings

The Non-Newtonian Flow application mode shares boundary conditions, point settings, and artificial diffusion functionality with the Incompressible Navier-Stokes application mode. See "Boundary Conditions" on page 111, "Point Settings" on page 121, and "Numerical Stability—Artificial Diffusion" on page 121. It is also possible to use corner smoothing in the Non-Newtonian Flow application mode. See "Corner Smoothing" on page 125.

Example Model

The model "Non-Newtonian Flow" on page 30 in the *Chemical Engineering Model Library* demonstrates the use of this application mode.

Turbulent Flow

Turbulence often appears in industrial applications that involve fluid flow. It can be considered a characteristic state of the flow that you can only describe in a statistical manner. In this statistical representation of turbulent flow, the mean quantities are of main interest. However, fluctuations in different field quantities also play an important role.

In order to predict the behavior of turbulent systems, numerical modeling can reduce the costs of experiments and prototype equipment. In these turbulence models it is important to account for the nature of the fluctuations around the mean flow. This is included in the Chemical Engineering Module as the Reynolds-averaged Navier-Stokes (RANS) equations.

The Navier-Stokes equations describe the basic phenomena of mass and momentum transport. These equations can also be used for turbulent flow simulations, although that requires a large number of elements to capture all the dynamics of the flow. An alternative is to consider the averaged equations, resulting in a hierarchy of equations and statistical unknowns. These unknown terms are modeled by so-called closure relations of which the eddy viscosity approach is the most common one. The Chemical Engineering Module includes two such closure schemes: k-ε and k-ω.

Because the k-ε Turbulence Model application mode and the k-ω Turbulence Model application mode depend on the same fundamental principles, this section describes both.

Subdomain Equations

Both application modes assume that the flow is incompressible and that the fluid is Newtonian. The flow is then in theory guided by the incompressible Navier-Stokes equations:

$$\begin{gathered}\nabla \cdot \mathbf{u} = 0 \\ \rho\frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u} \cdot \nabla)\mathbf{u} = -\nabla p + \nabla \cdot \eta(\nabla \mathbf{u} + (\nabla \mathbf{u})^T) + \mathbf{F}\end{gathered} \qquad (5\text{-}31)$$

Fundamental to the analysis of fluid flow is the Reynolds number:

$$\mathrm{Re}= \frac{\rho UL}{\eta} \tag{5-32}$$

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 fluid flow field eventually ends up in a chaotic state called turbulence.

REYNOLDS-AVERAGED NAVIER-STOKES (RANS) EQUATIONS

Once flow has become turbulent, all quantities fluctuate in time and space, and there is little advantage in having an exact and detailed picture of the flow. Instead, an averaged representation almost always provides sufficient information about the flow.

The Reynolds-averaged representation of turbulent flows divides the flow quantities into an averaged value and a fluctuating part,

$$\phi = \bar{\phi} + \phi' \tag{5-33}$$

where ϕ can represent any scalar quantity of the flow. In general, the mean value can vary in space and time. This is exemplified in Figure 5-12, which shows time averaging of one component of the velocity vector for nonstationary turbulence. The unfiltered flow has a time scale Δt_1. After a time filter with width $\Delta t_2 >> \Delta t_1$ has been applied, there is a fluctuating part, u'_i, and an average part, U_i. Because the flow field also varies on a time scale longer than Δt_2, U_i is still time dependent but is much smoother than the unfiltered velocity u_i.

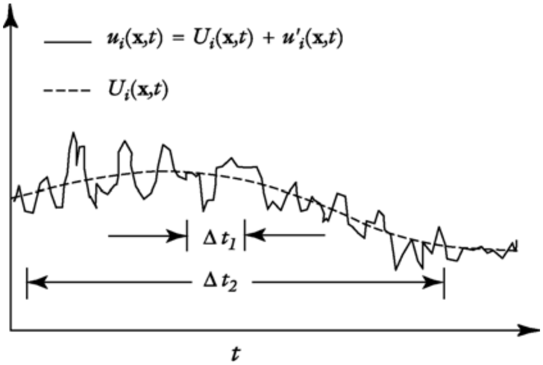


Figure 5-12: The unfiltered velocity component u_i, with a time scale Δt_1 and the averaged velocity component, U_i, with time scale Δt_2.

Decomposition of flow fields into an averaged part and a fluctuating part, followed by insertion into the Navier-Stokes equation, then averaging, gives the Reynolds-averaged Navier-Stokes (RANS) equations:

$$\rho\frac{\partial \mathbf{U}}{\partial t} + \rho\mathbf{U}\cdot\nabla\mathbf{U} + \nabla\cdot\overline{(\rho\mathbf{u}'\otimes\mathbf{u}')} = -\nabla P + \nabla\cdot\eta(\nabla\mathbf{U} + (\nabla\mathbf{U})^T) + \mathbf{F} \qquad (5\text{-}34)$$

$$\nabla\cdot\mathbf{U} = 0$$

where **U** is the averaged velocity field. ⊗ is the outer vector product. A comparison with Equation 5-31 indicates that the only difference is the appearance of the last term on the left hand side of Equation 5-34. This term represents interaction between the fluctuating velocities and is called the Reynolds stress tensor. This means that to obtain the mean flow characteristics, you need information about the small-scale structure of the flow. In this case, that information is the correlation between fluctuations in different directions.

The governing equations for the components of the Reynolds stress tensor follow from a procedure similar to that above. In 3D, this introduces six additional equations and six additional unknowns. Because the resulting equations contain additional unknowns emanating from higher-order statistics, this is still not sufficient for solving flow problems. Resolving this dilemma requires imposing assumptions about the flow, a so-called closure of a turbulence model.

EDDY VISCOSITY CLOSURE

The most common by far way to model turbulence is by assuming the turbulence to be of a purely diffusive nature. The deviating part of the Reynolds stress is then expressed by

$$\rho(\overline{\mathbf{u}' \otimes \mathbf{u}'}) - \frac{\rho}{3}\mathrm{trace}(\overline{\mathbf{u}' \otimes \mathbf{u}'})\mathrm{I} = -\eta_T(\nabla\mathbf{U} + (\nabla\mathbf{U})^T) \tag{5-35}$$

where η_T is the *eddy viscosity*, also known as the turbulent viscosity. The spherical part can be written

$$\frac{\rho}{3}\mathrm{trace}(\overline{\mathbf{u}' \otimes \mathbf{u}'})\mathrm{I} = \frac{2}{3}\rho k \tag{5-36}$$

where k is the turbulent kinetic energy. In simulations of incompressible flows, this term is included in the pressure, but when the absolute pressure level is of importance (in compressible flows, for example) this term must be explicitly modeled.

The k-ε Turbulence Model

The k-ε model is one of the most used turbulence models for industrial applications. The Chemical Engineering Module includes the standard k-ε model (Ref. 1). This model introduces two additional transport equations and two dependent variables: the turbulence kinetic energy, k, and the dissipation rate of turbulence energy, ε. Turbulent viscosity is modeled by

$$\eta_T = \rho C_\mu \frac{k^2}{\varepsilon} \tag{5-37}$$

where C_μ is a model constant.

The transport equation for k can be derived by taking the trace of the equations for the Reynolds stresses:

$$\rho\frac{\partial k}{\partial t} - \nabla\cdot\left[\left(\eta + \frac{\eta_T}{\sigma_k}\right)\nabla k\right] + \rho\mathbf{U}\cdot\nabla k = \frac{1}{2}\eta_T(\nabla\mathbf{U} + (\nabla\mathbf{U})^T)^2 - \rho\varepsilon \tag{5-38}$$

An equation for ε can be derived in a similar manner. That equation is, however, impossible to model on a term-by-term basis. Instead, all terms that do not have an equivalent term in the k equation are discarded. The resulting equation reads:

$$\rho\frac{\partial\varepsilon}{\partial t}-\nabla\cdot\left[\left(\eta+\frac{\eta_T}{\sigma_\varepsilon}\right)\nabla\varepsilon\right]+\rho\mathbf{U}\cdot\nabla\varepsilon = \frac{1}{2}C_{\varepsilon1}\frac{\varepsilon}{k}\eta_T(\nabla\mathbf{U}+(\nabla\mathbf{U})^T)^2-\rho C_{\varepsilon2}\frac{\varepsilon^2}{k} \quad (5\text{-}39)$$

The model constants in the above equations are determined from experimental data (Ref. 1); their values are listed in Table 5-1.

TABLE 5-1: MODEL CONSTANTS IN EQUATION 5-37, EQUATION 5-38, AND EQUATION 5-39

CONSTANT	VALUE
C_μ	0.09
$C_{\varepsilon1}$	1.44
$C_{\varepsilon2}$	1.92
σ_k	1.0
σ_ε	1.3

The k-ε turbulence model relies on several assumptions, the most important of which are that the Reynolds number is high enough and that the turbulence is in equilibrium in boundary layers, which means that production equal dissipation. The assumptions limit the accuracy of the model, since this is not always true. The spatial extension of recirculation zones, for example, is underpredicted by the k-ε turbulence model. Furthermore, in the description of rotating flows, the model often shows poor agreement with experimental data. In most cases, the limited accuracy is often a fair trade for the amount of computational resources saved compared to more complicated turbulence models.

The k–ω Turbulence Model

Much of the k-ε model's inaccuracy can be attributed to the rather inaccurate modeling of the ε equation. An alternative to modeling the dissipation is to model transport of the dissipation per unit turbulent kinetic energy, ω. The turbulent viscosity is modeled by:

$$\eta_T = \rho\frac{k}{\omega} \quad (5\text{-}40)$$

The transport of turbulent kinetic energy and ω is given by

$$\rho\frac{\partial k}{\partial t}+\rho\mathbf{U}\cdot\nabla k = \nabla\cdot[(\eta+\sigma_k\eta_T)\nabla k]+\frac{1}{2}\eta_T(\nabla\mathbf{U}+\nabla\mathbf{U}^T)^2-\rho\beta_k k\omega \quad (5\text{-}41)$$

and

$$\rho\frac{\partial\omega}{\partial t} + \rho\mathbf{U}\cdot\nabla\omega = \nabla\cdot[(\eta + \sigma_\omega\eta_T)\nabla\omega] + \frac{\alpha}{2}\eta_T\frac{\omega}{k}(\nabla\mathbf{U} + \nabla\mathbf{U}^T)^2 - \beta\rho\omega^2 \qquad (5\text{-}42)$$

where the previous equations use the following closure constants and functions:

$$\alpha = \frac{13}{25},\ \beta = \beta_0 f_\beta,\ \beta_k = \beta_{k,0} f_{\beta,k},\ \sigma_w = \sigma_k = \frac{1}{2}$$

$$\beta_0 = \frac{9}{125},\ f_\beta = \frac{1 + 70\chi_\omega}{1 + 80\chi_\omega},\ \ \chi_\omega \equiv \left|\frac{\Omega_{ij}\Omega_{jk}S_{ki}}{(\beta_{k,0}\omega)^3}\right|$$

$$\beta_{0,k} = \frac{9}{100},\ f_{\beta,k} = \begin{cases} 1 & \chi_k \le 0 \\ \dfrac{1 + 680\chi_k^2}{1 + 400\chi_k^2} & \chi_k > 0 \end{cases},\ \ \chi_k \equiv \frac{1}{\omega^3}\nabla k\cdot\nabla\omega$$

The tensors, Ω_{ij} and S_{ij}, are the mean rotation-rate tensor and the mean strain-rate tensor, defined by

$$\Omega_{ij} = \frac{1}{2}\left(\frac{\partial U_i}{\partial x_j} - \frac{\partial U_j}{\partial x_i}\right),\ S_{ij} = \frac{1}{2}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) \qquad (5\text{-}43)$$

The use of the correction factors f_β and $f_{\beta,k}$ is a modification of the original (1988) k-ω model that Wilcox made for the updated model from 1998. The corrections improve the accuracy for free-shear flows significantly by decreasing the spreading of such flows.

There are no guarantees that one turbulence model provides better results than the other for a certain flow case. As a general rule, the k-ω model behaves better close to walls and usually predicts free shear flows more accurately than the k-ε model. For external flows, on the other hand, results obtained by the k-ω model have a tendency to be strongly dependent on the free-stream values of k and ω.

It is often harder to reach convergence with the k-ω model than with the k-ε model. A way to reach convergence is to first solve the problem with the k-ε model and then change to the k-ω model. The application mode automatically changes the initial values so that the k-ε result becomes the initial guess for the k-ω model.

Boundary Conditions

The turbulence application modes have the same boundary types as the Incompressible Navier-Stokes application modes; see “Boundary Conditions” on page 111. With exception for the boundary type **Wall**, the functionality described for the incompressible Navier-Stokes equations applies to the Reynolds-averaged Navier-Stokes equations as well. Note that any expression that in the incompressible Navier-Stokes formulation is of the form $g(\eta)$ is of the form $g(\eta + \eta_T)$ for the turbulence application modes.

All boundary conditions need additional conditions for the turbulence transport equations. The following sections describe these boundary conditions for each boundary type.

WALL

Slip

The Slip boundary condition assumes that there is no viscous interaction between the wall and the fluid. In addition to the conditions prescribed for the momentum equations, this boundary condition also prescribes

$$\mathbf{n}\cdot\left(\left(\eta+\frac{\eta_T}{\sigma_k}\right)\nabla k-\rho\mathbf{U}k\right)=0 \;\;/\;\; (\mathbf{n}\cdot((\eta+\sigma_k\eta_T)\nabla k-\rho\mathbf{U}k)=0)$$

$$\mathbf{n}\cdot\left(\left(\eta+\frac{\eta_T}{\sigma_\varepsilon}\right)\nabla\varepsilon-\rho\mathbf{U}\varepsilon\right)=0 \;\;/\;\; (\mathbf{n}\cdot((\eta+\sigma_\omega\eta_T)\nabla\omega-\rho\mathbf{U}\omega)=0)$$

where the expressions to the left are used for the k-ε Turbulence Model and the expressions to the right are used for the k-ω Turbulence Model.

Logarithmic Wall Function

Turbulence close to a solid wall is very different from isotropic free-stream turbulence. This must be accounted for in a proper model.

There are basically two approaches to account for solid walls subjected to turbulent flow. The first approach, used for low Reynolds number turbulence models, modifies the equations by additional terms and factors that account for near-wall effects. In such a case, you must refine the mesh near the wall to resolve the so-called viscous sublayer. Such methods are of interest for moderate Reynolds numbers, where near-wall resolution lead to a reasonable number or elements.

In the second approach, which is used by these application modes, an empirical relation between the value of velocity and wall friction replaces the thin boundary layer near the wall. Such relations are known as wall functions and are accurate for high Reynolds numbers and situations where pressure variations along the wall are not very large. However, this approach can often be used outside its frame of validity with reasonable success.

Logarithmic wall functions applied to finite elements assume that the computational domain begins a distance δ_w from the real wall. They also assume that the flow is parallel to the wall and that the velocity can be described by

$$U^+ = \frac{U}{u_\tau} = \frac{1}{\kappa}\ln\left(\frac{\delta_w}{l^*}\right) + C^+ \tag{5-44}$$

Here U is the velocity parallel to the wall, u_τ is the friction velocity defined by

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}}\,, \tag{5-45}$$

κ denotes the Kármán's constant (about 0.42), and C^+ is a universal constant for smooth walls. In the application mode, C^+ is defined as an application scalar variable named `Cplus_chns` and has a default value of 5.5 (it is possible to change this value in the **Application Scalar Variables** dialog box). Further, l^* is known as the viscous length scale and is defined by

$$l^* = \frac{\eta}{\rho u_\tau} \tag{5-46}$$

You must specify the distance δ_w or its equivalent in viscous units, $\delta_w^+ = \delta_w / l^*$. Their internal relation is given by

$$\delta_w^+ = \frac{\rho C_\mu^{1/4} k^{1/2} \delta_w}{\eta} \tag{5-47}$$

The logarithmic wall functions are formally valid for values of δ_w^+ between 30 and 100. For high Reynolds number, the upper limit can be extended to a few hundred. δ_w^+ is available as the variable `dwplus_chns` and is defined on the boundaries. Because the wall function assumes that the flow is parallel to the wall, the velocity component perpendicular to the wall is zero.

The boundary conditions for k, ε, and ω are derived from the assumption that turbulent production equals the dissipation:

$$\nabla k \cdot \mathbf{n} = 0 , \varepsilon = \frac{C_\mu^{3/4} k^{3/2}}{\kappa \delta_w} , \omega = \frac{k^{1/2}}{C_\mu^{1/4} \kappa \delta_w} \tag{5-48}$$

Sliding Wall

This boundary condition assumes that the wall is moving with a given velocity tangential to the wall. It extends the logarithmic wall function concept by changing Equation 5-44 to

$$\frac{(U - \mathrm{U_w})}{u_\tau} = \frac{1}{\kappa}\ln\left(\frac{\delta_w}{l^*}\right) + C^+$$

where $\mathrm{U_w}$ is calculated in the same way as described for the Incompressible Navier-Stokes application mode in the section “Sliding Wall” on page 114.

INLET BOUNDARIES

At inlet boundaries, the functionality for the RANS equations are the same as described for the incompressible Navier-Stokes in the section “Inlet” on page 115. In addition, values for the two turbulent quantities need to be specified (k and ε, or k and ω). Alternatively, you can specify a turbulent length scale, $\mathrm{L_T}$, and a turbulent intensity, $\mathrm{I_T}$. They are related to the turbulent variables via

$$k = \frac{3}{2}(|\mathbf{U}| I_T)^2 , \varepsilon = C_\mu^{3/4} \frac{k^{3/2}}{L_T} , \omega = C_\mu^{-1/4} \frac{k^{1/2}}{L_T} \tag{5-49}$$

A value below 0.1% is a low intensity. Good wind tunnels can produce values of $\mathrm{I_T}$ as low as 0.05%. Fully turbulent flows have intensities between five and ten percent.

The turbulent length scale is a measure of the size of the eddies that are not resolved. For free-stream flows these are typically very small (in the order of centimeters). The length scale cannot be zero, however, because that would imply infinite dissipation. You can use the following table as a guideline when specifying L_T (Ref. 3):

TABLE 5-2: TURBULENT LENGTH SCALES FOR TWO-DIMENSIONAL FLOWS

FLOW CASE	$\mathbf{L}_T$	$\mathbf{L}$
Mixing layer	0.07L	Layer width
Plane jet	0.09L	Jet half width
Wake	0.08L	Wake width
Axisymmetric jet	0.075L	Jet half width

TABLE 5-2: TURBULENT LENGTH SCALES FOR TWO-DIMENSIONAL FLOWS

FLOW CASE	$\mathbf{L}_T$	$\mathbf{L}$
Boundary layer ($\partial p/\partial x = 0$)		
– Viscous sublayer and log-layer	$\kappa l_{\mathrm{w}}(1-\exp(-l_{\mathrm{w}}^{+}/26))$	Boundary layer thickness
– Outer layer	0.09L	
Pipes and channels (fully developed flows)	0.07L	Pipe diameter or channel width

where l_{w} is the wall distance and $l_{\mathrm{w}}^{+} = l_{\mathrm{w}}/l^{*}$ is the wall distance in viscous units.

OUTLET BOUNDARIES

At outlet boundaries, the functionality for the RANS equations are the same as described for the incompressible Navier-Stokes equations in the section "Outflow" on page 116. In addition, convective flux conditions are prescribed for the turbulence variables:

$$\mathbf{n}\cdot\nabla k = 0$$
$$\mathbf{n}\cdot\nabla\varepsilon = 0 \;/\; \mathbf{n}\cdot\nabla\omega = 0$$

SYMMETRY BOUNDARIES

The equations for this condition read

$$(-p\mathbf{I}+(\eta+\eta_T)(\nabla\mathbf{U}+(\nabla\mathbf{U})^T))\cdot\mathbf{t}=0$$
$$\mathbf{n}\cdot\mathbf{U}= 0$$
$$\mathbf{n}\cdot\left(\left(\eta+\frac{\eta_T}{\sigma_k}\right)\nabla k-\rho\mathbf{U}k\right)= 0 \;/\; (\mathbf{n}\cdot((\eta+\sigma_k\eta_T)\nabla k-\rho\mathbf{U}k)= 0)$$
$$\mathbf{n}\cdot\left(\left(\eta+\frac{\eta_T}{\sigma_\varepsilon}\right)\nabla\varepsilon-\rho\mathbf{U}\varepsilon\right)= 0 \;/\; (\mathbf{n}\cdot((\eta+\sigma_\omega\eta_T)\nabla\omega-\rho\mathbf{U}\omega)= 0)$$

where the constraints on the turbulent quantities express zero total normal flux. The expressions to the left are used for the k-ε Turbulence Model and the expressions to the right are used for the k-ω Turbulence Model.

OPEN BOUNDARIES

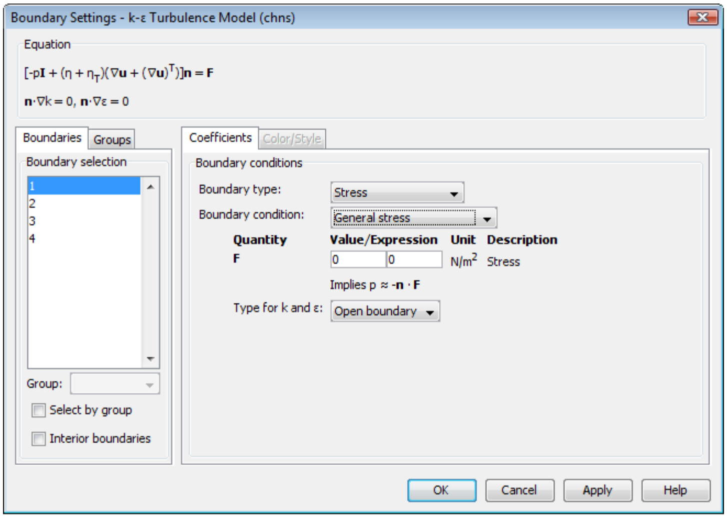
The expressions for the momentum equations are the same as for the Incompressible Navier-Stokes (see page 119), except that $\eta+\eta_T$ replaces η. Homogeneous Neumann conditions are prescribed for the turbulence variables

$$\mathbf{n} \cdot \nabla k = 0$$
$$\mathbf{n} \cdot \nabla \varepsilon = 0 \ / \ \mathbf{n} \cdot \nabla \omega = 0$$

Note that these are the same boundary conditions used for outlet boundaries. This expresses that the turbulent characteristic of whatever is outside the computational domain is guided by the flow inside the computational domain. Such an assumption is physically reasonable as long as relatively small amounts of fluid enters via an open boundary. Large amount of fluids may, however, never enter via an open boundary. If this is the case, switch to an inlet condition instead.

STRESS CONDITIONS

Because this type of boundary condition is very general, different conditions have to be prescribed for the turbulence variables depending on what stresses that are prescribed. As Figure 5-13 shows, there are three choices for the turbulence variables: **Inlet**, **Outlet**, and **Open boundary**. The functionality of these boundary types are described in the previous sections.



Boundary Settings - k-ε Turbulence Model (chns)

Equation

$[-p\mathbf{I} + (\eta + \eta_T)(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)]\mathbf{n} = \mathbf{F}$

$\mathbf{n}\cdot\nabla k = 0, \ \mathbf{n}\cdot\nabla\varepsilon = 0$

Boundary conditions

Boundary type: Stress

Boundary condition: General stress

Quantity	Value/Expression		Unit	Description
F	0	0	N/m^2	Stress

Implies p ≈ -**n** · **F**

Type for k and ε: Open boundary

Figure 5-13: The Boundary Settings dialog box for the k-ε Turbulence Model application mode. For the Stress boundary type, there are three choices for the type for the turbulence variables.

Realizability Constraints

A stagnation point is defined as a point on a surface where a streamline ends. Typical examples are the leading edge of a flat plate or the front edge of a wing profile. The equilibrium assumption made when deriving the turbulent transport equations are not valid in such points. The result is an excessive production of turbulence known as the stagnation point anomaly. To cope with this issue, you must activate a realizability constraint.

The eddy-viscosity model of the Reynolds stress tensor can be written

$$\rho\overline{u_iu_j} = -2\eta_T S_{ij} + \frac{2}{3}\rho k\delta_{ij}$$

where δ_{ij} is the Kronecker delta and S_{ij} is the strain rate tensor. Durbin (Ref. 4) showed that the fact that diagonal elements of the Reynolds stress tensor must be nonnegative can be used to derive the following constraint on the turbulent viscosity:

$$\eta_T \le \frac{\rho k}{3\cdot\max(\lambda_\alpha)} \tag{5-50}$$

Here, $\max(\lambda_\alpha)$ is the largest eigenvalue of the strain rate tensor. When using realizability constraint for the k-ε model, the turbulent viscosity is given by

$$\eta_T = \min\left(\rho C_\mu\frac{k^2}{\varepsilon}, \frac{\rho k}{3\cdot\max(\lambda_\alpha)}\right) \tag{5-51}$$

The corresponding equivalence for the k-ω model is also available. In two dimensions, it is possible to derive an exact expression for the right-hand side of Equation 5-51:

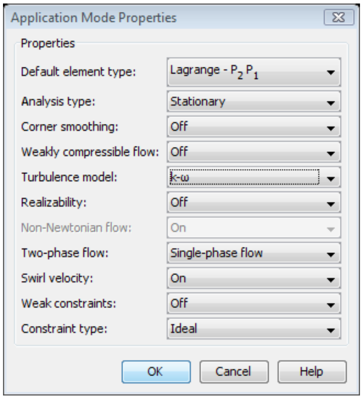
$$\eta_T \le \frac{\rho k\sqrt{2}}{3\sqrt{S_{ij}S_{ij}}} \tag{5-52}$$

In tree dimensions, however, you can only find an approximate upper limit without calculating all the eigenvalues of S_{ij}:

$$\eta_T \le \frac{\rho k}{\sqrt{6}\sqrt{S_{ij}S_{ij}}} \tag{5-53}$$

To activate the realizability constraint, use the **Application Mode Properties** dialog box (Figure 5-14), which you open by choosing **Properties** from the **Physics** menu.

Because an active realizability constraint implies a lower turbulent viscosity than the unconstrained case, the convergence will be hampered. If an active realizability constraint hinders convergence, try to first solve without realizability and then activate the constraint.



Default element type: Lagrange - P_2 P_1

Analysis type: Stationary

Corner smoothing: Off

Weakly compressible flow: Off

Turbulence model: k-ω

Realizability: Off

Non-Newtonian flow: On

Two-phase flow: Single-phase flow

Swirl velocity: On

Weak constraints: Off

Constraint type: Ideal

Figure 5-14: The Application Mode Properties dialog box for the k-ε Turbulence Model application mode.

The Turbulence Model Property

The **Turbulence model** property in the **Application Mode Properties** dialog box (see Figure 5-14) has the following options

- **None**
- **k–ε**
- **k–ω**

This property can be used to change between the k-ε Turbulence Model application mode and the k-ω Turbulence Model application mode. This can be useful because it can be hard to reach convergence using the k-ω model. A standard trick is then to start from a k-ε solution. When changing turbulence model, COMSOL automatically updates the **Initial value** edit fields in the **Subdomain Settings** dialog box.

The **Turbulence model** property is available for many application modes that per default do not utilize any turbulence model.

Example Models

The *Chemical Engineering Module Model Library* contains many models that demonstrate the k-ε Turbulence Model application mode and the k-ω Turbulence Model application mode.

The model Turbulent Flow over a Backward-Facing Step on page 87 of the *Chemical Engineering Module Model Library* demonstrates the accuracy of the *k*-ε in a 2D case and compares the result to experimental data.

An example in 3D is given in the model Turbulent Flow Through a Bending Pipe on page 65 of the *Chemical Engineering Module Model Library.* It demonstrates how to properly use the logarithmic wall functions. It also makes as comparison of the accuracy between the *k*-ε model and the *k*-ω model.

References

1. D.C. Wilcox, *Turbulence Modeling for CFD*, DCW Industries Inc., 1998.

2. D.M. Driver and H.L. Seegmiller, “Features of a reattaching turbulent shear layer in diverging channel flow,” *AIAA Journal*, vol. 23, pp. 163–171, 1985.

3. H.K. Versteeg and W. Malalasekera, *An introduction to Computational Fluid Dynamics*, Prentice Hall,1995.

4. A. Durbin, “On the *k*-ε stagnation point anomality,” *International Journal of Heat and Fluid Flow*, vol. 17, pp. 89–90, 1986.

Weakly Compressible Navier-Stokes

The Weakly Compressible Navier-Stokes application mode allows you to simulate flows where the density of the fluid varies, for instance, as a function of temperature or composition. You select the application mode from the Flow with Variable Density folder in the Model Navigator.

The Weakly Compressible Navier-Stokes application mode contains the fully compressible formulation of the continuity equation and momentum equations:

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

$$\rho\frac{\partial \mathbf{u}}{\partial t} + \rho\mathbf{u} \cdot \nabla\mathbf{u} = -\nabla p + \nabla \cdot \left(\eta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \left(\frac{2}{3}\eta - \kappa_{\mathrm{dv}}\right)(\nabla \cdot \mathbf{u})\mathbf{I}\right) + \mathbf{F} \tag{5-54}$$

The stress tensor used in Equation 5-54 describes a Newtonian fluid, with an added term κ_{dv}. This term expresses the deviation from Stokes' assumption, which states that the fluid particles are in thermodynamic equilibrium with their neighbors. It is very rare that a fluid shows a significant deviation from Stokes' assumption, and κ_{dv} is therefore by default set to zero.

Note: Note that the Weakly Compressible Navier-Stokes application was designed solely to model flow with variable density, where a second application mode—typically a heat balance—controls the density. This coupling is automatically set up by the Non-Isothermal Flow predefined multiphysics coupling (see "Turbulent Non-Isothermal Flow" on page 1).

THE MACH NUMBER LIMIT

An important dimensionless number in fluid dynamics is the Mach number, Ma, defined by

$$\mathrm{Ma} = \frac{|\mathbf{u}|}{a} \tag{5-55}$$

where a is the speed of sound. A flow is formally incompressible when Ma=0. This is theoretically achieved by letting the speed of sound tend to infinity. The Navier-Stokes equations will then have the numerical property that a disturbance anywhere in the computational domain will instantaneously spread to the entire domain. This results in a parabolic equation system.

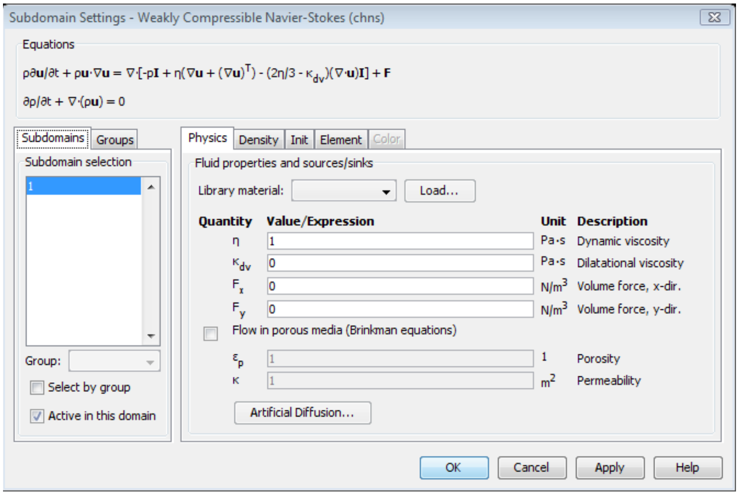
The fully compressible Navier-Stokes equations, Equation 5-1 through Equation 5-3, have a finite speed of sound and hence a Mach number larger than zero. This will have no numerical significance as long as the Mach number is well below one. However, when the Mach number approaches unity, the equations will turn from parabolic to hyperbolic. When this happens, the numerical properties of the equation changes, one of several implications being that the boundary conditions used for incompressible Navier-Stokes equations become invalid. The Weakly Compressible Navier-Stokes application mode uses the same boundary condition as the Incompressible Navier-Stokes application mode, which implies that the Weakly Compressible Navier-Stokes application mode cannot be used for flows with Mach number larger than or equal to one.

The practical Mach number limit is however lower than one. The first reason for this is the sound wave transport term that has been neglected in the Convection and Conduction application mode (see "Pressure Work" on page 200). This term becomes important already at moderate Mach numbers. The second reason is that already at moderate Mach number, the fully compressible Navier-Stokes equations start to display very sharp gradients. To handle these gradient, special numerical techniques are needed. 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.

Subdomain Settings

The **Subdomain Settings** dialog box is shown in Figure 5-15. Unlike the Incompressible Navier-Stokes application mode, the edit field for the density, ρ, is not on the **Physics** page. Instead, it has its own page labeled **Density**. The content of the **Density** page is shown in Figure 5-16. In addition to the edit field for the density, ρ, there are two check boxes. These are used to control the exact weak expression for the Galerkin least-squares (GLS) artificial diffusion. The GLS method is necessary to get a stable finite element discretization of the equation system 5-54. You can define the density as a function of any of the other dependent variables, but GLS can only stabilize with respect to pressure and temperature, which are the most common variables.

To learn more about GLS, read Chapter 15, “Stabilization Techniques,” of the *COMSOL Multiphysics Modeling Guide*.



Subdomain Settings - Weakly Compressible Navier-Stokes (chns)

Equations

$\rho\partial\mathbf{u}/\partial t + \rho\mathbf{u}\cdot\nabla\mathbf{u} = \nabla\cdot[-p\mathbf{I} + \eta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - (2\eta/3 - \kappa_{dv})(\nabla\cdot\mathbf{u})\mathbf{I}] + \mathbf{F}$

$\partial\rho/\partial t + \nabla\cdot(\rho\mathbf{u}) = 0$

☐ Select by group

☑ Active in this domain

Fluid properties and sources/sinks

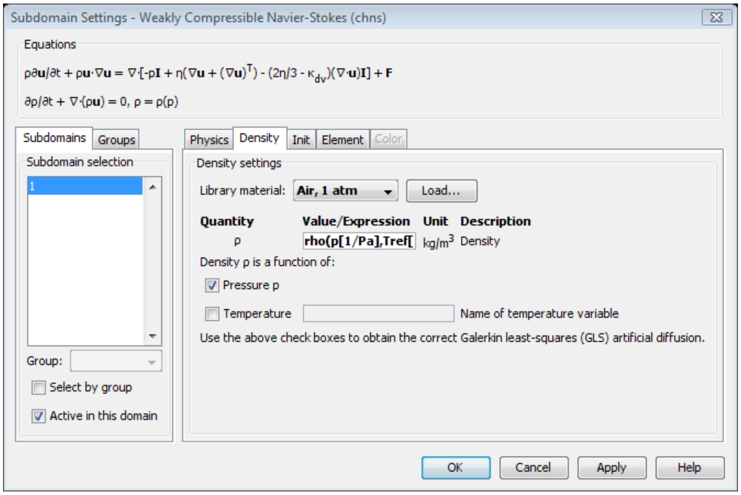
Library material:

Quantity	Value/Expression	Unit	Description
η	1	Pa·s	Dynamic viscosity
κ_{dv}	0	Pa·s	Dilatational viscosity
F_x	0	N/m^3	Volume force, x-dir.
F_y	0	N/m^3	Volume force, y-dir.

☐ Flow in porous media (Brinkman equations)

ε_p	1	1	Porosity
κ	1	m^2	Permeability

Figure 5-15: The Subdomain Settings dialog box for the Weakly Compressible Navier-Stokes application mode. The Physics page is selected.



Subdomain Settings - Weakly Compressible Navier-Stokes (chns)

Equations

$\rho\partial\mathbf{u}/\partial t + \rho\mathbf{u}\cdot\nabla\mathbf{u} = \nabla\cdot[-p\mathbf{I} + \eta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - (2\eta/3 - \kappa_{dv})(\nabla\cdot\mathbf{u})\mathbf{I}] + \mathbf{F}$

$\partial\rho/\partial t + \nabla\cdot(\rho\mathbf{u}) = 0, \rho = \rho(p)$

☐ Select by group

☑ Active in this domain

Density settings

Library material: **Air, 1 atm**

Quantity	Value/Expression	Unit	Description
ρ	**rho(p[1/Pa],Tref[**	kg/m^3	Density

Density ρ is a function of:

☑ Pressure p

☐ Temperature Name of temperature variable

Use the above check boxes to obtain the correct Galerkin least-squares (GLS) artificial diffusion.

Figure 5-16: The Subdomain Settings dialog box for the Weakly Compressible Navier-Stokes application mode. The Density page is selected.

The weakly compressible description of the Navier-Stokes equations is useful for describing free convection due to changes in density. In such a case, set the volume force term, **F**, to the gravity force, ρ**g**, where **g** is a vector that represents the acceleration due to gravity.

Other Settings

The Weakly Compressible Navier-Stokes application mode shares boundary conditions, point settings, and artificial diffusion functionality with the Incompressible Navier-Stokes application mode; see “Boundary Conditions” on page 111, “Point Settings” on page 121, and “Numerical Stability—Artificial Diffusion” on page 121. The Corner Smoothing property can be applied to the Weakly Compressible Navier-Stokes application mode as well; see “Corner Smoothing” on page 125.

Turbulent Weakly Compressible Flow

When a turbulence model is used, the Reynolds stress tensor (see the section “Turbulent Flow” on page 135) is modeled by

$$\rho(\overline{\mathbf{u}' \otimes \mathbf{u}'}) = -\eta_T(\nabla\mathbf{U} + (\nabla\mathbf{U})^T) - \frac{2}{3}\rho k$$

where $\otimes$ is the outer vector product.

The turbulent transport equations (k-ε or k-ω) are used in their fully compressible formulations (Ref. 1).

Reference

1. L. Ignat, D. Pelletier, and F. Ilinca, “A universal formulation of two-equation models for adaptive computation of turbulent flows,” *Computer methods in applied mechanics and engineering*, vol. 189, pp. 1119–1139, 2000.

Darcy's Law

In a porous medium, the global transport of momentum by shear stresses in the fluid is often negligible because the pore walls impede momentum transport to the fluid outside the individual pores. A detailed description, down to the resolution of every pore, is not practical in most applications. A homogenization of the porous and fluid media into a single medium is a common alternative approach. Darcy's law together with the continuity equation and equation of state for the pore fluid (or gas) provide a complete mathematical model suitable for a wide variety of applications involving porous media flows, for which the pressure gradient is the major driving force.

Subdomain Equations

Darcy's law operates with the volume-averaged flow variables such as the flow velocity, pressure, and density, for example. These averaged properties are defined at any point inside the medium by means of averaging of the actual fluid properties over a certain volume surrounding the point. The volume is small compared to the typical macroscopic dimensions of the problem, but it is large enough to contain many pores and solid matrix elements.

Darcy's law states that the velocity field is determined by the pressure gradient, the fluid viscosity, and the structure of the porous medium:

$$\mathbf{u} = -\frac{\kappa}{\eta}\nabla p \qquad (5\text{-}56)$$

In this equation, κ denotes the hydraulic permeability of the porous medium (m^2), η the dynamic viscosity of the fluid (kg/(m · s)), p the pressure (Pa), and $\mathbf{u}$ the velocity (m/s).

The Darcy's Law application mode in the Chemical Engineering Module combines Darcy's law with the continuity equation:

$$\frac{\partial}{\partial t}(\rho\varepsilon) + \nabla\cdot(\rho\mathbf{u}) = 0 \qquad (5\text{-}57)$$

In the above equation, ρ is the density of the fluid (kg/m^3), and ε is the porosity. The application mode can be coupled with an energy balance, and the fluid density can be a function of the temperature, pressure, and composition (for mixture flows). For gas flows in porous media, the dependence is given by the ideal gas law:

$$\rho = \frac{pM}{RT} \qquad (5\text{-}58)$$

where R= 8.314 J/(mol · K) is the universal gas constant, M is the molecular weight of the gas (kg/mol), and T is the temperature (K).

Boundary Conditions

The boundary conditions you can impose in the Darcy's Law application mode comprise the pressure condition:

$$p = p_0 \qquad (5\text{-}59)$$

an impervious or symmetric boundary condition:

$$-\frac{\kappa}{\eta}\nabla p \cdot \mathbf{n} = 0 \qquad (5\text{-}60)$$

and a specific inflow/outflow perpendicular to the boundary:

$$-\frac{\kappa}{\eta}\nabla p \cdot \mathbf{n} = u_0 \qquad (5\text{-}61)$$

The default boundary condition on an interior boundary is continuity:

$$\mathbf{n} \cdot (\rho_1\mathbf{u}_1 - \rho_2\mathbf{u}_2) = 0 \qquad (5\text{-}62)$$

The pressure boundary condition is also available on interior boundaries.

Example Model

The model "Variations in Density in Porous Media Flow" on page 46 of the *Chemical Engineering Module Model Library* exemplifies the use of the Darcy's Law application mode for modeling flows of varying density in porous media.

The Brinkman Equations

The Brinkman equations describe flows 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. You can use the Brinkman Equations application mode for modeling combinations of porous media and free flow domains. These types of problems are often encountered in applications such as monolithic reactors and fuel cells.

Introduction

In porous subdomains, 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, which is occupied by pores. Thus, porosity can vary from zero for pure solid regions to unity for subdomains of free flow.

The physical properties of the fluid, such as density and viscosity, and also the pressure are defined as so-called intrinsic volume averages that correspond to a unit volume of 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 velocities are defined as so-called superficial volume averages, and they correspond to a unit volume of the medium including both pores and matrix. They are sometimes called Darcy velocities, defined as volume flow rates 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.

The above presented approach eliminates the need for explicit consideration of interfaces. Thus, the flow can be modeled by using the same unknown variables for the entire domain including free flow subdomains and porous subdomains. The distinction is made via switching on and off certain terms in the governing equations. The boundaries between adjacent subdomains are treated as interior boundaries.

Subdomain Equations

The flow is governed by a combination of the continuity equation and momentum balance equation

$$\frac{\partial}{\partial t}(\varepsilon\rho)+\nabla\cdot(\rho\mathbf{u}) = 0 \tag{5-63}$$

$$\frac{\rho}{\varepsilon}\frac{\partial\mathbf{u}}{\partial t}+\frac{\eta}{\kappa}\mathbf{u} = \nabla\cdot\left[-p\mathbf{I}+\frac{1}{\varepsilon}\left\{\eta(\nabla\mathbf{u}+(\nabla\mathbf{u})^T)-\left(\frac{2}{3}\eta-\kappa_{\mathrm{dv}}\right)(\nabla\cdot\mathbf{u})\mathbf{I}\right\}\right]+\mathbf{F} \tag{5-64}$$

where η and κ_{dv} denote, respectively, the dynamic and dilatational viscosities of the fluid (both in kg/(m · s)), $\mathbf{u}$ is the velocity vector (m/s), ρ is the density of the fluid (kg/m^3), p is the pressure (Pa), ε is the porosity, and κ is the permeability of the porous medium (m^2). Influence of gravity and other body forces can be accounted for via the force term $\mathbf{F}$ (kg/(m^2·s^2)).

In case of a flow with variable density, Equation 5-63 and Equation 5-64 must be solved together with the equation of state that expresses the density as a function of the temperature and pressure (see Equation 5-58).

For subdomains of free flow, Equation 5-63 and Equation 5-64 transform into the Navier-Stokes equations

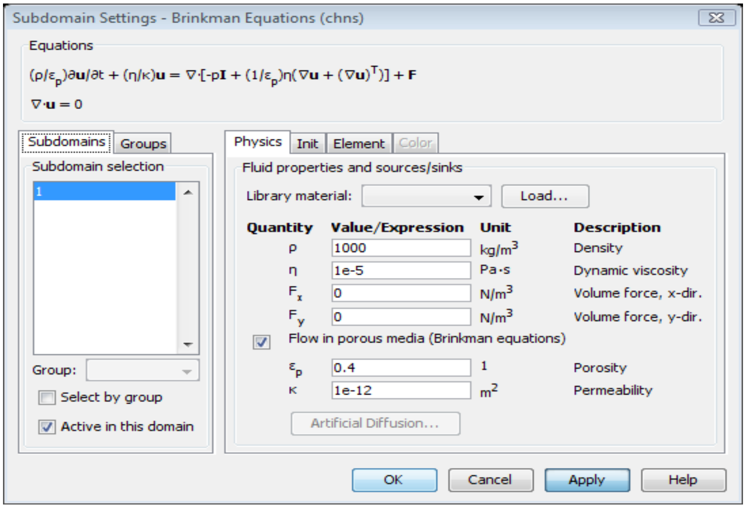
$$\frac{\partial\rho}{\partial t}+\nabla\cdot(\rho\mathbf{u}) = 0 \tag{5-65}$$

$$\begin{aligned}\rho\frac{\partial\mathbf{u}}{\partial t}+\rho(\mathbf{u}\cdot\nabla)\mathbf{u} &= \\ = \nabla\cdot\left[-p\mathbf{I}+\eta(\nabla\mathbf{u}+(\nabla\mathbf{u})^T)-\left(\frac{2}{3}\eta-\kappa_{\mathrm{dv}}\right)(\nabla\cdot\mathbf{u})\mathbf{I}\right]+\mathbf{F}\end{aligned} \tag{5-66}$$

The switch is achieved by setting $\varepsilon = 1$ everywhere and replacing the Darcy term in the left-hand side of the momentum equation by the advection term.

The Brinkman Equations application mode is fully integrated with the Incompressible Navier-Stokes application mode and other application modes for fluid flow. You can switch between free flow and porous media flow in a given subdomain by selecting the

Flow in porous media (Brinkman equations) check box in the **Subdomain Settings** dialog box; see Figure 5-17.



Subdomain Settings - Brinkman Equations (chns)

Equations

$(\rho/\varepsilon_p)\partial\mathbf{u}/\partial t + (\eta/\kappa)\mathbf{u} = \nabla\cdot[-p\mathbf{I} + (1/\varepsilon_p)\eta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)] + \mathbf{F}$

$\nabla\cdot\mathbf{u} = 0$

Subdomain selection: 1

Group:

☐ Select by group

☑ Active in this domain

Fluid properties and sources/sinks

Library material:

Quantity	Value/Expression	Unit	Description
ρ	1000	kg/m^3	Density
η	1e-5	Pa·s	Dynamic viscosity
F_x	0	N/m^3	Volume force, x-dir.
F_y	0	N/m^3	Volume force, y-dir.
☑ Flow in porous media (Brinkman equations)			
ε_p	0.4	1	Porosity
κ	1e-12	m^2	Permeability

Figure 5-17: Physics page when Flow in porous media is switched on.

Boundary Conditions

The application mode's boundary conditions are essentially the same as for the Navier-Stokes application modes; see "Boundary Conditions" on page 157. Differences exist for the following boundary types: Outlet, Symmetry boundary, Open boundary, and Stress. In all four cases, the viscous part of the stress is divided by the porosity to appear as

$$\frac{1}{\varepsilon}\left\{\eta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \left(\frac{2}{3}\eta - \kappa_{\mathrm{dv}}\right)(\nabla\cdot\mathbf{u})\mathbf{I}\right\} \qquad (5\text{-}67)$$

Example Model

The model "Coupled Free and Porous Media Flow" on page 57 of the *Chemical Engineering Module Model Library* shows an example how you can use the Brinkman equations to solve for the flow in a porous medium.

Reference

1. M. Le Bars and M.G. Worster, “Interfacial conditions between a pure fluid and a porous medium: implications for binary alloy solidification,” *Journal of Fluid Mechanics*, vol. 550, pp. 149–173, 2006.

The Level Set Method

Flow problems 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 possible 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 phases. With the Level Set application mode you can move the interface within any velocity field.

For two-phase flow problems modeled with the level set method, use the Level Set Two-Phase Flow application mode, see "The Level Set Method for Two-Phase Flow" on page 165.

Subdomain Equations

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 5-18 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, ϕ . In COMSOL Multiphysics, ϕ is a smooth step function that equals zero in a domain and one 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 the level set, of ϕ . Figure 5-19 shows the level set representation of the interface in Figure 5-18.

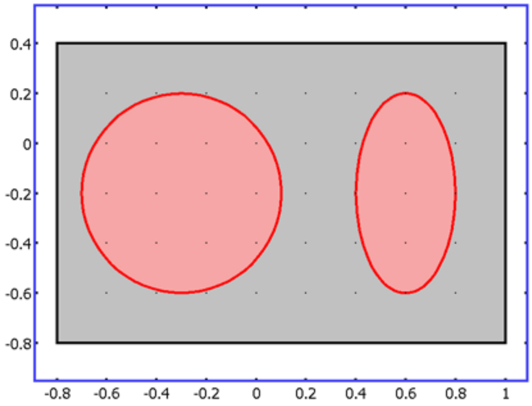


Figure 5-18: Example of two domains divided by an interface. In this case, one of the domain consists of two parts. Figure 5-19 shows the corresponding level set representation.

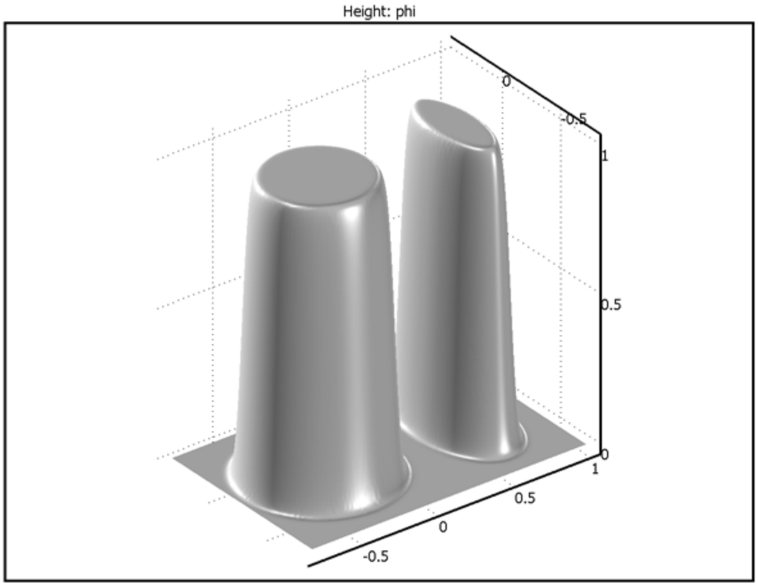


Figure 5-19: Surface plot of the level set function corresponding to Figure 5-18.

The application mode solves the following equations in order to move the interface with the velocity field, **u**:

$$\phi_t + \mathbf{u} \cdot \nabla\phi = \gamma\nabla \cdot \left(\varepsilon\nabla\phi - \phi(1-\phi)\frac{\nabla\phi}{|\nabla\phi|}\right) \tag{5-68}$$

The terms on the left hand side gives the correct motion of the interface, while the right hand side terms are necessary for numerical stability. The parameter, ε, determines the thickness of the region where ϕ goes smoothly from zero to one and should be of the same order as the size of the elements of your mesh. By default, ε is constant within each subdomain and equals the largest value of the mesh size, h, within the subdomain. The parameter γ determines the amount of reinitialization or stabilization of the level set function. It needs to be tuned for each specific problem. If γ is too small, the thickness of the interface might not remain constant, and oscillations in ϕ may appear because of numerical instabilities. On the other hand, if γ is too large, the interface moves incorrectly. A suitable value for γ is the maximum magnitude of the velocity field **u**. The example "Rigid Body Motion" on page 236 of the *Chemical Engineering Module Model Library* demonstrates how the values of ε and γ influence the solution.

CONSERVATIVE AND NONCONSERVATIVE FORM

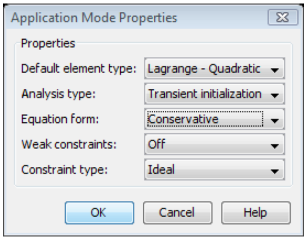
If the velocity is divergence free, that is, if

$$\nabla \cdot \mathbf{u} = 0 \tag{5-69}$$

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, you can switch to the conservative form

$$\phi_t + \nabla \cdot (\mathbf{u}\phi) = \gamma\nabla \cdot \left(\varepsilon\nabla\phi - \phi(1-\phi)\frac{\nabla\phi}{|\nabla\phi|}\right) \tag{5-70}$$

using the **Application Mode Properties** dialog box (see Figure 5-20).



Application Mode Properties

Properties

Default element type: Lagrange - Quadratic

Analysis type: Transient initialization

Equation form: Conservative

Weak constraints: Off

Constraint type: Ideal

Figure 5-20: In the Application Mode Properties dialog box, you can switch between the conservative and the nonconservative form of the level set equation.

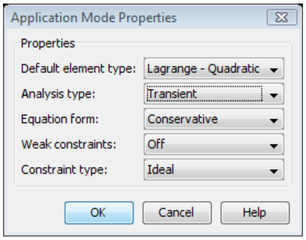
Using the conservative level set form you obtain exact numerical conservation of the integral of ϕ . Note, however, that the nonconservative form is better suited for numerical calculations and usually converges more easily. The integral of the level set function is then only approximately conserved, but this is sufficient for most applications.

INITIALIZING THE LEVEL SET FUNCTION

Before you can solve Equation 5-68 or Equation 5-70, you must initialize the level set function such that it varies smoothly from zero to one across the interface. Do so by letting ϕ_0 be zero on one side of the interface and one on the other. Then solve

$$\phi_t = \gamma\nabla\cdot\left(\varepsilon\nabla\phi - \phi(1-\phi)\frac{\nabla\phi}{|\nabla\phi|}\right) \qquad (5\text{-}71)$$

using ϕ_0 as the initial condition from $t = 0$ to $t \approx 5\varepsilon/\gamma$. The resulting ϕ is smooth across the interface and a suitable initial condition to the level set equation. The Level Set application mode automatically sets up Equation 5-71 if you select **Transient initialization** from the **Analysis type** list in the **Application Mode Properties** dialog box (Figure 5-21). You then solve the equation, store the solution, change the analysis type to **Transient**, and finally use the stored solution as the initial condition. The example "Rigid Body Motion" on page 236 of the *Chemical Engineering Module Model Library* demonstrates the procedure.



Application Mode Properties

Properties

Default element type: Lagrange - Quadratic

Analysis type: Transient

Equation form: Conservative

Weak constraints: Off

Constraint type: Ideal

Figure 5-21: By default, the Analysis type is set to Transient initialization. Note that you must switch it to Transient after you have initialized the level set function.

VARIABLES FOR GEOMETRIC PROPERTIES OF THE INTERFACE

Geometric properties of the interface are often needed. The unit normal to the interface is given by

$$\mathbf{n} = \left.\frac{\nabla\phi}{|\nabla\phi|}\right|_{\phi = 0.5} \tag{5-72}$$

The curvature is defined as

$$\kappa = -\nabla\cdot\mathbf{n}\big|_{\phi = 0.5} \tag{5-73}$$

These variables are available in the application mode.

Note: It is only possible to compute the curvature explicitly when using second-order or higher-order elements.

Boundary Conditions

Inflow/Phi At inflow boundaries you must specify a value of the level set function. Typically you set ϕ to either 0 or 1.

Outflow For outflow boundaries no boundary condition is imposed on the level set function.

Insulation/Symmetry Use this boundary condition when there should be no flow across the boundary or if the boundary is a symmetry line or plane.

Axial Symmetry Boundary condition at $r = 0$ for axisymmetric problems.

Initial interface Defines the boundary as the initial position of the interface.

Example Model

The model "Rigid Body Motion" on page 236 of the *Chemical Engineering Module Model Library* demonstrates how you can use the Level Set application mode to move an interface in a given velocity field.

Reference

1. E. Olsson and G. Kreiss, "A conservative level set method for two phase flow," *J. Comput. Phys.*, vol. 210, pp. 225–246, 2005.

The Level Set Method for Two-Phase Flow

To model the flow of two different, immiscible fluids, where the exact position of the interface is of interest, you can use the Level Set Two-Phase Flow application mode. The application mode tracks the fluid-fluid interface using a level set method. It accounts for differences in the two fluids' densities and viscosities and includes the effect of surface tension and gravity.

Subdomain Settings

For incompressible fluids, the application mode uses the equations

$$\rho\frac{\partial\mathbf{u}}{\partial t}-\nabla\cdot\eta(\nabla\mathbf{u}+(\nabla\mathbf{u})^T)+\rho\mathbf{u}\cdot\nabla\mathbf{u}+\nabla p = \mathbf{F}_{\mathrm{st}}+\mathbf{F}_{\mathrm{g}}+\mathbf{F} \tag{5-74}$$

$$\nabla\cdot\mathbf{u} = 0 \tag{5-75}$$

$$\frac{\partial\phi}{\partial t}+\nabla\phi\cdot\mathbf{u} = \gamma\nabla\cdot\left(\varepsilon\nabla\phi-\phi(1-\phi)\frac{\nabla\phi}{|\nabla\phi|}\right) \tag{5-76}$$

The density is a function of the level set function according to

$$\rho = \rho_1+(\rho_2-\rho_1)\phi$$

and the dynamic viscosity is

$$\eta = \eta_1+(\eta_2-\eta_1)\phi$$

where ρ_1 and ρ_2 are the constant densities of Fluid 1 and Fluid 2, respectively, and η_1 and η_2 are the dynamic viscosities 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 $\phi>0.5$.

The surface tension force acting at the interface between the two fluids is

$$\mathbf{F}_{\mathrm{st}} = \sigma\kappa\delta\mathbf{n}$$

where σ is the surface tensions coefficient (N/m), κ is the curvature, and $\mathbf{n}$ is the unit normal to the interface, as defined in Equation 5-72 and Equation 5-73 in the section

“Variables For Geometric Properties of the Interface” on page 163. δ (1/m) is a Dirac delta function concentrated to the interface. κ depends on second derivatives of the level set function ϕ . This can lead to poor accuracy of the surface tension force. Therefore, the application mode uses the alternative formulation

$$\mathbf{F}_{\mathrm{st}} = \nabla \cdot (\sigma(\mathbf{I} - (\mathbf{n}\mathbf{n}^T))\delta)$$

In the weak formulation of the momentum equations, it is possible to move the divergence operator, using integration by parts, to the test functions for the velocity components.

The δ-function is approximated by a smooth function according to

$$\delta = 6|\nabla\phi||\phi(1-\phi)|$$

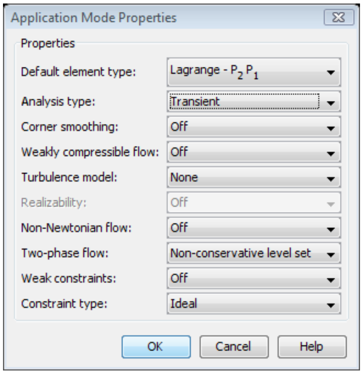
In addition to the surface tension force, there is also a term that accounts for gravity. The gravity force is

$$\mathbf{F}_{\mathrm{g}} = \rho\mathbf{g}$$

where **g** is the gravity vector.

Because the velocity field is divergence free, you can use either the conservative or the nonconservative form of the level set equation (see the section “Conservative and Nonconservative form” on page 161). The conservative form perfectly conserves the mass of each fluid, but the computational time is in general longer. Note that when you use the conservative form, the default element type for the velocity and pressure automatically changes to P_2 P_{-1} in 2D and P_2+ P_{-1} in 3D, since they result in a better approximation of the continuity equation. For details on the different element types used for fluid flow modeling, see “Special Element Types For Fluid Flow” on page 41 of the *Chemical Engineering Module Reference Guide*.

You can modify the model in a number of different ways from the **Application Mode Properties** dialog box. Here you can switch between the conservative and the nonconservative form. You can also change the fluid flow model to weakly compressible flow, turbulent flow, swirl flow, and so on, and switch from two-phase flow to single-phase flow (see Figure 5-22).



Application Mode Properties

Properties

Default element type:	Lagrange - P_2 P_1
Analysis type:	Transient
Corner smoothing:	Off
Weakly compressible flow:	Off
Turbulence model:	None
Realizability:	Off
Non-Newtonian flow:	Off
Two-phase flow:	Non-conservative level set
Weak constraints:	Off
Constraint type:	Ideal

Figure 5-22: In the Application Mode Properties dialog box, you can switch on or off a turbulence model or turn on the weakly compressible flow model for the fluid flow. Here, you also switch between the Analysis types Transient initialization for the initialization of the level set function, and Transient for the dynamic fluid-flow modeling.

INITIALIZING THE LEVEL SET FUNCTION

Before you can solve Equation 5-74–Equation 5-76, you must initialize the level set function such that it varies smoothly from zero to one across the interface. Do so by specifying which fluid each subdomain is initially filled with. Then solve

$$\phi_t = \gamma\nabla\cdot\left(\varepsilon\nabla\phi - \phi(1-\phi)\frac{\nabla\phi}{|\nabla\phi|}\right) \tag{5-77}$$

using ϕ_0 as the initial condition from $t = 0$ to $t \approx 5\varepsilon/\gamma$. The resulting ϕ is smooth across the interface and a suitable initial condition to the level set equation. The Level Set Two-Phase Flow application mode automatically sets up Equation 5-77 if you select **Transient initialization** from the **Analysis type** list in the **Application Mode Properties** dialog box (see Figure 5-22). You then solve the equation, store the solution, change the analysis type to **Transient**, and finally use the stored solution as the initial condition.

Boundary Conditions

All the boundary conditions available for single-phase flow are also available for the Level Set Two-Phase Flow application mode. The boundary condition used on the level set function ϕ depends on whether the boundary is an inlet, an outlet, or a wall or symmetry boundary. For inlets, you must specify the value of the level set function. Typically, you set it to either zero or one, depending on which fluid enters the boundary. At outlets, no boundary conditions on the level set function is imposed. For walls and symmetry lines, the application mode sets the flux of the level set function to zero. The boundary types Open boundary and Stress are not available for two-phase flows.

There are also two boundary conditions that are only available for the Level Set Two-Phase Flow application mode: the Initial fluid interface and the Wetted wall boundary conditions.

THE INITIAL FLUID INTERFACE CONDITION

To specify the initial position of the interface, use the Initial fluid interface boundary condition on interior boundaries. During the initialization step, the boundary condition sets the level set function to 0.5. For the transient simulation of the fluid flow, the boundary is treated as an interior boundary.

THE WETTED WALL CONDITION

The Wetted wall boundary condition is suitable for walls in contact with the fluid interface. It enforces the slip condition

$$\mathbf{u}\cdot\mathbf{n}_{\mathrm{wall}} = 0$$

and adds a frictional force of the form

$$\mathbf{F}_{\mathrm{fr}} = -\frac{\eta}{\beta}\mathbf{u}$$

where β is the slip length. For numerical calculations, it is suitable to set $\beta = h$, 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 the distance β outside the wall, as Figure 5-23 illustrates.

Finally, the boundary condition adds the following weak boundary term:

$$\int_{\partial\Omega} \mathrm{test}(\mathbf{u})\cdot[\sigma(\mathbf{n}_{\mathrm{wall}}-(\mathbf{n}\cos\theta))\delta]dS$$

The boundary term is a result of the partial integration of the surface tension force in the Navier-Stokes equations. It allows you to specify the contact angle θ, that is, the angle between the fluid interface and the wall. Figure 5-73 illustrates how the contact angle is defined.

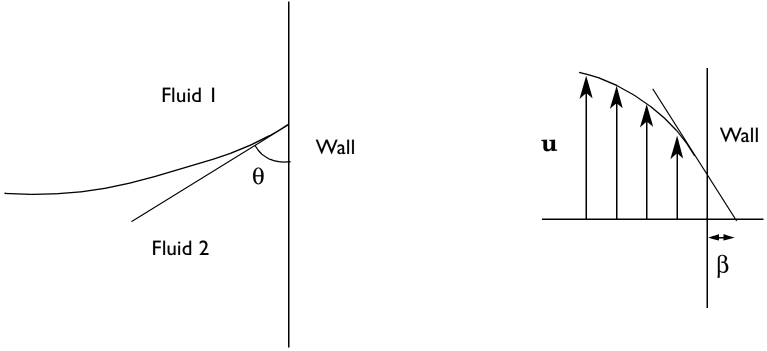


Figure 5-23: Definition of the contact angle θ *at interface/wall contact points (left) and illustration of the slip length* β *(right).*

If you use the Wetted wall boundary condition, the interface can move along the wall. For applications where the interface is fixed on the wall, the No slip condition is suitable.

Example Model

For an example model that uses the Level Set Two-Phase Flow application mode, see "Oscillating Liquid Cobalt Droplet" on page 75 in the *Chemical Engineering Model Library.*

Numerical Stabilization

In most cases the application mode works best if no stabilization is used. However, for convection-dominated flow some stabilization might be necessary. If you encounter convergence problems, make a surface plot of the Cell Reynolds number. If it is larger than 2, switch on streamline diffusion of the anisotropic diffusion type in the **Artificial Diffusion** dialog box.

The choice of the level set parameter, γ, is also important for stability. The maximum speed of the flow is usually a suitable value.

Whenever the effect of surface tension is negligible, set the surface tension coefficient, σ, to 0 for increased stability.

Reference

1. E. Olsson and G. Kreiss, "A conservative level set method for two phase flow," *J. Comput. Phys.*, vol. 210, pp. 225–246, 2005.

Bubbly Flow Model

The Bubbly Flow application mode is suitable for macroscopic modeling of mixtures of liquids and gas bubbles. The application mode solves for the averaged volume fraction occupied by each of the two phases, rather than tracking each bubble in detail. It also assumes two different velocity fields, one for each phase.

Subdomain Equations

The two-fluid Euler-Euler model is a general, macroscopic model for two-phase fluid flow. It treats the two phases as interpenetrating media, tracking the averaged concentration of the phases. One velocity field is associated with each phase, and a momentum balance and a continuity equation describe the dynamics of each of the phases. The bubbly flow model is a simplification of the two-fluid model, relying on the following assumptions:

- The gas density is negligible compared to the liquid density.
- The motion of the gas bubbles relative to the liquid is determined by a balance between viscous drag and pressure forces.
- The two phases share the same pressure field.

Based on these assumptions, the sum of the momentum equations for the two phases gives a momentum equation for the liquid velocity, a continuity equation, and a transport equation for the volume fraction of the gas phase. The momentum equation is

$$\begin{aligned}\phi_l\rho_l\frac{\partial\mathbf{u}_l}{\partial t}+\phi_l\rho_l\mathbf{u}_l\cdot\nabla\mathbf{u}_l &= \\ = -\nabla p+\nabla\cdot\left[\phi_l(\eta_l+\eta_T)\left(\nabla\mathbf{u}_l+\nabla\mathbf{u}_l{}^T-\frac{2}{3}(\nabla\cdot\mathbf{u}_l)\mathbf{I}\right)\right]&+\phi_l\rho_l\mathbf{g}+\mathbf{F}\end{aligned} \tag{5-78}$$

Here, $\mathbf{u}$ denotes velocity (m/s), p pressure (Pa), ϕ phase volume fraction ($\mathrm{m^3/m^3}$), ρ density ($\mathrm{kg/m^3}$), $\mathbf{g}$ the gravity vector ($\mathrm{m/s^2}$), $\mathbf{F}$ any additional volume force ($\mathrm{N/m^3}$), η_l dynamic viscosity of the liquid (Pa·s), and η_T turbulent viscosity (Pa·s). The subscripts l and g denote quantities related to the liquid phase and the gas phase, respectively.

The continuity equation is

$$\frac{\partial}{\partial t}(\rho_l\phi_l + \rho_g\phi_g) + \nabla\cdot(\rho_l\phi_l\mathbf{u}_l + \rho_g\phi_g\mathbf{u}_g) = 0 \qquad (5\text{-}79)$$

and the transport of the volume fraction of gas is given by

$$\frac{\partial\rho_g\phi_g}{\partial t} + \nabla\cdot(\phi_g\rho_g\mathbf{u}_g) = -m_{\mathrm{gl}} \qquad (5\text{-}80)$$

where m_{gl} is the mass transfer rate from gas to liquid (kg/m^3/s).

For low gas volume fractions ($\phi_{\mathrm{g}} < 0.1$) it is in general valid to replace the continuity equation, Equation 5-79, by

$$\nabla\cdot\mathbf{u}_l = 0 \qquad (5\text{-}81)$$

By default, the application mode uses Equation 5-81. To switch to Equation 5-79, set the application mode property **Low gas concentration** to **Off** (see Figure 5-26).

The application mode solves for $\mathbf{u}_l$, p, and $\tilde{\rho}_g = \rho_g\phi_g$, the effective gas density. The gas velocity is calculated according to

$$\mathbf{u}_g = \mathbf{u}_l + \mathbf{u}_{\mathrm{slip}} + \mathbf{u}_{\mathrm{drift}}$$

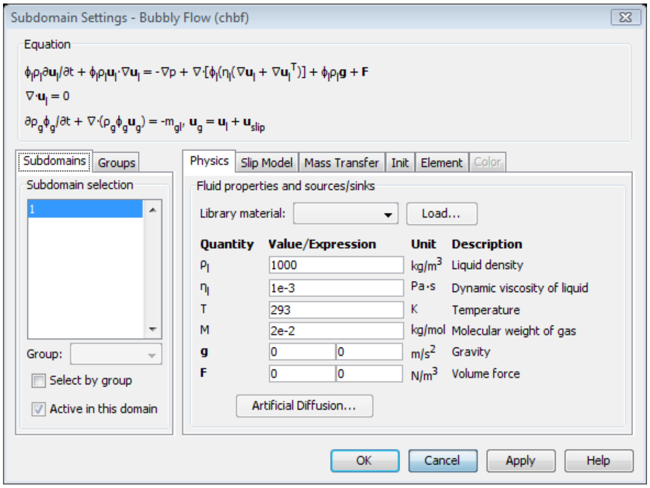
where $\mathbf{u}_{\mathrm{slip}}$ is the relative velocity between the phases and $\mathbf{u}_{\mathrm{drift}}$ is an additional contribution when a turbulence model is used. The application mode calculates the gas density from the ideal gas law:

$$\rho_g = \frac{(p + p_{\mathrm{ref}})M}{RT}$$

where M is the molecular weight of the gas (kg/mol), R is the ideal gas constant (8.31 J/(mol·K)), p_{ref} a reference pressure (Pa) and T is temperature (K). p_{ref} is a scalar variable, which by default is $1\cdot10^5$ Pa. The liquid volume fraction is calculated from

$$\phi_l = 1 - \phi_g$$

In the **Subdomain Settings** dialog box, the equations used by the application mode are displayed. Here you also specify the physical properties of the fluid (see Figure 5-24).



Subdomain Settings - Bubbly Flow (chbf)

Equation

$\phi_l\rho_l\partial\mathbf{u}_l/\partial t + \phi_l\rho_l\mathbf{u}_l\cdot\nabla\mathbf{u}_l = -\nabla p + \nabla\cdot[\phi_l(\eta_l(\nabla\mathbf{u}_l + \nabla\mathbf{u}_l^T))] + \phi_l\rho_l\mathbf{g} + \mathbf{F}$

$\nabla\cdot\mathbf{u}_l = 0$

$\partial\rho_g\phi_g/\partial t + \nabla\cdot(\rho_g\phi_g\mathbf{u}_g) = -m_{gl},\ \mathbf{u}_g = \mathbf{u}_l + \mathbf{u}_{slip}$

Subdomain selection: 1

Group:

☐ Select by group

☑ Active in this domain

Fluid properties and sources/sinks

Library material:

Quantity	Value/Expression	Unit	Description
ρ_l	1000	kg/m^3	Liquid density
η_l	1e-3	Pa·s	Dynamic viscosity of liquid
T	293	K	Temperature
M	2e-2	kg/mol	Molecular weight of gas
g	0 0	m/s^2	Gravity
F	0 0	N/m^3	Volume force

Figure 5-24: The subdomain equations appear in the Subdomain Settings dialog box, where you also specify the physical properties of the fluids.

SLIP VELOCITY

Several different models for the slip velocity $\mathbf{u}_{\mathrm{slip}}$ are available in the application mode. The simplest model assumes that the velocity of the two phases are equal, that is, $\mathbf{u}_{\mathrm{slip}} = 0$. The other models are based on the assumption that the pressure forces on a bubble are balanced by the viscous drag:

$$\frac{3}{4}\frac{C_d}{d_b}\rho_l|\mathbf{u}_{\mathrm{slip}}|\mathbf{u}_{\mathrm{slip}} = -\nabla p$$

Here d_b denotes the bubble diameter (m) and C_d is the viscous drag coefficient (dimensionless). For the drag coefficient, you can either use one of the predefined models or specify an arbitrary expression.

For gas bubbles with a diameter larger than 2 mm, you can use the following predefined model for the drag coefficient (Ref. 1)

$$C_d = \frac{0.622}{\frac{1}{\mathrm{Eö}} + 0.235}$$

where Eö is the Eötvös number

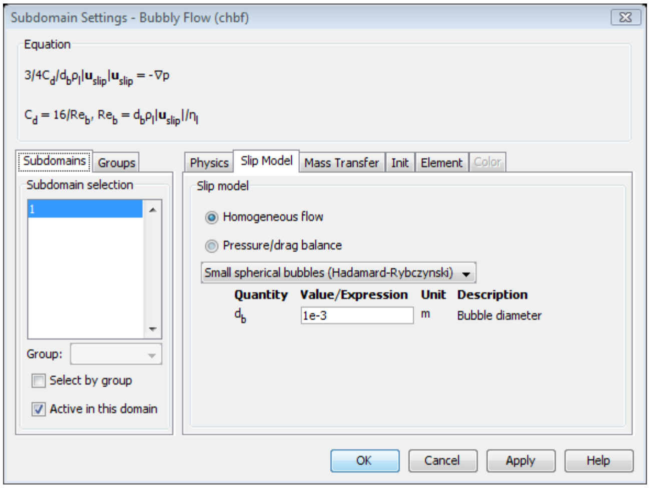
$$\text{Eö} = \frac{g\rho_l d_b^2}{\sigma}$$

Here, g is the magnitude of the gravity vector and σ the surface tension coefficient.

For bubbles with a diameter smaller than 2 mm, you can choose the Hadamard-Rybczynski drag law for spherical gas bubbles in liquid (Ref. 3)

$$C_d = \frac{16}{\text{Re}_b}, \text{Re}_b = \frac{d_b\rho_l|\mathbf{u}_{\text{slip}}|}{\eta_l}.$$

Figure 5-25 illustrates how you specify the slip model in **Subdomain Settings** dialog box.



Subdomain Settings - Bubbly Flow (chbf)

Equation

$3/4C_d/d_b\rho_l|\mathbf{u}_{\text{slip}}|\mathbf{u}_{\text{slip}} = -\nabla p$

$C_d = 16/\text{Re}_b$, $\text{Re}_b = d_b\rho_l|\mathbf{u}_{\text{slip}}|/\eta_l$

Subdomain selection

1

Group:

□ Select by group

☑ Active in this domain

Slip model

● Homogeneous flow

○ Pressure/drag balance

Small spherical bubbles (Hadamard-Rybczynski)

Quantity	Value/Expression	Unit	Description
d_b	1e-3	m	Bubble diameter

Figure 5-25: Graphical user interface for the specification of the slip model.

TURBULENCE MODELING

For most bubbly flow applications the flow field becomes turbulent. In that case, you can use a turbulence model and solve for the averaged velocity field. In the Bubbly Flow application mode you can use the k-ε turbulence model. It is available with or without the inclusion of bubble-induced turbulence, that is, extra production of turbulence due to the relative motion between the gas bubbles and the liquid.

The k-ε model solves two extra transport equations for two additional variables: the turbulent kinetic energy, k (m^2/s^2) and the dissipation rate of turbulent energy, ε (m/s^3). The turbulent viscosity is then

$$\eta_T = \rho_l C_\mu \frac{k^2}{\varepsilon} \tag{5-82}$$

where C_μ is a model constant.

The transport equation for the turbulent kinetic energy, k, is

$$\rho_l \frac{\partial k}{\partial t} - \nabla \cdot \left[\left(\eta + \frac{\eta_T}{\sigma_k} \right) \nabla k \right] + \rho_l \mathbf{u}_l \cdot \nabla k = \frac{1}{2} \eta_T (\nabla \mathbf{u}_l + (\nabla \mathbf{u}_l)^T)^2 - \rho_l \varepsilon + S_k \tag{5-83}$$

and the evolution of the turbulent energy's dissipation rate ε:

$$\begin{aligned} \rho_l \frac{\partial \varepsilon}{\partial t} - \nabla \cdot \left[\left(\eta + \frac{\eta_T}{\sigma_\varepsilon} \right) \nabla \varepsilon \right] + \rho_l \mathbf{u}_l \cdot \nabla \varepsilon &= \\ = \frac{1}{2} C_{\varepsilon 1} \frac{\varepsilon}{k} \eta_T (\nabla \mathbf{u}_l + (\nabla \mathbf{u}_l)^T)^2 - \rho_l C_{\varepsilon 2} \frac{\varepsilon^2}{k} + \frac{\varepsilon}{k} C_\varepsilon S_k \end{aligned} \tag{5-84}$$

In all the previous equations, the velocity, $\mathbf{u}_l$, is the liquid phase averaged velocity field.

The constants used in the standard k-ε turbulence model are

CONSTANT	VALUE
C_μ	0.09
$C_{\varepsilon 1}$	1.44
$C_{\varepsilon 2}$	1.92
σ_k	1.0
σ_ε	1.3

The term S_k is related to the bubble-induced turbulence and is given by

$$S_k = -C_k \phi_g \nabla p \cdot \mathbf{u}_{\mathrm{slip}} .$$

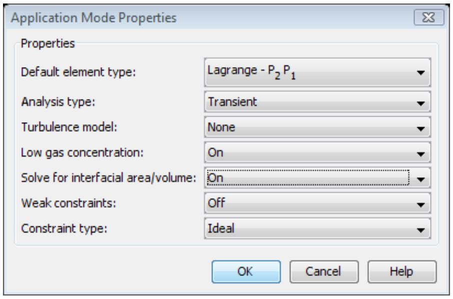
Suitable values for the model parameters C_k and C_ε are not as well established as the parameters for single-phase flow. In the literature, values within the ranges $0.01 < C_k < 1$ and $1 < C_\varepsilon < 1.92$ have been suggested (Ref. 1). The turbulent viscosity is taken into account in the momentum equations and by adding a drift velocity to the gas velocity:

$$\mathbf{u}_{\mathrm{drift}} = -\frac{\eta_{\mathrm{T}}}{\rho_{\mathrm{l}}} \frac{\nabla \phi_g}{\phi_g} .$$

MASS TRANSFER AND INTERFACIAL AREA

It is possible to account for mass transfer between the two phases by specifying an expression for the mass transfer rate from gas to liquid m_{gl} (kg/m^3/s).

The mass transfer rate typically depends on the interfacial area between the two phases. An example is when gas dissolves into the liquid. In order to determine the interfacial area, it is necessary to solve for the bubble number density, that is, the number of bubbles per volume, in addition to the phase volume fraction. The application mode automatically adds a transport equation for the bubble density if you switch the application mode property **Solve for interfacial area/volume** to **On** (Figure 5-26).


Application Mode Properties

Properties

Default element type: Lagrange - P_2 P_1

Analysis type: Transient

Turbulence model: None

Low gas concentration: On

Solve for interfacial area/volume: On

Weak constraints: Off

Constraint type: Ideal

Figure 5-26: The Application Mode Properties dialog box.

The application mode assumes that the gas bubbles can expand or shrink but not completely vanish, merge, or split. The conservation of the number density n (1/m^3) then gives

$$\frac{\partial n}{\partial t} + \nabla \cdot (n\mathbf{u}_g) = 0$$

The number density and the volume fraction of gas gives the interfacial area per unit volume (m^2/m^3):

$$a = (4n\pi)^{1/3}(3\phi_g)^{2/3}$$

Two-Film Theory

For the dissolution of gas into a liquid, the application mode can automatically calculate the mass transfer rate according to the two-film theory, given the concentration of the dissolved gas.

If the molar flux per interfacial area is N, the mass transport from gas to liquid is given by

$$m_{gl} = NMa$$

where N (mol/(s·m^2)) is determined by

$$N = k(c^* - c)$$

Here k (m/s) is the mass transfer coefficient and c is the concentration of gas dissolved in liquid (mol/m^3). Henry's law gives the equilibrium concentration of gas dissolved in liquid c^* (mol/m^3):

$$c^* = \frac{p + p_{\mathrm{ref}}}{H}$$

where H is Henry's constant (Pa·m^3/mol).

In order to use the two-film theory model, you need the concentration of the dissolved gas. The dissolved gas transport can then be modeled as

$$\frac{\partial c}{\partial t} + \nabla \cdot (c\mathbf{u}_l) = \nabla \cdot (D\nabla c) + Na$$

which you obtain by adding a Convection and Diffusion application mode.

Boundary Conditions

LIQUID PHASE

For the liquid phase you first specify the boundary type and then choose an appropriate boundary condition. The boundary types are Wall, Inlet, Outlet, and Symmetry boundary. The boundary conditions available for each type are described in the following sections.

In all expressions, **n** denotes the unit vector normal to the boundary, pointing outward from the computational domain.

Boundary type: Wall

No slip Fixes the velocity to zero.

$$\mathbf{u}_l = 0$$

Slip Sets the velocity component normal to the wall to zero.

$$\mathbf{u}_l \cdot \mathbf{n} = 0$$

Wall function Uses empirical models to model the thin laminar boundary layer close to the walls. Conditions on the viscous stress are determined from the law of the wall according to

$$\mathbf{K} = -[\rho_l \phi_l C_\mu^{0.25} k^{0.5} / (\log(\delta_w^+)/\kappa + 5.5)]\mathbf{u}_l$$

where $\mathbf{K}$ is the viscous stress:

$$\mathbf{K} = \phi_l(\eta_l + \eta_T)\Big(\nabla\mathbf{u}_l + \nabla\mathbf{u}_l^T - \frac{2}{3}(\nabla \cdot \mathbf{u}_l)\mathbf{I}\Big)\mathbf{n}$$

The normal component of the velocity is set to zero.

$$\mathbf{u}_l \cdot \mathbf{n} = 0$$

For more information on the wall functions, see the documentation on the k-ε turbulence model for single-phase flows.

Boundary type: Inlet

Velocity Allows you to specify the velocity at the boundary.

$$\mathbf{u}_l = \mathbf{u}_0$$

Pressure Allows you to specify the pressure at the boundary and sets the viscous stress to zero.

$$p = p_0, \ \phi_l(\eta_l + \eta_T)\Big(\nabla\mathbf{u}_l + \nabla\mathbf{u}_l^T - \frac{2}{3}(\nabla \cdot \mathbf{u}_l)\mathbf{I}\Big)\mathbf{n} = 0$$

Boundary type: Outlet

Velocity Allows you to specify the velocity at the boundary.

$$\mathbf{u}_l = \mathbf{u}_0$$

Pressure Allows you to specify the pressure at the boundary and sets the viscous stress to zero.

$$p = p_0,\ \phi_l(\eta_l+\eta_T)\left(\nabla\mathbf{u}_l+\nabla\mathbf{u}_l^T-\frac{2}{3}(\nabla\cdot\mathbf{u}_l)\mathbf{I}\right)\mathbf{n} = 0$$

No viscous stress Sets the viscous stress to zero at the boundary, that is,

$$\phi_l(\eta_l+\eta_T)\left(\nabla\mathbf{u}_l+\nabla\mathbf{u}_l^T-\frac{2}{3}(\nabla\cdot\mathbf{u}_l)\mathbf{I}\right)\mathbf{n} = -\mathrm{f}_0\mathbf{n}$$

Normal stress Sets the total stress equal to a stress vector of magnitude, f_0, oriented in the negative normal direction, that is,

$$\phi_l(\eta_l+\eta_T)\left(\nabla\mathbf{u}_l+\nabla\mathbf{u}_l^T-\frac{2}{3}(\nabla\cdot\mathbf{u}_l)\mathbf{I}\right)\mathbf{n}-p\mathbf{n} = -\mathrm{f}_0\mathbf{n}$$

Boundary type: Symmetry

Symmetry Sets the normal component of the velocity to zero.

$$\mathbf{u}_l\cdot\mathbf{n} = 0$$

Axisymmetry Boundary condition at the axial symmetry line (axisymmetric models only).

GAS PHASE

For the gas phase, the following boundary conditions are available.

Gas concentration Allows you to specify the effective gas density.

$$\tilde{\rho}_g = \tilde{\rho}_g^0$$

Gas flux Allows you to specify the gas flux through the boundary.

$$-\mathbf{n}\cdot(\tilde{\rho}_g\mathbf{u}_g) = N_\rho$$

Gas outlet For boundaries where the gas phase flows outwards with the gas velocity, $\mathbf{u}_g$. Corresponds to no condition at the boundary.

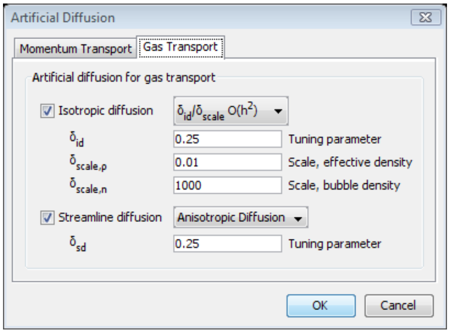
Insulation/Symmetry Sets the gas flux through the boundary to zero.

$$\mathbf{n}\cdot(\tilde{\rho}_g\mathbf{u}_g) = 0$$

Axial symmetry For boundaries at the axial symmetry line (axisymmetric models only).

Numerical Stabilization

For the momentum equations as well as the gas transport equation, you can add isotropic diffusion, streamline diffusion, or both. By default, the momentum transport equation is stabilized using streamline diffusion. Streamline diffusion and isotropic diffusion ($O(h^2)$) are, by default, added to the gas transport equation. Note that you must specify a suitable scale for the effective gas density and the bubble number density (see Figure 5-27). Appropriate scales are the maximum values of the gas volume fraction and the maximum bubble number density.



Artificial diffusion for gas transport

☑ Isotropic diffusion $\delta_{id}/\delta_{scale}$ O(h^2)

δ_{id} 0.25 Tuning parameter

$\delta_{scale,\rho}$ 0.01 Scale, effective density

$\delta_{scale,n}$ 1000 Scale, bubble density

☑ Streamline diffusion Anisotropic Diffusion

δ_{sd} 0.25 Tuning parameter

Figure 5-27: Artificial Diffusion settings for the gas transport.

See “Numerical Stability—Artificial Diffusion” on page 121 for more details on the different artificial diffusion techniques.

Example Model

The model “Bubble Column Reactor” on page 279 of the *Chemical Engineering Module Model Library* shows how to model a bubbly flow in a reactor using the Bubbly Flow application mode.

References

1. A. Sokolichin, G. Eigenberger, and A. Lapin, “Simulations of Buoyancy Driven Bubbly Flow: Established Simplifications and Open Questions,” *AIChE Journal*, vol. 50, no. 1, pp. 24–49, 2004.

2. D. Kuzmin and S. Turek, *Efficient Numerical Techniques for Flow simulation in Bubble Column Reactors,* Institute of Applied Mathematics, University of Dortmund.

3. C. Crowe, M. Sommerfeld, and Y. Tsuji, *Multiphase Flows with Droplets and Particles,* CRC Press, 1998.

The Mixture Model

The mixture model is a macroscopic two-phase flow model, in many ways similar to the Bubbly Flow model. It tracks the averaged phase concentration, or volume fraction, and solves for one velocity field for each phase. It is suitable for mixtures consisting of solid particles or liquid droplets immersed in a liquid.

Subdomain Equations

Just as the Bubbly Flow application mode, the Mixture Model application mode is based on the two fluid Euler-Euler model. The two phases consist of one dispersed phase and one continuous phase. The mixture model is valid if the continuous phase is a liquid, and the dispersed phase consists of either solid particles, liquid droplets, or gas bubbles. For gas bubbles in a liquid, however, the Bubbly Flow model is preferable. The mixture model relies on the following assumptions

- The density of each phase is approximately constant.
- Both phases share the same pressure field.
- The relative velocity between the two phases is essentially determined assuming a balance between pressure, gravity, and viscous drag.

The momentum equation for the mixture is

$$\rho\mathbf{u}_t + \rho(\mathbf{u}\cdot\nabla)\mathbf{u} = -\nabla p - \nabla\cdot(\rho c_d(1-c_d)\mathbf{u}_{\text{slip}}\mathbf{u}_{\text{slip}}) + \nabla\cdot\tau_{\text{Gm}} + \rho\mathbf{g} + \mathbf{F} \quad (5\text{-}85)$$

where $\mathbf{u}$ denotes velocity (m/s), ρ density (kg/m^3), p pressure (Pa), c_d mass fraction of the dispersed phase (kg/kg), $\mathbf{u}_{\text{slip}}$ the relative velocity between the two phases (m/s), τ_{Gm} the sum of viscous and turbulent stress (kg/(m·s^2)), $\mathbf{g}$ the gravity vector (m/s^2), and $\mathbf{F}$ additional volume forces (N/m^3). The velocity $\mathbf{u}$ used here is the mixture velocity (m/s), defined as

$$\mathbf{u} = \frac{\phi_c\rho_c\mathbf{u}_c + \phi_d\rho_d\mathbf{u}_d}{\rho}$$

where ϕ_c and ϕ_d denote the volume fractions of the continuous phase and the dispersed phase (m^3/m^3), respectively, $\mathbf{u}_c$ the continuous phase velocity (m/s), $\mathbf{u}_d$ the dispersed phase velocity (m/s), ρ_c the continuous phase density (kg/m^3), ρ_d the dispersed phase density (kg/m^3), and ρ the mixture density (kg/m^3). The relation between the velocities of the two phases is defined by

$$\mathbf{u}_d - \mathbf{u}_c = \mathbf{u}_{cd} = \mathbf{u}_{\mathrm{slip}} - \frac{D_{md}}{(1-c_d)\phi_d}\nabla\phi_d \qquad (5\text{-}86)$$

Here, $\mathbf{u}_{\mathrm{slip}}$ denotes the relative velocity between the two phases (m/s). For different available models for the slip velocity, see "Slip Velocity" on page 185. D_{md} is a turbulent dispersion coefficient (m^2/s) (see "Turbulence Modeling" on page 188), accounting for extra diffusion due to turbulent eddies. If you do not use any turbulence model, D_{md} is zero.

The mixture density is given by

$$\rho = \phi_c\rho_c + \phi_d\rho_d$$

where ρ_c and ρ_d are the densities of each of the two phases (kg/m^3). The mass fraction of the dispersed phase c_d is given by

$$c_d = \frac{\phi_d\rho_d}{\rho}$$

The sum of viscous and turbulent stress is

$$\tau_{Gm} = (\eta + \eta_T)[\nabla\mathbf{u} + \nabla\mathbf{u}^T] - \frac{2}{3}\rho k\mathbf{I}$$

where η is the mixture viscosity (Pa·s) (see the section "Mixture Viscosity" on page 187), η_T the turbulent viscosity (Pa·s), and k the turbulent kinetic energy (m^2/s^2). If no turbulence model is used, η_T and k equal zero.

The transport equation for ϕ_d , the dispersed phase volume fraction, is

$$\frac{\partial}{\partial t}(\phi_d\rho_d) + \nabla\cdot(\phi_d\rho_d\mathbf{u}_d) = -m_{dc} \qquad (5\text{-}87)$$

where m_{dc} is the mass transfer rate from dispersed to continuous phase (kg/(m^3·s)) and $\mathbf{u}_d$ is the dispersed phase velocity (m/s) according to Equation 5-86.

The continuous phase volume fraction ϕ_c is

$$\phi_c = 1 - \phi_d$$

The continuity equation for the mixture is

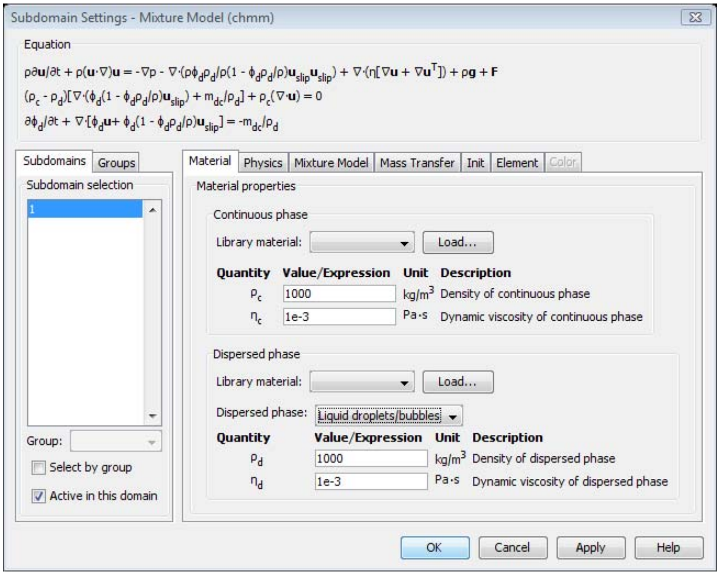
$$\rho_t + \nabla\cdot(\rho\mathbf{u}) = 0 \qquad (5\text{-}88)$$

The application mode assumes that the densities of each phase, ρ_c and ρ_d, are constant, and therefore uses the following alternative form of the continuity equation of the mixture

$$(\rho_c - \rho_d)\left[\nabla \cdot (\phi_d(1 - c_d)\mathbf{u}_{\text{slip}} - D_{md}\nabla\phi_d) + \frac{m_{dc}}{\rho_d}\right] + \rho_c(\nabla \cdot \mathbf{u}) \;=\; 0 \qquad (5\text{-}89)$$

You can derive Equation 5-89 from Equation 5-87 and Equation 5-88.

The application mode automatically sets up all the equations. In the **Subdomain Settings** dialog box, you specify the physical parameters as well as which model to use for the slip velocity and the mixture viscosity (see Figure 5-28).



Subdomain Settings - Mixture Model (chmm)

Equation

$\rho\partial\mathbf{u}/\partial t + \rho(\mathbf{u}\cdot\nabla)\mathbf{u} = -\nabla p - \nabla\cdot(\rho\phi_d\rho_d/\rho(1 - \phi_d\rho_d/\rho)\mathbf{u}_{slip}\mathbf{u}_{slip}) + \nabla\cdot(\eta[\nabla\mathbf{u} + \nabla\mathbf{u}^T]) + \rho\mathbf{g} + \mathbf{F}$

$(\rho_c - \rho_d)[\nabla\cdot(\phi_d(1 - \phi_d\rho_d/\rho)\mathbf{u}_{slip}) + m_{dc}/\rho_d] + \rho_c(\nabla\cdot\mathbf{u}) = 0$

$\partial\phi_d/\partial t + \nabla\cdot[\phi_d\mathbf{u} + \phi_d(1 - \phi_d\rho_d/\rho)\mathbf{u}_{slip}] = -m_{dc}/\rho_d$

Subdomain selection: 1

Group:

☐ Select by group

☑ Active in this domain

Material properties

Continuous phase

Library material:

Quantity	Value/Expression	Unit	Description
ρ_c	1000	kg/m^3	Density of continuous phase
η_c	1e-3	Pa·s	Dynamic viscosity of continuous phase

Dispersed phase

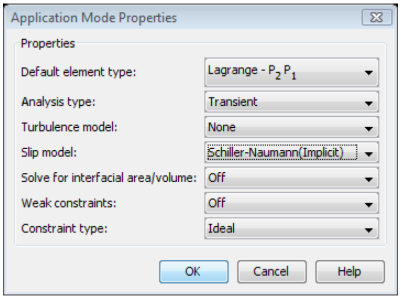
Library material:

Dispersed phase: Liquid droplets/bubbles

Quantity	Value/Expression	Unit	Description
ρ_d	1000	kg/m^3	Density of dispersed phase
η_d	1e-3	Pa·s	Dynamic viscosity of dispersed phase

Figure 5-28: The Subdomain Settings dialog box for the Mixture Model application mode.

In the **Application Mode Properties** dialog box (see Figure 5-29) you specify the form of the equations. Here you can, for instance, choose between a stationary or transient analysis and specify a turbulence model.



Application Mode Properties

Properties

Default element type: Lagrange - P_2 P_1

Analysis type: Transient

Turbulence model: None

Slip model: Schiller-Naumann(Implicit)

Solve for interfacial area/volume: Off

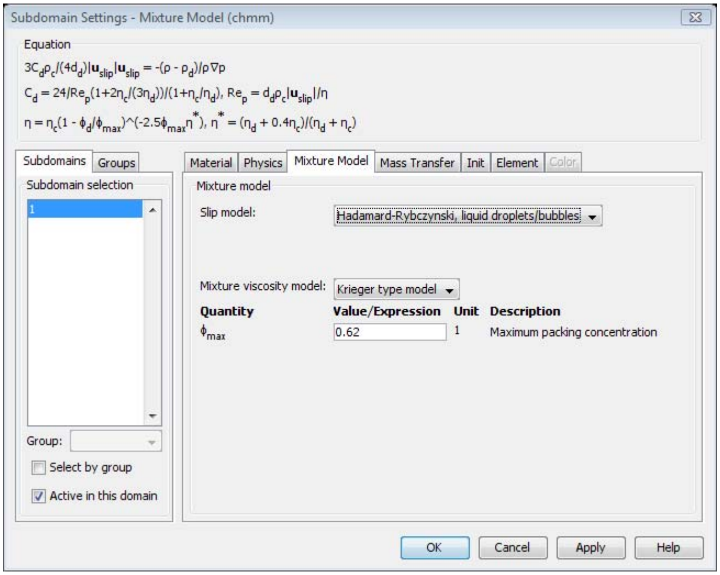
Weak constraints: Off

Constraint type: Ideal

Figure 5-29: The Application Mode Properties dialog box.

SLIP VELOCITY

Several different models for the slip velocity, $\mathbf{u}_{\mathrm{slip}}$ (m/s), are available in the application mode. You can choose between **Homogeneous flow**, **Hadamard-Rybczynski**, **Schiller-Naumann**, and **User defined** from the **Slip model** list.



Subdomain Settings - Mixture Model (chmm)

Equation

$3C_d\rho_c/(4d_d)|\mathbf{u}_{slip}|\mathbf{u}_{slip} = -(\rho - \rho_d)/\rho\nabla p$

$C_d = 24/Re_p(1+2\eta_c/(3\eta_d))/(1+\eta_c/\eta_d),\ Re_p = d_d\rho_c|\mathbf{u}_{slip}|/\eta$

$\eta = \eta_c(1 - \phi_d/\phi_{max})$^$(-2.5\phi_{max}\eta^*),\ \eta^* = (\eta_d + 0.4\eta_c)/(\eta_d + \eta_c)$

Mixture model

Slip model: Hadamard-Rybczynski, liquid droplets/bubbles

Mixture viscosity model: Krieger type model

Quantity	Value/Expression	Unit	Description
ϕ_{max}	0.62	1	Maximum packing concentration

□ Select by group

☑ Active in this domain

Figure 5-30: In the Subdomain Settings dialog box, on the Mixture Model tab, you specify which slip model and mixture viscosity model to use.

The Homogeneous flow model assumes that the velocity of the two phases are equal, that is, $\mathbf{u}_{\text{slip}} = 0$. In most cases there is a significant difference in the velocity fields, mainly due to gravity, that you need to account for.

The Hadamard-Rybczynski and the Schiller-Naumann models use the following relation for the slip velocity

$$\frac{3}{4}\frac{C_d}{d_d}\rho_c|\mathbf{u}_{\text{slip}}|\mathbf{u}_{\text{slip}} = -\frac{(\rho-\rho_d)}{\rho}\nabla p$$

where C_d is the drag coefficient (dimensionless). Essentially, you can interpret the relation as a balance between viscous drag, pressure, and gravity forces acting on the dispersed phase.

The Schiller-Naumann model models the drag coefficient according to

$$C_d = \begin{cases} \frac{24}{\text{Re}_p}(1+0.15\text{Re}_p^{0.687}) & \text{Re}_p < 1000 \\ 0.44 & \text{Re}_p > 1000 \end{cases}$$

where Re_p is the particle Reynolds number

$$\text{Re}_p = \frac{d_d\rho_c|\mathbf{u}_{\text{slip}}|}{\eta}$$

Because the particle Reynolds number depends on the slip velocity, an implicit equation must be solved for to obtain the slip velocity. Therefore, the application mode adds an additional equation for

$$|\mathbf{u}_{\text{slip}}|^2$$

if you use the Schiller-Naumann slip model. The Schiller-Naumann model is particularly well suited for solid particles in liquid.

The Hadamard-Rybczynski drag law is valid for particle Reynolds number less than 1, for both particles, bubbles, and droplets. The drag coefficient is then

$$C_d = \frac{24}{\text{Re}_p}\left(\frac{1+\frac{2}{3}\frac{\eta_c}{\eta_d}}{1+\frac{\eta_c}{\eta_d}}\right)$$

which yields the following explicit expression for the slip velocity for liquid droplets or bubbles

$$\mathbf{u}_{\mathrm{slip}} = -\frac{(\rho-\rho_d)d_d^2}{18\rho\eta_c}\left(\frac{1+\frac{\eta_c}{\eta_d}}{1+\frac{2}{3}\frac{\eta_c}{\eta_d}}\right)\nabla p$$

For solid particles, the slip velocity is

$$\mathbf{u}_{\mathrm{slip}} = -\frac{(\rho-\rho_d)d_d^2}{18\rho\eta_c}\nabla p$$

when $\mathrm{Re}_\mathrm{p} < 1$.

Finally, you can specify an arbitrary expression for the relative velocity, using the User defined option. For instance, you can give a constant velocity based on experimental data.

MIXTURE VISCOSITY

Two alternative forms are available for the mixture viscosity.

The most generally valid expression for the mixture viscosity is the Krieger type model

$$\eta = \eta_c\left(1-\frac{\phi_d}{\phi_{\mathrm{max}}}\right)^{-2.5\phi_{\mathrm{max}}\eta^*}$$

where ϕ_{max} is the maximum packing concentration, which for solid particles is approximately 0.62. The dimensionless parameter $\eta^* = 1$ for solid particles and

$$\eta^* = \frac{\eta_d + 0.4\eta_c}{\eta_d + \eta_c}$$

for drops and bubbles.

For liquid-liquid mixtures, another model for the mixture viscosity is given by

$$\eta = \phi_d\eta_d + \phi_c\eta_c$$

The mixture model application mode always uses the mixture viscosity in the particle Reynolds number expression used to calculate the slip velocity, thereby accounting for the increase in viscous drag due to particle-particle interactions.

TURBULENCE MODELING

For turbulence modeling, you can use the k-ε turbulence model. Turbulence modeling is particularly useful for dilute flows, that is, for flows with low dispersed phase volume fraction. For dense flow, the mixture viscosity usually becomes high. In that case, the flow is laminar and no turbulence modeling is necessary.

The k-ε turbulence model solves two extra transport equations for two additional variables: the turbulent kinetic energy, k ($\mathrm{m}^2/\mathrm{s}^2$), and the dissipation rate of turbulent energy, ε ($\mathrm{m}^2/\mathrm{s}^3$). The turbulent viscosity is

$$\eta_T = \rho C_\mu \frac{k^2}{\varepsilon}$$

where C_μ is a model constant.

The transport equation for the turbulent kinetic energy k is

$$\rho\frac{\partial k}{\partial t} - \nabla\cdot\left[\left(\eta + \frac{\eta_T}{\sigma_k}\right)\nabla k\right] + \rho\mathbf{u}\cdot\nabla k = \frac{1}{2}\eta_T(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)^2 - \rho\varepsilon$$

and the evolution of the turbulent energy's dissipation rate, ε, is determined by

$$\rho\frac{\partial \varepsilon}{\partial t} - \nabla\cdot\left[\left(\eta + \frac{\eta_T}{\sigma_\varepsilon}\right)\nabla \varepsilon\right] + \rho\mathbf{u}\cdot\nabla \varepsilon =$$

$$= \frac{1}{2}C_{\varepsilon 1}\frac{\varepsilon}{k}\eta_T(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)^2 - \rho C_{\varepsilon 2}\frac{\varepsilon^2}{k}$$

By default, the following variables are used for the dimensionless parameters:

CONSTANT	VALUE
C_μ	0.09
$C_{\varepsilon 1}$	1.44
$C_{\varepsilon 2}$	1.92
σ_k	1.0
σ_ε	1.3

The turbulence modeling must also be accounted for in the calculation of the dispersed phase velocity. This is accomplished by calculating the diffusion coefficient D_{md} (m^2/s) in Equation 5-86 as

$$D_{md} = \frac{\eta_T}{\rho\sigma_T}$$

where σ_T is the turbulent particle Schmidt number (dimensionless). The particle Schmidt number is usually suggested a value ranging from 0.35 to 0.7. In the application mode the default value is 0.35.

MASS TRANSFER AND INTERFACIAL AREA

For the mass transfer rate you can choose between no mass transfer, a two-film theory model, and a user-defined expression. For the two-film theory model, the mass transfer is

$$m_{dc} = k(c_d - c_c)Ma$$

where k denotes the mass transfer rate (m/s), and c_d and c_c are the species concentrations (mol/m^3) in the dispersed and the continuous phase, respectively. M equals the species' molecular weight (kg/mol), and a is the interfacial area between the two phases per unit volume. The interfacial area per volume can be calculated if the number density n, that is, the number of dispersed particles per volume, is known. Setting the application mode property **Solve for interfacial area/volume** to **On** (see Figure 5-29) automatically adds the following equation for the number density n:

$$\frac{\partial n}{\partial t} + \nabla \cdot (n\mathbf{u}_d) = 0$$

This equation states that a dispersed phase particle cannot disappear, appear, or merge with other particles, although it may expand or shrink.

The application mode calculates the interfacial area a (m^2/m^3) from

$$a = (4n\pi)^{1/3}(3\phi_d)^{2/3}$$

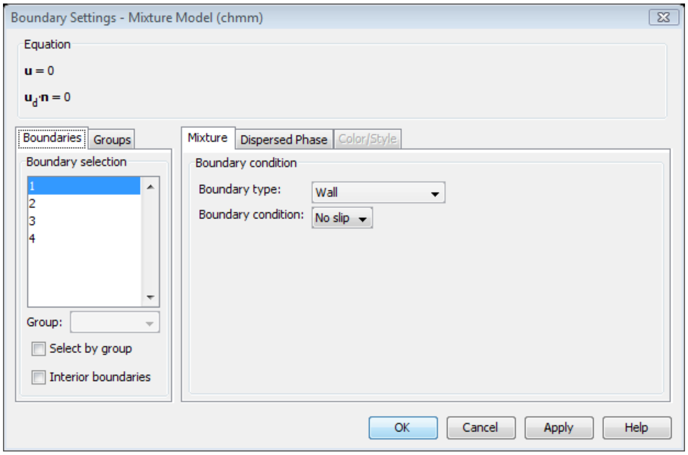
Boundary Conditions

You need to assign boundary conditions for the mixture velocity and pressure as well as the dispersed phase volume fraction. The following sections describe the available boundary conditions for the mixture and the dispersed phase volume fraction.

In all equations, $\mathbf{n}$ denotes the outward pointing unit vector normal to the boundary.

MIXTURE

For the mixture flow you first specify the boundary type, then choose an appropriate boundary condition. The boundary types are Wall, Inlet, Outlet, and Symmetry boundary. The boundary conditions available for each type are described in the following sections.



Boundary Settings - Mixture Model (chmm)

Equation

$\mathbf{u} = 0$

$\mathbf{u}_d \cdot \mathbf{n} = 0$

Boundary selection: 1

Boundary type: Wall

Boundary condition: No slip

□ Select by group

□ Interior boundaries

Figure 5-31: The Boundary Settings dialog box. For each Boundary type, several different Boundary conditions are available.

Boundary type: Wall

No slip Fixes the mixture velocity to zero.

$$\mathbf{u} = 0$$

Slip Sets the velocity component normal to the wall to zero.

$$\mathbf{u} \cdot \mathbf{n} = 0$$

Wall function Uses empirical models to model the thin laminar boundary layer close to the walls. Conditions on the viscous stress is determined from the law of the wall, according to

$$\mathbf{K} = -[\rho C_\mu^{0.25} k^{0.5} / (\log(\delta_w^+)/\kappa + 5.5)]\mathbf{u}$$

where $\mathbf{K}$ is the viscous stress:

$$\mathbf{K} = (\eta + \eta_T)(\nabla \mathbf{u}_l + \nabla \mathbf{u}_l^T)\mathbf{n}$$

The normal component of the velocity is set to zero.

$$\mathbf{u} \cdot \mathbf{n} = 0$$

For more information on the wall functions, see the documentation on the k-ε turbulence model for single-phase flows.

Boundary type: Inlet

Velocity Allows you to specify the velocity at the boundary.

$$\mathbf{u} = \mathbf{u}_0$$

Pressure Allows you to specify the pressure at the boundary and sets the viscous stress to zero.

$$p = p_0\,,\ (\eta + \eta_T)(\nabla \mathbf{u}_l + \nabla \mathbf{u}_l^T)\mathbf{n} = 0$$

Boundary type: Outlet

Velocity Allows you to specify the velocity at the boundary.

$$\mathbf{u} = \mathbf{u}_0$$

Pressure Allows you to specify the pressure at the boundary and sets the viscous stress to zero.

$$p = p_0\,,\ (\eta + \eta_T)(\nabla \mathbf{u}_l + \nabla \mathbf{u}_l^T)\mathbf{n} = 0$$

No viscous stress Sets the viscous stress to zero at the boundary, that is,

$$(\eta + \eta_T)(\nabla \mathbf{u}_l + \nabla \mathbf{u}_l^T)\mathbf{n} = 0$$

Boundary type: Symmetry

Symmetry Sets the normal component of the velocity to zero.

$$\mathbf{u} \cdot \mathbf{n} = 0$$

Axisymmetry Boundary condition at the axial symmetry line (axisymmetric models only).

DISPERSED PHASE

For the dispersed phase volume fraction, the following boundary conditions are available.

Dispersed phase concentration Allows you to specify the dispersed phase volume fraction.

$$\phi_d = \phi_d^0$$

Dispersed phase flux Allows you to specify the dispersed phase flux through the boundary.

$$-\mathbf{n} \cdot (\phi_d \mathbf{u}_d) = N_\phi$$

Dispersed phase outlet For boundaries where the dispersed phase flows outwards with the dispersed phase velocity $\mathbf{u}_d$. No condition is imposed on the volume fractions at the boundary.

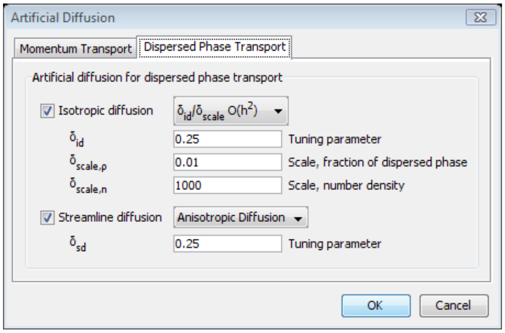
Insulation/Symmetry Sets the dispersed phase flux through the boundary to zero.

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

Axial symmetry For boundaries at the axial symmetry line (axisymmetric models only).

Numerical Stabilization

For the momentum equations as well as the dispersed phase transport equation you can add isotropic diffusion, streamline diffusion, or both. By default, the momentum transport equation is stabilized using streamline diffusion. The dispersed phase transport is stabilized by streamline diffusion and isotropic diffusion ($O(h^2)$) by default. Note that you must specify a suitable scale for the dispersed phase volume fraction and the number density (see Figure 5-32). Appropriate scales are the maximum values of the dispersed phase volume fraction and the maximum density, respectively.



Artificial diffusion for dispersed phase transport

☑ Isotropic diffusion: $\delta_{id}/\delta_{scale}$ O(h^2)

δ_{id}: 0.25 Tuning parameter

$\delta_{scale,p}$: 0.01 Scale, fraction of dispersed phase

$\delta_{scale,n}$: 1000 Scale, number density

☑ Streamline diffusion: Anisotropic Diffusion

δ_{sd}: 0.25 Tuning parameter

Figure 5-32: Artificial diffusion settings for the dispersed phase transport.

See “Numerical Stability—Artificial Diffusion” on page 121 for more details on the different artificial diffusion techniques.

Example Models

The models “Contaminant-Removal from Wastewater in a Secondary Clarifier” on page 294 and “Two-Phase Flow Modeling of a Dense Suspension” on page 305 of the *Chemical Engineering Module Model Library* both use the Mixture Model application mode.

References

1. M. Manninen, V. Taivassalo, and S. Kallio, *On the Mixture Model for Multiphase Flow,* VTT Publications 288, VTT Energy, Nuclear Energy, Technical Research Center of Finland (VTT), 1996.

2. C. Crowe, M. Sommerfeld, and Y. Tsuji, *Multiphase Flows with Droplets and Particles,* CRC Press, 1998.

6

Energy Transport

Energy transport is an important part of chemical engineering. Most chemical reactions either require or produce heat, which in turn affects both the reactions themselves and other physical processes connected to the system. Supplying or removing heat in process equipment is also a central aspect of chemical and unit operation engineering.

The energy transport equations included in the Chemical Engineering Module allow for the modeling of reacting systems involving exothermic or endothermic reactions. Application modes for simple heat transfer through conduction as well as through both convection and conduction, with available terms for reaction kinetics are provided.

The following sections presents the application modes for energy transport in the Chemical Engineering Module.

Convection and Conduction

Subdomain Equations

As mentioned in the introduction to Chapter 5, “Momentum Transport,” the most general energy balance is one of the compressible Navier-Stokes equations:

$$\rho C_p\left(\frac{\partial T}{\partial t} + (\mathbf{u}\cdot\nabla)T\right) = -(\nabla\cdot\mathbf{q}) + \tau{:}\mathbf{S} - \frac{T}{\rho}\frac{\partial\rho}{\partial T}\bigg|_p\left(\frac{\partial p}{\partial t} + (\mathbf{u}\cdot\nabla)p\right) + Q \qquad (6\text{-}1)$$

where

- ρ is the density (kg/m^3)
- $\mathbf{u}$ is the velocity vector (m/s)
- p is pressure (Pa)
- τ is the viscous stress tensor (Pa)
- C_p is the specific heat capacity at constant pressure (J/(kg·K))
- T is absolute temperature (K)
- $\mathbf{q}$ is the heat flux vector (W/m^2)
- Q contains the heat sources (W/m^3)

S is the strain rate tensor:

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

The operation “:” is a contraction and can in this case be written on the following form:

$$\mathbf{a}{:}\mathbf{b} = \sum_n\sum_m a_{nm}b_{nm} \qquad (6\text{-}2)$$

The Convection and Conduction application mode uses Fourier’s law of conduction which states that the heat flux, **q**, is proportional to the temperature gradient:

$$q_i = -k\frac{\partial T}{\partial x_j} \qquad (6\text{-}3)$$

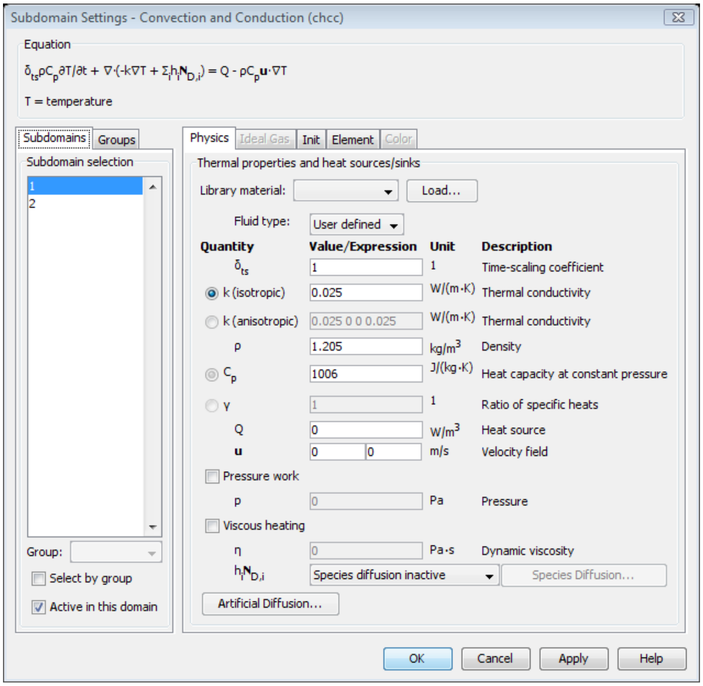
where k is the thermal conductivity. Inserting Equation 6-3 into Equation 6-1 gives the energy balance used in the Convection and Conduction application mode

$$\rho C_p\left(\frac{\partial T}{\partial t} + u \cdot \nabla T\right) = \nabla \cdot (k \nabla T) + Q \qquad (6\text{-}4)$$

The second and third terms on the right-hand side of Equation 6-1 are usually small and are not included per default. You can read more about these terms in the section "Subdomain Settings."

Subdomain Settings

Figure 6-1 shows the **Subdomain Settings** dialog box for the two-dimensional Convection and Conduction application mode. Here you specify the material properties.



Subdomain Settings - Convection and Conduction (chcc)

Equation

$\delta_{ts}\rho C_p \partial T/\partial t + \nabla\cdot(-k\nabla T + \Sigma_i h_i \mathbf{N}_{D,i}) = Q - \rho C_p \mathbf{u}\cdot\nabla T$

T = temperature

Subdomain selection: 1 (selected), 2

Group:

☐ Select by group

☑ Active in this domain

Thermal properties and heat sources/sinks

Library material:

Fluid type: User defined

Quantity	Value/Expression	Unit	Description
δ_{ts}	1	1	Time-scaling coefficient
● k (isotropic)	0.025	W/(m·K)	Thermal conductivity
○ k (anisotropic)	0.025 0 0 0.025	W/(m·K)	Thermal conductivity
ρ	1.205	kg/m³	Density
● C_p	1006	J/(kg·K)	Heat capacity at constant pressure
○ γ	1	1	Ratio of specific heats
Q	0	W/m³	Heat source
u	0 0	m/s	Velocity field
☐ Pressure work			
p	0	Pa	Pressure
☐ Viscous heating			
η	0	Pa·s	Dynamic viscosity
$h_i \mathbf{N}_{D,i}$	Species diffusion inactive		

Figure 6-1: The Subdomain Settings dialog box for the two-dimensional Convection and Conduction application mode. Physics page.

The **Time-scaling coefficient**, δ_{ts}, is normally 1, but you can change the time scale, for example, from seconds to minutes by setting it to 1/60.

The **Thermal conductivity** can be given by a scalar value or as a tensor, k_{ij}. In the latter case, Equation 6-3 is replaced by the general version of Fourier's law:

$$q_i = -k_{ij}\frac{\partial T}{\partial x_j}$$

This formulation describes heat conduction in anisotropic materials. This is very rare for fluids but common for solids.

The velocity field, $\mathbf{u}$, can be expressed analytically or obtained by coupling to a momentum balance.

Chemical reactions that affect energy balances can be accounted for by introducing heat sources or sinks, dependent on the reaction kinetics. The radiation contribution can also be expressed as a volume source.

FLUID TYPE

The **Fluid type** list has four options:

- **User defined**
- **Ideal gas**
- **Gas**
- **Liquid**

When you select a **Fluid type**, only those properties compatible with the chosen type become available. Two properties are special in this context. The first is the **Ratio of specific heats**, γ, which is used by the Galerkin least-squares (GLS) method for artificial diffusion. Selecting **Ideal Gas** can make γ available even if GLS is not active. If GLS is active, **Ideal Gas** is the only setting that can inactivate γ. The other special property is **Species Diffusion**, which is active only if the model includes a Convection and Diffusion application mode with several species.

User Defined

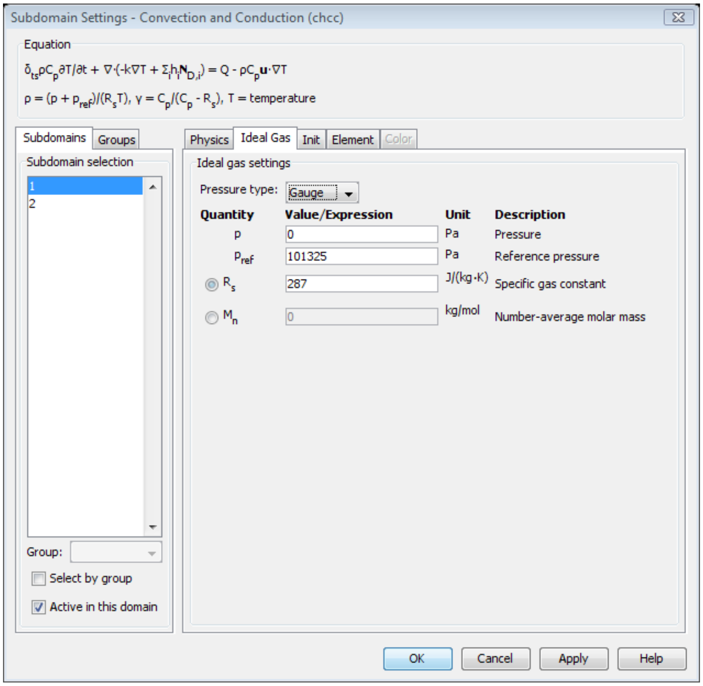
This option makes all properties available, except γ and **Species Diffusion** as outlined above.

Ideal Gas

This option prescribes the ideal gas law. The **Ideal Gas** page becomes active, the **Density** edit field is deactivated, and the option buttons in front of $\mathbf{C_p}$ and γ are enabled (see Figure 6-1).

By using the option buttons in front of $\mathbf{C_p}$ and γ, you can choose between specifying $\mathbf{C_p}$ or γ. The one of these two that is not specified is calculated from the ideal gas law.

The **Ideal Gas** page is shown in Figure 6-2. Here you enter data used to calculate ρ and C_p or γ. The **Pressure type** list has two options: **Gauge** and **Absolute**. If the former is chosen, the density becomes a function of $p + \mathrm{p_{ref}}$. If the latter is chosen, the density is a function of p only. Use the **Pressure** edit field to specify p, either as a constant or by coupling a momentum balance. The **Reference pressure** edit field can contain either a constant or a dual pressure variable that via an ODE keeps track of the pressure level in the domain (Ref. 1). Finally, you can choose to specify either the **Specific gas constant**, $\mathrm{R_s}$, or the **Number-average molar mass**, $\mathrm{M_n}$. If you chose the latter, the molar gas constant $\mathrm{R_g}$=8.314 J/(mol·K) is used.



Subdomain Settings - Convection and Conduction (chcc)

Equation

$\delta_{ts}\rho C_p\partial T/\partial t + \nabla\cdot(-k\nabla T + \Sigma_i h_i \mathbf{N}_{D,i}) = Q - \rho C_p \mathbf{u}\cdot\nabla T$

$\rho = (p + p_{ref})/(R_s T)$, $\gamma = C_p/(C_p - R_s)$, T = temperature

Subdomain selection: 1

Ideal gas settings

Pressure type: Gauge

	Quantity	Value/Expression	Unit	Description
	p	0	Pa	Pressure
	p_{ref}	101325	Pa	Reference pressure
●	R_s	287	J/(kg·K)	Specific gas constant
○	M_n	0	kg/mol	Number-average molar mass

Group:

☐ Select by group

☑ Active in this domain

Figure 6-2: The Ideal Gas page in the Subdomain Settings dialog box for the Convection and Conduction application mode.

Gas

This option makes all properties available, except γ and **Species Diffusion**.

Liquid

This option assumes that the density is independent of the pressure. This is an accurate assumption for all liquids unless the pressure is extremely high.

This option disables the **Pressure work** option because that term is always small for liquids.

PRESSURE WORK

In Equation 6-1, the term

$$-\frac{T}{\rho}\left(\frac{\partial\rho}{\partial T}\right)_p\left(\frac{\partial p}{\partial t}+u_j\frac{\partial p}{\partial x_j}\right)$$

represents pressure work but also includes acoustic effects. The acoustic effect has extremely short time and spatial scales and is therefore very demanding with respect to spatial resolution and time advancement. The term can, however, be shown to be small for most flows at low Mach numbers and is most often neglected. There are some special cases where this term is important even for low Mach numbers. It is, for example, responsible for the heating of an enclosed gas if the pressure is increased.

Select the **Pressure work** check box to include the term

$$-\frac{T}{\rho}\left(\frac{\partial\rho}{\partial T}\right)_p\frac{\partial p}{\partial t}$$

which is an accurate approximation for low Mach numbers (Ref. 2). The edit field for the pressure is used to provide the Convection and Conduction application mode with the name of the pressure variable. This can either be given by a pressure level ODE (Ref. 1) or a momentum balance application mode.

Pressure work is only available for transient analysis.

VISCOUS HEATING

The **Viscous heating** check box includes the term

$$\tau{:}\mathbf{S} = \eta\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}-\frac{2}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right)\frac{\partial u_i}{\partial x_j}$$

It is normally neglected for low-Mach number flows but, if the fluid is highly viscous, this term can be of importance for flows with high rate of strain. Viscous heating is computationally rather expensive to include but poses no numerical difficulties.

Use the **Dynamic viscosity** edit field to provide the Convection and Conduction application mode with either an analytical expression for the viscosity or the name of a variable for the dynamic viscosity.

SPECIES DIFFUSION

If your model includes species diffusion, select the application mode for the species mass transport from the **$h_iN_{D,i}$** list. Then click the **Species Diffusion** check box and type in the enthalpy of each species.

Boundary Conditions

The available boundary conditions in the Convection and Conduction application mode include the following conditions:

HEAT FLUX

The Heat flux condition is:

$$\mathbf{q}\cdot\mathbf{n} = q_0 \tag{6-5}$$

Here $\mathbf{q}$ denotes the heat flux vector. The source term q_0 can be arbitrarily defined to represent flux into an infinite medium, a radiation term, or heat from chemical reactions or nuclear decay.

THERMAL INSULATION

The Thermal insulation condition has the thermal flux set to zero:

$$\mathbf{q}\cdot\mathbf{n} = 0 \tag{6-6}$$

This boundary condition can also be used as a symmetry conditions.

TEMPERATURE

The Chemical Engineering Module also allows for the imposition of a Temperature condition at a boundary:

$$T = T_0 \tag{6-7}$$

CONVECTIVE FLUX

The Convective flux boundary condition assumes that all energy passing through this boundary does so through the convective flux mechanism. This firstly assumes that any heat flux due to conduction across this boundary is zero

$$\mathbf{q}_{\mathrm{cond}} \cdot \mathbf{n} = -k\nabla T \cdot \mathbf{n} = 0 \qquad (6\text{-}8)$$

so that the resulting equation becomes

$$\mathbf{q} \cdot \mathbf{n} = \rho C_p T \mathbf{u} \cdot \mathbf{n} \qquad (6\text{-}9)$$

This is a useful boundary condition, particularly in convection-dominated energy balances where the outlet temperature is unknown.

AXIAL SYMMETRY

The axial symmetry condition is available only for axisymmetric versions of the heat transfer application modes. Apply this boundary condition at the symmetry axis $r = 0$. Note that from the mathematical point of view, this condition is identical to thermal insulation.

CONTINUITY

The default setting for interior boundaries is Continuity, which is a special case of the above Heat source/sink condition. In the absence of sources or sinks, that condition becomes $-\mathbf{n}_1 \cdot (\mathbf{q}_1 - \mathbf{q}_2) = 0$. This means that the heat flux in the normal direction is continuous across the boundary. This is the default boundary condition on interior boundaries.

The temperature is naturally continuous on an internal boundary following the continuity of the finite element field. Therefore, the Continuity boundary condition is identical to the condition that applies between any two neighboring elements in the mesh. In fact, as long as you have not selected **Enable interior boundaries** in the **Boundary Settings** dialog box, the interior boundaries are not in any way different from any other mesh element boundaries, where the Continuity condition effectively applies.

The Continuity boundary condition is available on interior boundaries and assembly pairs only.

HEAT FLUX DISCONTINUITY

The Heat flux discontinuity condition is available on interior boundaries and pairs representing borders between parts in an assembly:

$$-\mathbf{n}_1 \cdot \mathbf{q}_1 - \mathbf{n}_2 \cdot \mathbf{q}_2 = q \quad \text{on } \partial\Omega$$

The right-hand side represents a flux discontinuity, or equivalently, a heat source or heat sink depending on sign. The temperature is always continuous due to the continuity of the finite element field. If the right-hand side is zero, this equation specifies continuity also in the normal heat flux.

THIN THERMALLY RESISTIVE LAYER

Use this boundary condition to model a thin layer of thermally resistive material using these equations:

$$\mathbf{n}_1 \cdot (k_1 \nabla T_1) = \frac{k}{d}(T_2 - T_1)$$
$$\mathbf{n}_2 \cdot (k_2 \nabla T_2) = \frac{k}{d}(T_1 - T_2)$$

The layer has the thickness d and the thermal conductivity k. This boundary condition is only available at the border between the parts in an assembly.

Extension to Turbulent Flows

If the **Turbulence model** application mode property has the value **k–ε** or **k–ω**, k is replaced by $k + k_T$, where k_T is the turbulent heat conductivity. A common expression for k_T is

$$k_T = \frac{C_p \eta_T}{\mathrm{Pr}_T}$$

where Pr_T is the turbulent Prandtl number. Even though there is evidence that the turbulent Prandtl number varies with the molecular Prandtl number and the characteristic of the turbulence (Ref. 3), a constant value 0.85 is often used.

THERMAL WALL FUNCTION

An algebraic relationship—the logarithmic wall function—describes the momentum transfer at the solid-fluid interface. This means that the modeled fluid domain ends at the top of the laminar boundary layer where the fluid experiences a significant wall-tangential velocity. See the section “Boundary Conditions” on page 141 for details. Similar to the fluid velocity, the temperature is not modeled in the laminar sublayer. Instead of assuming continuity of the temperature across the layer, a thermal wall function is used. There will be a jump in temperature from the solid surface to the fluid due to the omitted laminar sublayer. To take this into account, use the heat flux boundary condition at solid walls, with q according to

$$q = \frac{\rho C_p C_\mu^{1/4} k_w^{\ 1/2} (T_w - T)}{T^+}$$

C_μ is a constant of the turbulence model and k_w is the value of the turbulent kinematic energy at the wall. Furthermore, T_w equals the temperature of the solid at the wall, while T is the temperature of the fluid on the other side of the omitted laminar sublayer.

The quantity T^+ is related to the dimensionless wall offset, δ_w^+, through the definition

$$T^+ = \frac{\mathrm{Pr}_T}{\kappa} \ln(\delta_w^+) + \beta$$

Here κ is the von Karman constant, which is equal to 0.42; and β is a model constant equal to 3.27. The wall offset in viscous units is given by

$$\delta_w^+ = \frac{\delta_w C_\mu^{1/4} k_w^{1/2}}{\nu}$$

where δ_w is the specified wall offset, which in COMSOL Multiphysics defaults to half the local mesh size at the boundary, and $\nu = \eta/\rho$ denotes the kinematic viscosity.

Example Model

The model "A 3D Model of a MEMS Heat Exchanger" on page 119 in the Chemical Engineering Module Model Library demonstrates how you can use the Convection and Conduction application mode if you have an analytical expression for the velocities. The model "MEMS Heat Exchanger" on page 142 shows how to obtain a velocity field by coupling the Convection and Conduction application mode to an Incompressible Navier-Stokes application mode.

References

1. H. Paillare, C. Viozat, A. Kumbaro, and I. Toumi, "Comparison of low Mach number models for natural convection problems," *Heat and Mass Transfer*, vol. 36, pp. 567–573, 2000.

2. C.-D. Munz, M. Dumbser, and M. Zucchini, “The Multiple Pressure Variables Method for Fluid Dynamics and Aeroacoustics at Low Mach Numbers,” in *Numerical Methods for Hyperbolic and Kinetic Problems*, S. Cordier, T. Goudon & E. Sonnendrücker (editors), EMS Publishing House, pp. 335–339, 2005.

3. W. M. Kays, “Turbulent Prandtl Number—Where Are We?,” *Journal of Heat Transfer,* vol. 116, pp. 284–295, 1994.

Conduction

The Conduction application mode is a simplification of the Convection and Conduction application mode with a couple of added features. Basically, the convective part of Equation 6-4 is removed and a couple of source terms are added:

$$\rho C_p\frac{\partial T}{\partial t}+\nabla\cdot(-k\nabla T) = Q+h(T_{\mathrm{ext}}-T)+C(T_{\mathrm{amb}}^4-T^4) \qquad (6\text{-}10)$$

where h is the heat transfer coefficient (W/(m^3·K)) and C is a constant for describing the material's ability to radiate to a nonreflecting surface (W/(m^3·K^4)). The second term on the right-hand side of Equation 6-10 is a term to describe the transfer of energy to an outside medium, at constant temperature, where h can be held constant or set to an arbitrary equation. The third term is denoted as heat transfer through ambient radiation, sometimes (perhaps erroneously) referred to as black-body radiation. The term specifies radiation emitting from the modeled domain toward a body that can emit, but not reflect, radiation.

The boundary conditions for the Conduction application mode are similar to those for the Convection and Conduction application mode. Thermal insulation (Equation 6-6) and Temperature (Equation 6-7) boundary conditions are available, while the Convective flux boundary condition (Equation 6-8) is excluded. The heat flux conditions extends Equation 6-5 to include an ambient radiation and heat transfer condition:

$$\mathbf{q}\cdot\mathbf{n} = q_0+h(T_{\mathrm{ext}}-T)+C(T_{\mathrm{amb}}^4-T^4) \qquad (6\text{-}11)$$

Although the heat transfer and ambient radiation terms are available in the 3D boundary conditions for the Conduction application mode (Equation 6-11), they are of course not a part of the main equations (Equation 6-10).

Note: When including radiation in the model, you must work with absolute (thermodynamical) temperature units. See "Using Units" on page 183 in the *COMSOL Multiphysics User's Guide*.

Pseudo 2D and Pseudo 3D Energy Balance Application Modes

There is one additional application mode (in 1D and 2D) for modeling stationary energy balances, which is suitable for applications where convection is constant and dominates exclusively in one direction so that you can completely neglect the conduction term. This means that it is possible to describe the model geometries by extrusions of a cross section for applications such as ducts and tubular reactors.

Provided that you know the velocity profile in the cross section and that the model is defined for steady state, you can replace the space variable along the direction of the flow with time, t. This implies that a time increment, at a given velocity, represents a displacement along the flow direction. This allows for the modeling of a 3D system in 2D (or a 2D system in 1D) by letting time represent the third dimension, for example the z-coordinate. The equation for this type of energy balance is

$$u_{\mathrm{dl}}\rho C_p\frac{\partial T}{\partial t} + \nabla\cdot(-k\nabla T) = Q \tag{6-12}$$

where u_{dl} gives the velocity profile. The convective term is included in the first term of the equation and time stepping automatically implies a change in position in the model domain. Use of these Pseudo Energy Balance application modes reduces problem size and computational time. The application modes' boundary conditions include those given in Equation 6-5, Equation 6-6, and Equation 6-7, but exclude the Convective flux condition (Equation 6-8).

Example Model

For an example demonstrating how to use Chemical Engineering Module's Conduction application modes, see "Heating of a Finite Slab" on page 114 in the *Chemical Engineering Module Model Library.* This model demonstrates the different menus and operations and provides a sample study of a heat-transfer process.

7

Mass Transport

Mass transfer is an important part of chemical engineering, because this is the field of engineering that considers the conversion of one type of substance into another. A lot of this occurs through chemical reactions, although separation and other unit operations are an important part. Most chemical reactions either require or produce heat, which in turn affects both the reactions themselves and other physical processes connected to the system. Furthermore, most unit operations and reactors require an optimized and steady production of product, which is often supplied through flow. This means that most mass balance applications require coupling to momentum and energy balances.

The mass balance equations in the Chemical Engineering Module account for mass transport by diffusion, migration, and convection—either alone or in combination with each other. The Chemical Engineering Module includes an application mode for mass transfer through diffusion as well as convection and diffusion according to Fick's Law.

A specialized mass transfer application mode included in the module is the description of diffusion as described by the Maxwell-Stefan equations. These take into account the diffusion properties of a solute compared to a solvent and vice versa and do not assume the one-way effect as is done with Fick's Law. Convection is also included in this application mode.

Finally, two application modes are included to consider mass transport through diffusion and convection as well as electric migration. The Nernst-Planck application mode includes a migration term along with convection and conduction mass transport, as does the Electrokinetic Flow application mode. The difference between the two is that the Nernst-Planck application mode also includes the electroneutrality condition.

All application modes include edit fields for defining arbitrary reaction kinetics, or material source and sink terms. You have access to all material properties, such as the diffusion coefficient, and it is easy to specify anisotropy.

The following sections present the application modes for mass balances in the Chemical Engineering Module.

Convection and Diffusion

In the Diffusion and the Convection and Diffusion application modes in the Chemical Engineering Module, Fick's law describes the diffusive transport in the flux vector. Fick's law is adequate when the diffusing species is dilute with respect to a solvent. Assuming a binary mixture of solute A in solvent B, concentrations of up to 10 mol% of A can be considered dilute. For concentrated mixtures, the Maxwell-Stefan Diffusion and Convention application mode is recommended (see "Maxwell-Stefan Diffusion and Convection" on page 217).

Subdomain Equations

The mass balance equation implemented in the Convection and Diffusion application mode is given by:

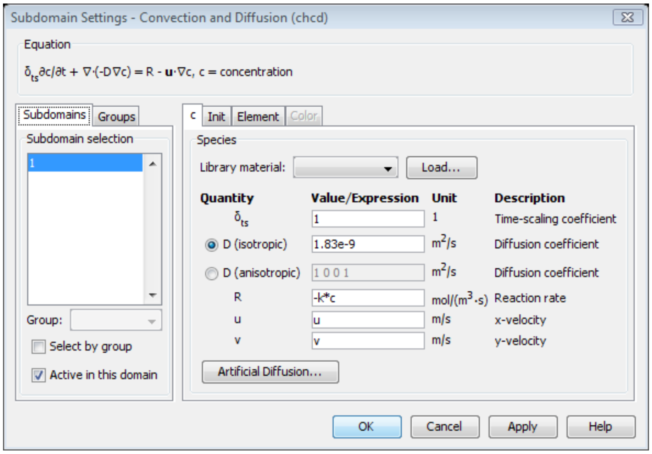
$$\frac{\partial c_i}{\partial t} + \nabla \cdot (-D_i \nabla c_i) = R_i - \mathbf{u} \nabla \cdot c_i \tag{7-1}$$

where

- c_i is the concentration of species i (mol/m^3)
- D_i denotes its diffusion coefficient (m^2/s)
- Q contains the heat sources (W/m^3)
- $\mathbf{u}$ is the velocity vector (m/s)

The first term on the left-hand side of Equation 7-1 corresponds to the accumulation of species i in the system. The second term describes the diffusional transport, accounting for interaction between the dilute species and the solvent. On the right-hand side, the first term represents a source or sink, typically due to chemical reaction. The last term corresponds to transport due to convection. Arbitrary kinetic expressions of the reacting species can be introduced in the application mode interface.

Finally, the second term accounts for the convective transport due to a velocity field **u**. This field can be expressed analytically or be obtained by coupling a momentum balance to the equation system.



Subdomain Settings - Convection and Diffusion (chcd)

Equation

$\delta_{ts}\partial c/\partial t + \nabla\cdot(-D\nabla c) = R - \mathbf{u}\cdot\nabla c$, c = concentration

Subdomain selection: 1

Group:

☐ Select by group

☑ Active in this domain

Species

Library material:

Quantity	Value/Expression	Unit	Description
δ_{ts}	1	1	Time-scaling coefficient
● D (isotropic)	1.83e-9	m^2/s	Diffusion coefficient
○ D (anisotropic)	1 0 0 1	m^2/s	Diffusion coefficient
R	-k*c	$mol/(m^3\cdot s)$	Reaction rate
u	u	m/s	x-velocity
v	v	m/s	y-velocity

Figure 7-1: The subdomain settings of the Convection and Diffusion application mode.

Equation 7-1 expresses the mass balance on nonconservative form, which is the default form in the Convection and Diffusion applications mode. It is applicable to when the overall mass balance of the system satisfies:

$$\nabla\cdot\mathbf{u} = 0 \tag{7-2}$$

Boundary Conditions

You can choose from a number of boundary conditions:

Use the Flux condition to define a mass flux

$$\mathbf{N}_i\cdot\mathbf{n} = N_0 \tag{7-3}$$

where the boundary source term, N_0, can be arbitrarily defined. For instance, N_0 can represent flux into an infinite medium, a phase change, or a flux due to chemical reaction.

The Insulation/Symmetry condition has the source term in Equation 7-3 set to zero:

$$\mathbf{N}_i\cdot\mathbf{n} = 0 \tag{7-4}$$

There is also a Concentration boundary condition:

$$c_i = c_{i,0} \tag{7-5}$$

The Convective flux boundary condition assumes that the mass flow through the boundary is convection dominated. In other words, the mass flux due to diffusion across this boundary is zero

$$\mathbf{n} \cdot (-D_i \nabla c_i) = 0 \tag{7-6}$$

so that

$$\mathbf{N}_i \cdot \mathbf{n} = c_i \mathbf{u} \cdot \mathbf{n} \tag{7-7}$$

This is a useful boundary condition, particularly in convection-dominated mass balances where the outlet concentration is unknown.

BOUNDARY CONDITIONS FOR INTERIOR BOUNDARIES

The following boundary conditions are available on interior boundaries and pair boundaries in assemblies:

Continuity is the default boundary condition on interior boundaries and pair boundaries; it is not applicable to exterior boundaries.

$$\mathbf{n} \cdot (\mathbf{N}_1 - \mathbf{N}_2) = 0$$

The Flux discontinuity boundary condition represents a discontinuity in the mass flux across an interior boundary or a border between parts in an assembly:

$$-\mathbf{n} \cdot (\mathbf{N}_1 - \mathbf{N}_2) = N_0$$

It is not applicable to exterior boundaries.

You can use the Thin boundary layer condition to model a thin layer of a material with a small diffusion coefficient compared to the adjacent domains:

$$\mathbf{n}_1 \cdot (-D\nabla c + c\mathbf{u})_1 = \frac{D}{d}(c_1 - c_2)$$

$$\mathbf{n}_2 \cdot (-D\nabla c + c\mathbf{u})_2 = \frac{D}{d}(c_2 - c_1)$$

The layer has the thickness d and the diffusion coefficient D. This boundary condition is only available at the border between the parts in an assembly.

Example Model

For an example of how to set up and use the Convection and Diffusion application mode, see the section “Quick Start Model” on page 16.

Diffusion

Subdomain Equations

The Diffusion application mode is appropriate when convection does not contribute to mass transport. The application mode sets up the following equation:

$$\frac{\partial c_i}{\partial t} + \nabla \cdot (-D_i \nabla c_i) = R_i \qquad (7\text{-}8)$$

Boundary Conditions

The boundary conditions for the Diffusion application mode are likewise similar to those for the Convection and Diffusion application mode. The Insulation/Symmetry (Equation 7-4) and Concentration (Equation 7-5) boundary conditions are available, while the Convective flux boundary condition (Equation 7-7) is excluded. The Flux condition extends Equation 7-3 to include a mass transfer condition

$$\mathbf{N}_i \cdot \mathbf{n} = N_0 + k_c (c_{\mathrm{bulk}} - c) \qquad (7\text{-}9)$$

where $\mathbf{N}_i \equiv -D_i \nabla c_i$, k_c is the mass transfer coefficient, and c_{bulk} is the concentration in the surroundings of the modeled system. The transfer coefficient is given by boundary-layer theory, and the condition requires that the flow conditions and concentration in the bulk outside of the boundary are known.

You have also access to the same boundary conditions on pair boundaries for modeling flux discontinuities and thin boundary layers between parts in an assembly.

Pseudo 2D and Pseudo 3D Mass Balance Application Modes

There is one additional application mode (in 1D and 2D) for modeling stationary mass balances, which is suitable for applications where convection is constant and dominates exclusively in one direction so that you can completely neglect the diffusion term. This means that it is possible to describe the modeled geometries by extrusions of a cross section for applications such as ducts and tubular reactors.

Providing that the velocity profile can be expressed in the cross section and that the model is defined for steady state, you can replace the space variable along the direction of the flow with time, t. This implies that a time increment, at a given velocity, represents a displacement along the direction of the flow. This allows for the modeling of a 3D system in 2D (or a 2D system in 1D) by letting time represent the third dimension, for example, the z-coordinate. The equation for this type of mass balance is

$$u_{\mathrm{dl}}\frac{\partial c_i}{\partial t} + \nabla \cdot (-D_i \nabla c_i) = R_i \tag{7-10}$$

where u_{dl} gives the velocity profile. The convective term is now included in the first term of the equation and time stepping automatically implies a change in position in the modeled domain. Use of these Pseudo Mass Balance application modes reduces problem size and computational time. The application modes' boundary conditions include those given in Equation 7-3, Equation 7-4, and Equation 7-5, but excludes the Convective flux condition (Equation 7-7).

Example Model

The model "Monolithic Reactor" on page 461 of the *Chemical Engineering Module Model Library* demonstrates how to use the Pseudo formulation of the Convection and Conduction application mode.

Maxwell-Stefan Diffusion and Convection

Fick's law is based upon the assumption that species dissolved in a solution or gas stream interact only with the solvent or the carrier gas. The diffusion coefficients describing such interactions are independent of concentration. Yet, in concentrated solutions or gas mixtures, where relative concentrations are of the same order of magnitude, all species interact with each other and themselves. Their diffusion coefficients are therefore species and concentration dependent, and can also depend on temperature and pressure. The Maxwell-Stefan Diffusion and Convection application mode allows you to accurately model concentrated mixture by setting up the proper multicomponent mass transport equations.

Subdomain Equations

The Maxwell-Stefan Diffusion and Convection application mode is built upon the following mass transport equation:

$$\frac{\partial}{\partial t}(\rho\omega_i) + \nabla \cdot (\mathbf{j}_i + \rho\omega_i\mathbf{u}) = R_i \qquad (7\text{-}11)$$

Above, $\mathbf{j}_i$ describes the diffusion-driven transport, as outlined by Curtiss and Bird (Ref. 1 and Ref. 2). This transport is a function of temperature and a diffusional driving force $\mathbf{d}_j$:

$$\mathbf{j}_i = (-D_i^T \nabla \ln T) - \rho_i \sum_{j=1}^{n} \tilde{D}_{ij}\mathbf{d}_j \qquad (7\text{-}12)$$

In Equation 7-12, $\mathbf{d}_j$ accounts for diffusional driving forces through concentration, pressure and external forces, such as gravity. Substituting Equation 7-12 into Equation 7-11 gives the mass transport equation set up by the application mode:

$$\frac{\partial\rho\omega_i}{\partial t} + \nabla \cdot \left(\rho\omega_i\mathbf{u} - \rho\omega_i \sum_{j=1}^{n} \tilde{D}_{ij}\left(\nabla x_j + (x_j - \omega_j)\frac{\nabla p}{p}\right) - D_i^T \frac{\nabla T}{T}\right) = R_i \qquad (7\text{-}13)$$

In Equation 7-13, the species molar fraction, x_j, and its gradient ∇x_j , can be expressed in terms of the mass fractions ω_j:

$$x_j = \frac{\omega_j}{M_j} \cdot M \tag{7-14}$$

$$\nabla x_j = \frac{M^2}{M_j} \cdot \sum_{\substack{j = 1 \\ k \neq j}}^{n} \left[\frac{1}{M} + \omega_k\left(\frac{1}{M_k} - \frac{1}{M_j}\right)\right]\nabla\omega_k \tag{7-15}$$

This all implies that the only dependent variable in the application mode is the mass fraction ω, while the temperature field, T, the pressure field, p, and the velocity, $\mathbf{u}$, are obtained in combination with energy, momentum, and continuity equations.

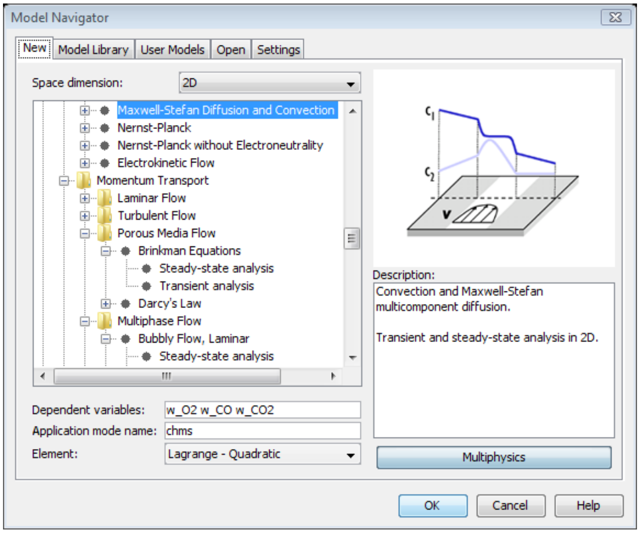
In the equations above

- ρ denotes the density (kg/m^3)
- ω_i is the mass fraction of species i
- x_j is the molar fraction of species j
- $\mathbf{u}$ is the velocity vector (m/s)
- p is the pressure (Pa)
- D_{ij} is the ij component of the multicomponent Fick diffusivity.
- D_i^T denotes the generalized thermal diffusion coefficient kg/(m·s)
- M_j is the molar mass of species j (kg/mol)
- M is the molar mass of the mixture (kg/mol)
- T is the temperature (K)
- R_i the reaction rate (kg/(m^3·s))

The Maxwell-Stefan Diffusion and Convection application mode in the Chemical Engineering Module solves (n – 1) versions of Equation 7-13 for n species. The final species is solved through the fact that:

$$\sum_{i = 1}^{n} \omega_i = 1 \tag{7-16}$$

This species is chosen by COMSOL Multiphysics as being the last entered variable name when you have entered all variables in the **Dependent variables** edit field in the **Model Navigator**.



Model Navigator

Space dimension: 2D

Description:
Convection and Maxwell-Stefan multicomponent diffusion.

Transient and steady-state analysis in 2D.

Dependent variables: w_O2 w_CO w_CO2

Application mode name: chms

Element: Lagrange - Quadratic

Figure 7-2: The last entered variable in the Dependent variables edit field will be calculated by Equation 7-16.

MULTICOMPONENT DIFFUSIVITIES

To solve Equation 7-13 you need to be able to specify the *multicomponent Fick diffusivities*. The Maxwell-Stefan Diffusion and Convection application mode includes two methods to do this: one for two or three species and one for four or more species.

For two or three species, the multicomponent Fick diffusivities are related to the *multicomponent Maxwell-Stefan diffusivities* through the expression:

$$\frac{x_i x_k}{D_{ik}} = -\omega_i\omega_k \frac{\sum_{j \neq i} (\mathrm{adj}B_i)_{jk}}{\sum_{j \neq i} \tilde{D}_{ij}(\mathrm{adj}B_i)_{jk}} \qquad (7\text{-}17)$$

where D_{ik} is the ik component of the multicomponent Maxwell-Stefan diffusivity matrix, and $(B_i)_{jk}$ is the jk component and adj$B$ the adjoint matrix of the matrix B_i. $(B_i)_{jk}$ is given by Equation 7-18:

$$(B_i)_{kj} = \tilde{D}_{kj} - \tilde{D}_{ij} \; i \neq j \tag{7-18}$$

Solving for Equation 7-17 leads to a number of algebraic expressions for each of the components in the multicomponent Fick diffusivity matrix in Equation 7-13. These are solved by COMSOL Multiphysics and implemented directly for two and three species' systems. As an example, the component D_{12} in a ternary system is given by:

$$\tilde{D}_{12} = \frac{\dfrac{\omega_1(\omega_2+\omega_3)}{x_1 D_{23}} + \dfrac{\omega_2(\omega_1+\omega_3)}{x_2 D_{13}} - \dfrac{\omega_3^2}{x_3 D_{12}}}{\dfrac{x_1}{D_{12}D_{13}} + \dfrac{x_2}{D_{12}D_{23}} + \dfrac{x_3}{D_{23}D_{13}}} \tag{7-19}$$

Expressions for the other matrix components of the ternary system, as well as those for the secondary system, are summarized in the Chemical Engineering Module Reference Guide.

For four components or more, the multicomponent Fick diffusivities are obtained numerically through matrix inversion. COMSOL Multiphysics starts with the multicomponent Maxwell-Stefan diffusivity matrix D to compute the multicomponent Fick diffusivity matrix, $\tilde{D}$, using the following equation:

$$D_{ij} = N_{ij} - g \tag{7-20}$$

where ij are indices in the matrices $\tilde{D}$ and N range from 1 to the number of components, n.

The elements of the matrix N in Equation 7-20 are defined as:

$$N_{ij} = (P^{-1})_{ij} \tag{7-21}$$

where P^{-1} is the inverse of a matrix P. The matrix P is defined as:

$$P_{ij} = \frac{\omega_i \omega_j}{g} - \tilde{C}_{ij} \tag{7-22}$$

where the matrix $\tilde{C}$ is defined as:

$$\tilde{C}_{ij} = \begin{cases} \dfrac{x_i x_j}{D_{ij}} & i \neq j \\ -\sum\limits_{k \neq j} \tilde{C}_{ik} & i = j \end{cases} \tag{7-23}$$

The term g in Equation 7-20 is a scalar value that provides numerical stability and should be of the same order of magnitude as the multicomponent Maxwell-Stefan diffusion coefficients. The application mode therefore defines g as the sum of the multicomponent Maxwell-Stefan diffusion coefficients:

$$g = \sum_{i=1}^{n-1}\left(\sum_{j=i+1}^{n} D_{ij}\right) \tag{7-24}$$

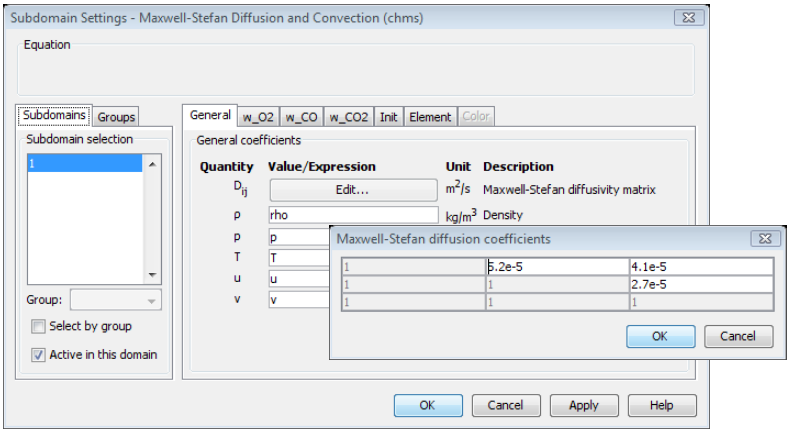
This definition of g works for most modeling. In rare cases, it might be necessary to change the value to obtain convergence. You can adjust the value for g by changing its expression on the **Variables** tab in the **Subdomain Settings - Equation System** dialog box. To open this dialog box, from the **Physics** menu, point to **Equation System** and then choose **Subdomain Settings**.

The multicomponent Fick diffusivity matrix is symmetric, which means that it is sufficient to apply Equation 7-20 to the upper triangle elements of $\tilde{D}$. The remaining elements are obtained by swapping the indices in the matrix, that is:

$$\tilde{D}_{lk} = \tilde{D}_{kl} \tag{7-25}$$

For low-density gas mixtures, the multicomponent Maxwell-Stefan diffusivities, D_{ij}, can be replaced with the binary diffusivities, $\hat{D}_{ij}$, for all pairs of species.

You enter the binary diffusivities into the Maxwell-Stefan diffusivity matrix in the application mode’s **Subdomain Settings** dialog box.



□ Select by group

☑ Active in this domain

Quantity	Value/Expression	Unit	Description
D_{ij}	Edit...	m^2/s	Maxwell-Stefan diffusivity matrix
ρ	rho	kg/m^3	Density
p	p		
T	T		
u	u		
v	v		

1	5.2e-5	4.1e-5
1	1	2.7e-5
1	1	1

Figure 7-3: Enter the binary diffusion coefficients in the Maxwell-Stefan diffusivity matrix.

NONCONSERVATIVE FORM

In COMSOL Multiphysics, you can select to use the conservative form of the mass transport equation (Equation 7-13) or the corresponding nonconservative formulation. In the nonconservative form, the convective term appears outside of the divergence operator on the right-hand side of the equation

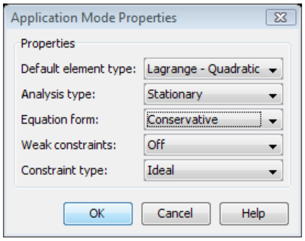
$$\frac{\partial}{\partial t}\rho\omega_i + \nabla\cdot\left[-\rho\omega_i\sum_{j=1}^{N}\tilde{D}_{ij}\left\{\frac{M}{M_j}\left(\nabla\omega_j + \omega_j\frac{\nabla M}{M}\right) + (x_j - \omega_j)\frac{\nabla p}{p}\right\}\right. \tag{7-26}$$
$$\left. -D_i^T\frac{\nabla T}{T}\right] = R_i - (\rho\mathbf{u}\cdot\nabla\omega_i)$$

This is valid for systems where the density is constant or very nearly so. For such incompressible fluids the continuity equation below is satisfied by:

$$\nabla\cdot\mathbf{u} = 0 \tag{7-27}$$

The nonconservative formulation provides additional stability to systems coupled to a momentum balance.

Note that Equation 7-26 is set as the default formulation when working with the Maxwell-Stefan Diffusion and Convection application mode. Make sure to change the formulation to the conservative form (Equation 7-13) in situations where mass change occurs in the subdomain, for example, in the case of condensation in a porous media. You are can change the formulation of the mass transport equation in the **Application Mode Properties** dialog box.


Application Mode Properties

Properties

Default element type: Lagrange - Quadratic

Analysis type: Stationary

Equation form: Conservative

Weak constraints: Off

Constraint type: Ideal

Figure 7-4: You can change between the conservative and the nonconservative formulation in the Application Mode Properties dialog box.

Boundary Conditions

The boundary conditions for the Maxwell-Stefan Diffusion and Convection application mode are analogous to the Convection and Diffusion application mode for Fickian diffusion.

The Flux boundary condition is:

$$\mathbf{n}_i \cdot \mathbf{n} = f_0(\omega_i) \tag{7-28}$$

where $\mathbf{n}_i$ is the mass flux vector for species *i,* and $\mathbf{n}$ is the normal vector to the boundary. The function f_0 can describe a heterogeneous reaction or a separation process at the boundary and can be an arbitrary function of ω_i, temperature, pressure, or even electric potential.

The Insulation/Symmetry condition has the source term in Equation 7-28 set to zero:

$$\mathbf{n}_i \cdot \mathbf{n} = 0 \tag{7-29}$$

The application mode also allows you to set a Mass fraction boundary condition:

$$\omega_i = \omega_{i,0} \tag{7-30}$$

You can set all mass fractions, bar one, because the sum of all fractions is equal to one.

The Convective flux boundary condition assumes that all mass passing through this boundary is convection-dominated. This firstly assumes that any mass flux due to diffusion across this boundary is zero:

$$\mathbf{n}\cdot\left(\left(-\rho\omega_i\sum_{j=1}^{n}\tilde{D}_{ij}\left(\nabla x_j+(x_j-\omega_j)\frac{\nabla p}{p}\right)\right)-D^T\frac{\nabla T}{T}\right) = 0 \tag{7-31}$$

so that:

$$\mathbf{n}_i\cdot\mathbf{n} = \omega_i\rho\mathbf{u}\cdot\mathbf{n} \tag{7-32}$$

This is a useful boundary condition in convection-dominated mass balances where the outlet concentration is unknown.

Pseudo 2D and Pseudo 3D Mass Balance Application Modes

There is an additional application mode (in 1D and 2D) for modeling stationary mass balances in the Maxwell-Stefan Diffusion and Convection application mode. This is suitable for applications where convection is constant and dominates exclusively in one direction so that diffusion can be completely neglected in that direction. This means that the modeled geometries can be described by extrusions of a cross section where the surface area does not change, for example, in ducts and tubular reactors.

Providing that the velocity profile can be expressed in the cross section and that the model is defined for steady state, you can replace the space variable along the direction of the flow with time, t. This implies that a time increment, at a given velocity, represents a displacement along the direction of the flow. This allows for the modeling of a 3D system in 2D, (or a 2D system in 1D) by letting time represent the third dimension, for example, the z coordinate. The equation for this type of mass balance is the following:

$$u_{\mathrm{dl}}\rho\frac{\partial\omega_i}{\partial t}+\nabla\cdot\left(\left(-\rho\omega_i\sum_{j=1}^{n}\tilde{D}_{ij}\left(\nabla x_j+(x_j-\omega_j)\frac{\nabla p}{p}\right)\right)-D^T\frac{\nabla T}{T}\right) = R_i \tag{7-33}$$

where u_{dl} is the function that gives the velocity profile (m/s). The convective term is now included in the first term of the equation, and time stepping automatically implies a change in position in the modeled domain. Use of the Pseudo Maxwell-Stefan Diffusion and Convection application modes reduces problem size and computational time. The application mode boundary conditions include those given in Equation 7-28, Equation 7-29, and Equation 7-30, while the Convective flux conditions (Equation 7-32) is excluded.

Example Model

The model "Stefan Tube" on page 164 of the *Chemical Engineering Module Model Library* is a simple example of how to use the Maxwell-Stefan Diffusion and Convection application mode in the Chemical Engineering Module. It gets you acquainted with the different menus and operations.

References

1. C.F. Curtiss and R.B. Bird, *Ind. Eng. Chem. Res.*, vol. 38, 1999, p. 2515.

2. R.B. Bird, W.E. Stewart, and E.N. Lightfoot, *Transport Phenomena*, Second edition, John Wiley & Sons, 2005.

Nernst-Planck Equations

Ionic solutions subjected to electric fields include migration transport to the existing diffusion and convection that is also possible. Migration implies that positive ions migrate from a positive potential to a negative potential along the direction of the electric field and vice versa for negatively charged ions. The flux of the ionic species *i* in the solution is given by the mass flux vector (mol/(m^2·s))

$$\mathbf{N}_i = -D_i\nabla c_i - z_i u_{\mathrm{m}i} F c_i \nabla V + c_i\mathbf{u} \tag{7-34}$$

where c_i denotes the concentration (mol/ m^3), D_i is the diffusion coefficient (m^2/s), $\mathbf{u}$ gives the velocity (m/s), F refers to Faraday's constant (s·A/mol), V denotes the electric potential (V), z_i is the charge number of the ionic species, and $u_{\mathrm{m}i}$ gives the ionic mobility (s·mol/kg).

The Chemical Engineering Module includes three application modes for this type of transport. The first accounts for the transport of every charged species and has to be solved in combination with the electroneutrality condition—the Nernst-Planck application mode. The second and third application modes are based upon the assumption that the contribution of the modeled species to the total transport of current is negligible—the Nernst-Planck without Electroneutrality and the Electrokinetic Flow application modes.

The Nernst-Planck Application Mode

SUBDOMAIN EQUATIONS

The Nernst-Planck application mode for transport by convection, diffusion, and migration treats the transport and mass balances of each dissolved species through the following equation:

$$\frac{\partial c_i}{\partial t} + \nabla\cdot(-D_i\nabla c_i - z_i u_{\mathrm{m}i} F c_i \nabla V + c_i\mathbf{u}) = R_i \tag{7-35}$$

where R_i denotes the reaction term (mol/ (m^3 ·s)). The velocity, $\mathbf{u}$, is equal to the velocity of the solvent. This implies that the solute's contribution to the solvent's velocity, through shear or any other forces, is negligible in comparison to the solvent's contribution to the solute. The velocity can be given by an analytical expression or be obtained by coupling a momentum balance. The above equation introduces one variable for the concentration of each of the dissolved species and the electric potential, ϕ. This implies that there is one more unknown than equations. The last unknown can be eliminated by the electroneutrality condition, which states that the net charge in every control volume in the solution is zero:

$$\sum z_i c_i = 0 \tag{7-36}$$

Continue by creating the following linear combination of the n mass balance equations, where each mass balance equation is weighted with the factor Fz_i:

$$F\sum_{i=1}^{n} z_i \frac{\partial c_i}{\partial t} + \nabla \cdot \left(F\sum_{i=1}^{n} z_i \mathbf{N}_i \right) = F\sum_{i=1}^{n} z_i R_i \tag{7-37}$$

where $\mathbf{N}_i$ is the flux vector defined in Equation 7-34. The first term in the equation above is zero, which can be shown by taking the time derivative of the electroneutrality condition. The expression under the divergence operator is identified as the total current density vector, defined by:

$$\mathbf{i} = F\sum_{i=1}^{n} z_i(-D_i \nabla c_i - z_i u_{\mathrm{m}i} F c_i \nabla V) \tag{7-38}$$

Note that no convective term is included in the expression for the current density, which is also a result of the electroneutrality condition.

It is now possible to rewrite Equation 7-37 as:

$$\nabla \cdot \mathbf{i} = F\sum_{i=1}^{n} z_i R_i \tag{7-39}$$

This equation states the conservation of electric charge. In the Nernst-Planck application mode, this equation replaces the mass balance equation for the first species.

The software always calculates the concentration of the first specified species through the electroneutrality condition. This means that you cannot specify boundary conditions for this species, although it can take part in the boundary condition descriptions for the current density. Most often, the species chosen for the first concentration variable is the oppositely charged ion from a supporting electrolyte.

Equation 7-35 and Equation 7-39 above are sufficient for describing the potential and concentration distribution in an electrochemical cell or in an electrolyte subjected to an electric field.

A useful observation from Equation 7-38 is that the ionic conductivity, defined in absence of concentration gradients, is implicitly given by:

$$\kappa = F^2 \sum_{i=1}^{n} z_i^2 c_i u_{\mathrm{m}i}$$

Furthermore, the potential gradient caused by the presence of a concentration gradient under situations with zero current becomes:

$$\nabla V = \frac{\sum_{i=1}^{n} (-D_i \nabla c_i)}{\sum_{i=1}^{n} z_i u_{\mathrm{m}i} F c_i} \qquad (7\text{-}40)$$

In the field of electrochemistry, this is known as the junction potential.

BOUNDARY CONDITIONS

In order to describe a system completely, a set of boundary conditions for the electric field and concentration fields is available in the application mode. You can set many different combinations of boundary conditions for the electric field and concentration fields on a boundary. However, not all combinations are relevant.

The first boundary condition for the electric field is the Current inflow condition:

$$-\mathbf{n} \cdot \mathbf{i} = i_0 \qquad (7\text{-}41)$$

where $\mathbf{n}$ denotes the normal vector to the boundary and i_0 is the inward current density.

The Current inflow boundary condition makes it possible to set the total current density at a boundary. It is common to combine this condition with concentration conditions for the species. Such a combination is useful in cases where the flow in an electrochemical cell is vigorous, and where the whole cell is not treated in a model, most often when the opposite electrode is a long way away. “Half-cell” investigations are very applicable for this type of condition.

The Electric insulation condition is simply the condition above with i_0 set to zero:

$$-\mathbf{n}\cdot\mathbf{i} = 0 \qquad (7\text{-}42)$$

Equation 7-42 also applies to Axial symmetry boundary condition when working in axisymmetric space dimensions. In COMSOL Multiphysics, this boundary condition is denoted with:

$$r = 0 \qquad (7\text{-}43)$$

The Potential boundary condition sets the electric potential on the boundary:

$$V = V_0 \qquad (7\text{-}44)$$

For each species, except for the first, the Flux boundary condition sets the flux of a given species at the surface of an electrode:

$$-\mathbf{n}\cdot\mathbf{N}_i = f(c, V) \qquad (7\text{-}45)$$

where **n** denotes the normal unit vector to the boundary. The function *f* can be any function of concentration and potential (and indeed temperature if an energy balance is coupled) usually given by a kinetic expression. In electrochemistry, this function usually depends on concentration and exponentially on the potential through either the Tafel equation or the Butler-Volmer equation.

The Insulation/Symmetry boundary condition is simply:

$$-\mathbf{n}\cdot\mathbf{N}_i = 0 \qquad (7\text{-}46)$$

The same equation also applies to Axial symmetry boundary condition when working in axisymmetric space dimensions. In COMSOL Multiphysics, this boundary condition is denoted with:

$$r = 0 \qquad (7\text{-}47)$$

The Concentration condition specifies the concentration of a species:

$$c_i = c_{i,0} \qquad (7\text{-}48)$$

The Convective flux boundary condition assumes that the mass flow through the boundary is convection dominated. This firstly assumes that any mass flux due to diffusion or migration across this boundary is zero

$$\mathbf{n}\cdot(-D_i\nabla c_i - z_i u_{\mathrm{m}i} F c_i \nabla V) = 0 \qquad (7\text{-}49)$$

so that

$$\mathbf{N}_i \cdot \mathbf{n} = c_i \mathbf{u} \cdot \mathbf{n} \tag{7-50}$$

Use of the Nernst-Planck application mode is appropriate in the modeling of electrochemical cells and corrosion processes. The application mode offers a straightforward way to perform tertiary current distribution analysis.

The Nernst-Planck without Electroneutrality Application Mode

The implementation of the Nernst-Planck without Electroneutrality application mode is identical to the Electrokinetic Flow application mode, which the next section describes.

The Electrokinetic Flow Application Mode

The Electrokinetic Flow application mode treats, in addition to transport through convection and diffusion, ionic species migration transport. It assumes that the investigated species are present in very low concentrations in an otherwise moderately concentrated electrolyte.

The application mode builds on the premise that the moderately concentrated electrolyte (often called a supporting electrolyte, buffer solution, or carrier electrolyte) is responsible for electric current transfer. This, in turn, gives the electric potential field for the ionic transport in the investigated species' mass balance.

SUBDOMAIN EQUATIONS

Assuming that current transport is carried out by the species that are present in moderate concentrations and that the concentration gradients of these species are small, the mass flux vectors for the dominating species reduce to

$$\mathbf{N}_i = -z_i u_{\mathrm{m}i} F c_i \nabla V + c_i \mathbf{u} \tag{7-51}$$

The current density is given by Faraday's law

$$\mathbf{i} = -F \sum_i z_i \mathbf{N}_i \tag{7-52}$$

which—when combined with the electroneutrality condition that eliminates the convective contribution in Equation 7-51— yields

$$\mathbf{i} = -F\sum_i -z_i^2 u_{\mathrm{m}i} F c_i \nabla V \tag{7-53}$$

where the summation runs over the dominating species.

The fact that current cannot be transported by convection is rather intuitive; if it could, it would be possible to transfer electric current by pumping. Equation 7-53 is simply Ohm's law for ionic current transport and can be simplified to the equation

$$\mathbf{i} = -\kappa \nabla V \tag{7-54}$$

where κ is the conductivity of the representative electrolyte. A current balance gives the current and potential density in the cell

$$\nabla \cdot \mathbf{i} = 0 \tag{7-55}$$

which in combination with Equation 7-54 yields:

$$\nabla \cdot (-\kappa \nabla V) = 0 \tag{7-56}$$

You can easily solve Equation 7-56 using the Conductive Media DC application mode in COMSOL Multiphysics. The potential distribution from the solution of this equation gives the migration term for the investigated species present at low concentrations. For these species, the mass balance is given by the full transport and mass balance equations:

$$\frac{\partial c_i}{\partial t} + \nabla \cdot (-D_i \nabla c_i - z_i u_{\mathrm{m}i} F c_i \nabla V + c_i \mathbf{u}) = R_i \tag{7-57}$$

The velocity, **u**, can be given by a coupled momentum balance or by a specified function of the space variables x, y, and z. The potential is given by coupling Equation 7-56 to the equation system.

Equation 7-57 is defined in the Electrokinetic Flow application mode in the Chemical Engineering Module. Unlike the Nernst-Planck application mode, the Electrokinetic Flow application mode does not solve the electroneutrality condition for this type of transport; instead, electroneutrality is inherently included in the current balance.

BOUNDARY CONDITIONS

In combination with proper boundary conditions, Equation 7-56 and Equation 7-57 give the current, potential, and mass distributions in electrolytic systems. You find the appropriate equation definitions and boundary conditions for the Conductive Media DC application mode in the *COMSOL Multiphysics Modeling Guide*.

The Flux boundary condition for the mass-balance equation states that the flux of the diluted species is known and given by a function, *f*:

$$\mathbf{N}_i \cdot \mathbf{n} = f(c, V) \tag{7-58}$$

This function can be a function of the concentration and the potential. The Insulation/Symmetry boundary condition is expressed as

$$\mathbf{N}_i \cdot \mathbf{n} = 0 \tag{7-59}$$

The Concentration boundary condition sets the composition at a given boundary

$$c_i = c_{i,0} \tag{7-60}$$

where the specified concentration is for that of the studied species. The total composition of the electrolyte is included in the expression for conductivity in the current balance equations.

The Convective flux boundary condition assumes that the mass flow through the boundary is convection dominated. This firstly assumes that any mass flux due to diffusion and migration across this boundary is zero

$$\mathbf{n} \cdot (-D_i \nabla c_i - z_i u_{\mathrm{m}i} F c_i \nabla V) = 0 \tag{7-61}$$

so that

$$\mathbf{N}_i \cdot \mathbf{n} = c_i \mathbf{u} \cdot \mathbf{n} \tag{7-62}$$

BOUNDARY CONDITIONS FOR INTERIOR BOUNDARIES

The following boundary conditions are available on pair boundaries in assemblies:

You can use the Thin boundary layer condition to model a thin layer of a material with a small diffusion coefficient compared to the adjacent domains:

$$\begin{aligned} \mathbf{n}_1 \cdot (-D\nabla c - z u_{\mathrm{m}} F c \nabla V + c\mathbf{u})_1 &= \frac{D}{d}(c_1 - c_2) \\ \mathbf{n}_2 \cdot (-D\nabla c - z u_{\mathrm{m}} F c \nabla V + c\mathbf{u})_2 &= \frac{D}{d}(c_2 - c_1) \end{aligned} \tag{7-63}$$

The layer has the thickness d and the diffusion coefficient D. This boundary condition is only available at the border between the parts in an assembly.

The Flux discontinuity boundary condition represents a discontinuity in the mass flux across an interior boundary or a border between parts in an assembly:

$$-\mathbf{n}\cdot(\mathbf{N}_1-\mathbf{N}_2) = N_0 \qquad (7\text{-}64)$$

It is not applicable to exterior boundaries.

Continuity is the default boundary condition on interior boundaries and pair boundaries; it is not applicable to exterior boundaries.

$$\mathbf{n}\cdot(\mathbf{N}_1-\mathbf{N}_2) = 0 \qquad (7\text{-}65)$$

Conservative and Nonconservative Formulations

When modeling mass balances in the Nernst-Planck and Electrokinetic Flow application modes of the Chemical Engineering Module, two mass balance formulations are available: a conservative and a nonconservative formulation. The conservative formulation is given by Equation 7-35.

The nonconservative formulation removes the convective term from the divergence operator in this equation, which gives more stability to systems coupled to a momentum balance. It is only applicable to systems that contain incompressible fluids. In these fluids, the following continuity equation is satisfied:

$$\nabla\cdot\mathbf{u} = 0 \qquad (7\text{-}66)$$

This condition means that the convective term in Equation 7-35 can be placed outside of the divergence operator, and on the right-hand side, see Equation 7-67:

$$\frac{\partial c_i}{\partial t}+\nabla\cdot(-D_i\nabla c_i-z_iu_{\mathrm{m}i}Fc_i\nabla V) = R_i-(\mathbf{u}\cdot\nabla c_i) \qquad (7\text{-}67)$$

Doing this provides stability because COMSOL Multiphysics is not required to additionally calculate a function similar to Equation 7-66 in the mass balance. Equation 7-67 is the default condition when working with the Nernst-Planck and Electrokinetic Flow application modes.

Pseudo 2D and Pseudo 3D Mass Balance Application Modes

There is an additional application mode for modeling stationary mass balances using the Nernst-Planck and Electrokinetic Flow application modes adapted to applications where convection is constant and dominates exclusively in one direction, so that the diffusion and migration terms can be completely neglected in that direction. It is then only possible to describe the modeled geometries by extrusions of a cross section. This approach is applicable to, for example, ducts and pipe tubular reactors where the surface area does not change.

Providing that the velocity profile can be expressed in the cross section and that the model is defined for steady state, you can replace the space variable along the direction of the flow with time, t. This implies that a time increment, at a given velocity, represent a displacement along the direction of the flow. This allows for the modeling of a 3D system in 2D, or a 2D system in 1D, by letting time represent the third dimension, for example, the z-coordinate. The equation for this type of mass balance is

$$u_{\mathrm{dl}}\frac{\partial c_i}{\partial t} + \nabla \cdot (-D_i \nabla c_i - z_i u_{\mathrm{m}i} F c_i \nabla V) = R_i \qquad (7\text{-}68)$$

where u_{dl} gives the velocity profile. The convective term is now included in the first term of the equation and time stepping automatically implies a change in position in the modeled domain. Use of these Pseudo Nernst-Planck and Pseudo Electrokinetic application modes reduces problem size and computational time.

For the Nernst-Planck pseudo application modes, the boundary conditions include those given in Equation 7-41, Equation 7-42, Equation 7-43, and Equation 7-44 for the current/potential field. For the mass balances, the boundary conditions include those given in Equation 7-45, Equation 7-46, Equation 7-47, and Equation 7-48.

For the Electrokinetic Flow pseudo application modes, the boundary conditions include those given in Equation 7-58, Equation 7-59, and Equation 7-60.

The Convective flux conditions (Equation 7-50 and Equation 7-61) are of course excluded from the Nernst-Planck and Electrokinetic Flow pseudo application modes.

The pseudo application modes are highly useful in parallel-plate electrochemical reactors, for example, the chlorate cell. In these cells, the inter-electrode distance is small compared to the height of the electrodes where flow takes place along this height. This implies that the potential lines are almost parallel to the electrodes, and that migration is small in the direction of the flow. The small distance between the plates indicates that most diffusion happens in this direction rather than in the direction of the flow.

Example Models

The sections “Transport in an Electrokinetic Valve” on page 348 and “Electroosmotic Flow in Porous Media” on page 389 of the Chemical Engineering Module Model Library show two examples that use the Electrokinetic Flow application mode.

8

Predefined Multiphysics Couplings

This chapter gives an overview of the Predefined Multiphysics Couplings in the Chemical Engineering Module. These couplings are available for non-isothermal flows, reacting flows, and rotating machinery.

Flow with Variable Density

In industrial applications it is common that the density of a process fluid varies. These variations can have a number of different sources but the most common one is the presence of an inhomogeneous temperature field. The Chemical Engineering Module includes the predefined multiphysics coupling Non-Isothermal Flow to simulate systems where density varies with temperature.

Other situations where the density might vary includes chemical reactions, for instance where reactants associate or dissociate. The Chemical Engineering Module provides the application mode Weakly Compressible Navier-Stokes, which can be used to simulate flows where the density varies for any reason.

Weakly Compressible Navier-Stokes

The Weakly Compressible Navier-Stokes application mode contains the fully compressible formulation of the continuity equation and momentum equations:

$$\begin{gathered}\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\mathbf{u}) = 0\\ \rho\frac{\partial\mathbf{u}}{\partial t} + \rho\mathbf{u}\cdot\nabla\mathbf{u} = -\nabla p + \nabla\cdot\left(\eta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \left(\frac{2}{3}\eta - \kappa_{\mathrm{dv}}\right)(\nabla\cdot\mathbf{u})\mathbf{I}\right) + \mathbf{F}\end{gathered} \tag{8-1}$$

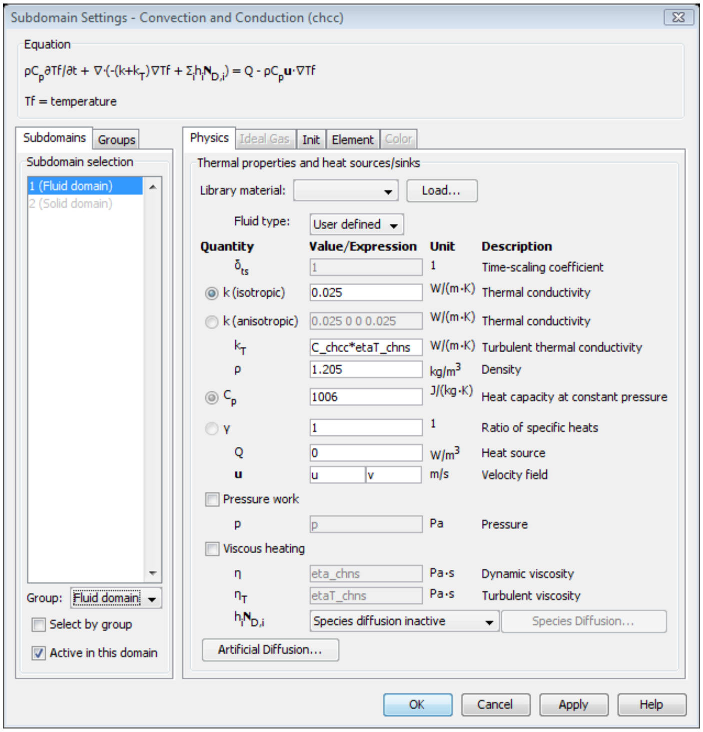
where

- ρ is the density (kg/m^3)
- $\mathbf{u}$ is the velocity vector (m/s)
- p is pressure (Pa)
- η is the dynamic viscosity (Pa·s)
- κ_{dv} is the dilatational viscosity (Pa·s)
- $\mathbf{F}$ is the body force vector (N/m^3)

The application mode is described in detail in the section “Weakly Compressible Navier-Stokes” on page 149.

Laminar Non-Isothermal Flow

The Non-Isothermal Flow predefined multiphysics coupling is a combination of two other application modes: Weakly Compressible Navier-Stokes and Convection and Conduction. The difference compared to loading these application modes separately is that the predefined groups **default** and **Solid domain** are available. This is illustrated in Figure 8-1, which shows the **Subdomain Settings** dialog box for the Convection and Conduction application mode. The **Group** list is in the lower-left corner.



Subdomain Settings - Convection and Conduction (chcc)

Equation

$\rho C_p \partial Tf/\partial t + \nabla\cdot(-(k+k_T)\nabla Tf + \Sigma_i h_i \mathbf{N}_{D,i}) = Q - \rho C_p \mathbf{u}\cdot\nabla Tf$

Tf = temperature

Subdomains | Groups

Subdomain selection

1 (Fluid domain)
2 (Solid domain)

Group: Fluid domain

□ Select by group

☑ Active in this domain

Physics | Ideal Gas | Init | Element | Color

Thermal properties and heat sources/sinks

Library material: Load...

Fluid type: User defined

Quantity	Value/Expression	Unit	Description
δ_{ts}	1	1	Time-scaling coefficient
● k (isotropic)	0.025	W/(m·K)	Thermal conductivity
○ k (anisotropic)	0.025 0 0 0.025	W/(m·K)	Thermal conductivity
k_T	C_chcc*etaT_chns	W/(m·K)	Turbulent thermal conductivity
ρ	1.205	kg/m^3	Density
● C_p	1006	J/(kg·K)	Heat capacity at constant pressure
○ γ	1	1	Ratio of specific heats
Q	0	W/m^3	Heat source
u	u v	m/s	Velocity field
□ Pressure work			
p	p	Pa	Pressure
□ Viscous heating			
η	eta_chns	Pa·s	Dynamic viscosity
η_T	etaT_chns	Pa·s	Turbulent viscosity
$h_i\mathbf{N}_{D,i}$	Species diffusion inactive	Species Diffusion...	

Artificial Diffusion...

OK | Cancel | Apply | Help

Figure 8-1: The Subdomain Settings dialog box for the Convection and Conduction application mode loaded via the Non-Isothermal Flow predefined multiphysics coupling. The velocities are predefined to be consistent with the selected group.

Use the **default** group for any fluid domain. The velocities are then taken from the Weakly Compressible Navier-Stokes application mode. If you change the **Fluid type** from **Liquid**, the subdomain loses the "default" label, but the couplings remain.

Use the **Solid domain** to model solid materials. Setting the velocities to zero then deactivates all artificial diffusions. This group also sets the **Fluid type** to **User defined**.

The groups **default** and **Solid domain** are available in the **Subdomain Settings** dialog box for the Weakly Compressible Navier-Stokes application mode as well. The effect is that the Navier-Stokes equations are active if the **default** group is selected while they are inactive (that is, not solved for) if the **Solid domain** group is selected.

Note that you have to set the proper group in both the Weakly Compressible Navier-Stokes and the Convection and Conduction application modes.

Turbulent Non-Isothermal Flow

This predefined multiphysics coupling is a combination of three other application modes: the Weakly Compressible Navier-Stokes and two Convection and Conduction application modes. The difference compared to loading these application modes separately is that the following predefined groups are available: **Fluid domain** and **Solid domain**. For each subdomain, select the proper group for each of the application modes.

Selecting **Solid domain** for each of the application modes in a subdomain has the following effects:

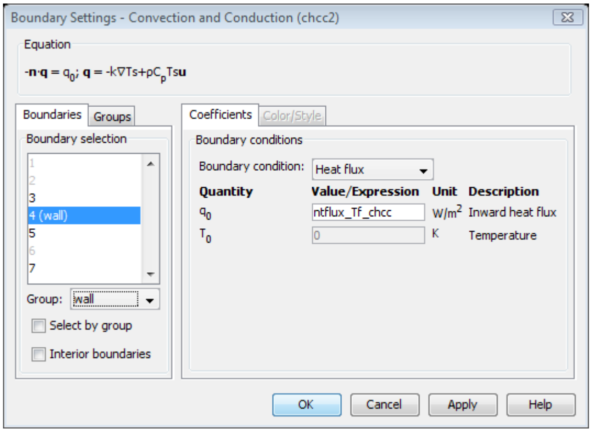
- The Convection and Conduction application mode 1 is deactivated.
- The velocities in Convection and Conduction application mode 2 are set to zero, giving a pure conduction equation.
- The Turbulence Model application mode is deactivated.

Selecting the **Fluid domain** for each of the application modes in a subdomain will have the following effects:

- The velocities in Convection and Conduction application mode 1 will be taken from the Turbulence Model application mode.
- The Convection and Conduction application mode 2 is deactivated.
- The density in the Turbulence Model application mode is taken from the active Convection and Conduction application mode.

The last point is to ensure that the ideal gas law applies to both the active application modes if you have selected **Ideal gas** from the **Fluid type** list in the Convection and Conduction application mode.

There is also a group, **wall**, available in the **Boundary Settings** dialog boxes of the Convection and Conduction application modes. This group setting for the Convection and Conduction application mode 2 appears in Figure 8-2. This group's settings apply to boundaries that separate a fluid domain from a solid domain. It helps you to set up the thermal wall function expression explained in the Chapter 6, “Energy Transport,” on page 195.



Boundary Settings - Convection and Conduction (chcc2)

Equation

$-\mathbf{n}\cdot\mathbf{q} = q_0$; $\mathbf{q} = -k\nabla Ts+\rho C_p Ts\mathbf{u}$

Boundary selection

1
2
3
4 (wall)
5
6
7

Group: wall

□ Select by group

□ Interior boundaries

Boundary conditions

Boundary condition: Heat flux

Quantity	Value/Expression	Unit	Description
q_0	ntflux_Tf_chcc	W/m^2	Inward heat flux
T_0	0	K	Temperature

Figure 8-2: The Boundary Settings dialog box for Convection and Conduction application mode 2 loaded via a turbulent Non-Isothermal Flow predefined multiphysics coupling. The Wall group helps you to set up the thermal wall function boundary condition.

Example Models

The model “Non-Isothermal Flow in a Cooling Process” on page 129 in the *Chemical Engineering Module Model Library* demonstrates the laminar Non-Isothermal Flow predefined multiphysics coupling. The model “Forced Turbulent Convection Cooling of a Hot Plate” on page 98 in the same book demonstrates the Non-Isothermal Flow application mode using the k-ε turbulence model.

Reference

1. L. Ignat, D. Pelletier, and F. Ilinca, “A universal formulation of two-equation models for adaptive computation of turbulent flows,” *Computer methods in applied mechanics and engineering*, vol. 189, pp. 1119–1139, 2000.

Fluid-Chemical Reactions Interaction

The predefined coupling Reacting Flow automatically couples an Incompressible Navier-Stokes application mode for calculation of fluid flow with a Convection and Diffusion application mode for mass transport. The velocity in the Convection and Diffusion application mode is automatically set to the velocity of the Incompressible Navier-Stokes mode.

Rotating Machinery

The Rotating Machinery predefined multiphysics coupling makes it possible to easily model moving rotating parts in, for example, stirred tanks, mixers, and pumps.

This combination of application modes formulates the incompressible Navier-Stokes equations in a rotating coordinate system. Parts that are not rotated are expressed in the fixed coordinate system. The predefined coupling then takes care of the necessary administration to join the parts of the model that use a rotating coordinate system with the parts that are fixed.

You can use this predefined multiphysics coupling in cases where the modeled device can be divided into rotationally invariant geometries. The desired operation can be, for example, to rotate an impeller in a baffled tank. This is illustrated in the following figure where the impeller rotates from position 1 to 2. The first step is to divide the geometry into two parts which are both rotationally invariant, as shown in Step 1a. Next specify the parts that should be modeled with a rotating frame and the ones that should be modeled in a fixed frame, shown in Step 1b in the following diagram. The application mode then automatically performs the coordinate transformation and the joining of the fixed and moving parts as in Step 2a.

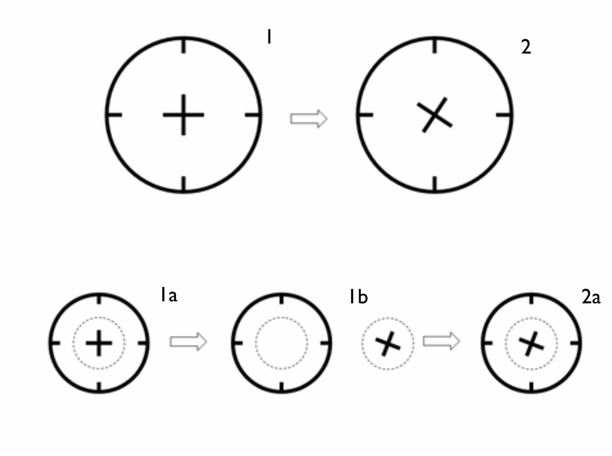


Figure 8-3: The modeling procedure in the rotating machinery application mode in the Chemical Engineering Module.

It is straightforward to use the Rotating Machinery predefined multiphysics coupling. First you draw the geometry using two separate subdomains for the fixed and rotating parts. Then you activate the assembly feature, which makes it possible to treat the two subdomains as two separate parts in an assembly. After this you specify which part uses a rotating frame. Once you have done this, you can proceed to the usual steps of setting the fluid properties, boundary conditions, and then mesh and solve the problem.

The Moving Mesh (ALE) application mode contains predefined groups for parts that are fixed (**fixed**) and parts that rotate clockwise (**rotate_CW**) and counterclockwise (**rotate_CCW**). In the Incompressible Navier-Stokes application mode you set up the fluid flow as usual, but there are separate predefined settings for no-slip conditions on the boundaries with clockwise rotation (**no_slip_CW**) and counterclockwise rotation (**no_slip_CCW**).

An example of how to build a model using the application mode is given in the section "Modeling of Rotating Parts" on page 227 in the *Chemical Engineering Module Model Library*.

9

The Materials/Coefficients Library

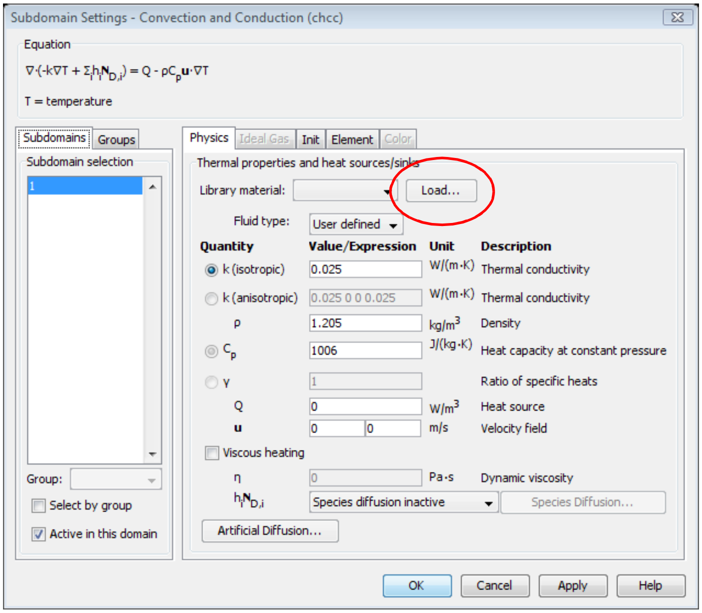
The Material/Coefficient Library contains an extensive set of mechanical and heat transfer properties for solid materials. In addition, it contains a limited set of fluid properties, which can be used mainly for the Momentum Transport and the Energy Transport application modes.

Material Libraries

A useful feature in COMSOL Multiphysics is the Materials/Coefficients library. The Chemical Engineering Module extends this library with two extra materials libraries: **Liquid and Gases**, with temperature-dependent fluid dynamic and thermal properties, and **MEMS Material Properties**, an extended solid materials library with metals, semiconductors, insulators, and polymers.

Loading Material Properties from the Library

For most application modes in COMSOL Multiphysics, you can load a material and its accompanying properties directly from within the **Subdomain Settings** dialog window; simply click the **Load** button as indicated in Figure 9-1.



Subdomain Settings - Convection and Conduction (chcc)

Equation

$\nabla\cdot(-k\nabla T + \Sigma_i h_i \mathbf{N}_{D,i}) = Q - \rho C_p \mathbf{u}\cdot\nabla T$

T = temperature

Subdomain selection: 1

Group:

□ Select by group

☑ Active in this domain

Thermal properties and heat sources/sinks

Library material:

Fluid type: User defined

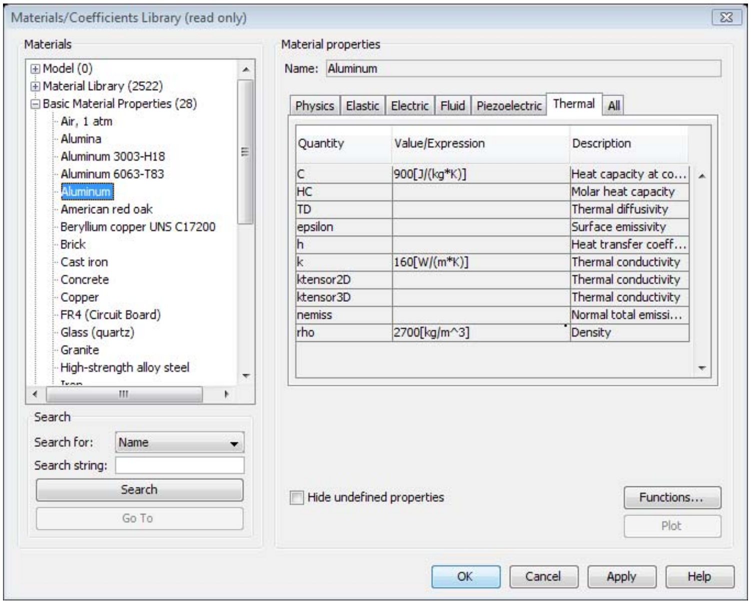
Quantity	Value/Expression	Unit	Description
● k (isotropic)	0.025	W/(m·K)	Thermal conductivity
○ k (anisotropic)	0.025 0 0 0.025	W/(m·K)	Thermal conductivity
ρ	1.205	kg/m^3	Density
○ C_p	1006	J/(kg·K)	Heat capacity at constant pressure
○ γ	1		Ratio of specific heats
Q	0	W/m^3	Heat source
u	0 0	m/s	Velocity field

□ Viscous heating

Quantity	Value/Expression	Unit	Description
η	0	Pa·s	Dynamic viscosity
$h_i \mathbf{N}_{D,i}$	Species diffusion inactive		

Figure 9-1: The Load button associated with library materials in the Subdomain Settings dialog box.

This action opens the **Materials/Coefficients Library** dialog box (see Figure 9-2). From there you can select a material to load.



Name: Aluminum

Quantity	Value/Expression	Description
C	900[J/(kg*K)]	Heat capacity at co...
HC		Molar heat capacity
TD		Thermal diffusivity
epsilon		Surface emissivity
h		Heat transfer coeff...
k	160[W/(m*K)]	Thermal conductivity
ktensor2D		Thermal conductivity
ktensor3D		Thermal conductivity
nemiss		Normal total emissi...
rho	2700[kg/m^3]	Density

Search for: Name

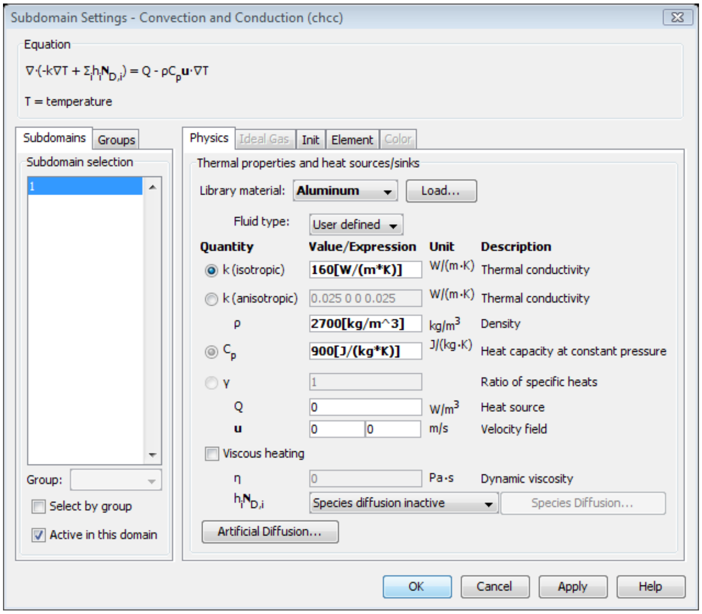
Search string:

□ Hide undefined properties

Figure 9-2: The Materials/Coefficients Library dialog box.

The **Materials** list on the left side contains the installed library folders plus a model folder that contains already-selected materials and coefficients. To load a material into the **Model** folder, select it from the **Materials** list and click **Apply** or **OK**; it now appears in the **Model** folder, which shows all the materials available for the model.

After clicking **OK**, the **Materials/Coefficients Library** dialog box closes and you return to the **Subdomain Settings** dialog box. Now the software has loaded the selected material properties into the corresponding edit fields of that dialog box. The edit fields that contain data taken from the materials library you just selected appear in bold (see Figure 9-3).



Subdomain Settings - Convection and Conduction (chcc)

Equation

$\nabla\cdot(-k\nabla T + \Sigma_i h_i \mathbf{N}_{D,i}) = Q - \rho C_p \mathbf{u}\cdot\nabla T$

T = temperature

Subdomains | Groups

Subdomain selection: 1

Group:

□ Select by group

☑ Active in this domain

Physics | Ideal Gas | Init | Element | Color

Thermal properties and heat sources/sinks

Library material: **Aluminum** Load...

Fluid type: User defined

Quantity	Value/Expression	Unit	Description
● k (isotropic)	**160[W/(m*K)]**	W/(m·K)	Thermal conductivity
○ k (anisotropic)	0.025 0 0 0.025	W/(m·K)	Thermal conductivity
ρ	**2700[kg/m^3]**	kg/m^3	Density
● C_p	**900[J/(kg*K)]**	J/(kg·K)	Heat capacity at constant pressure
○ γ	1		Ratio of specific heats
Q	0	W/m^3	Heat source
u	0 0	m/s	Velocity field

□ Viscous heating

η	0	Pa·s	Dynamic viscosity
$h_i \mathbf{N}_{D,i}$	Species diffusion inactive		Species Diffusion...

Figure 9-3: The Subdomain Settings dialog box after loading aluminum from the material library. Boldface characters indicate that the library material is active.

The Material Libraries

As noted earlier, the **Materials** list in the **Materials/Coefficients Library** dialog box contains multiple expandable folders. The list includes **Model**, **Basic Material Properties**, and additional libraries depending on the modules that your license includes.

Also, if your license includes the COMSOL Material Library, this also appears as a separate folder in the **Materials** list.

Note: The COMSOL Material Library is a separate add-on product that includes over 2500 materials and about 20,000 properties, most of which are temperature-dependent functions.

The Liquids and Gases Library and Material Property Functions

In many cases of modeling the material properties varies with the dependent variables describing the state (temperature, pressure, concentration, potential, stress, and so on). COMSOL Multiphysics is capable of describing material properties as functions of the dependent variables. The following discussion exemplifies use and access a library with material property functions (in this case Liquids and Gases), and also how to edit it and create your own material property functions.

The Liquids and Gases material library contains thermal and fluid dynamic properties for a set of common fluids. The properties are described as temperature-dependent functions. These functions are based on data collected from scientific publications. If you click the **Load** button in the **Subdomain Settings** dialog box of any application mode and select to expand **Liquids and Gases** you can choose from various fluids, both liquids or gases (as depicted in Figure 9-4).Their properties are described at atmospheric pressure conditions.

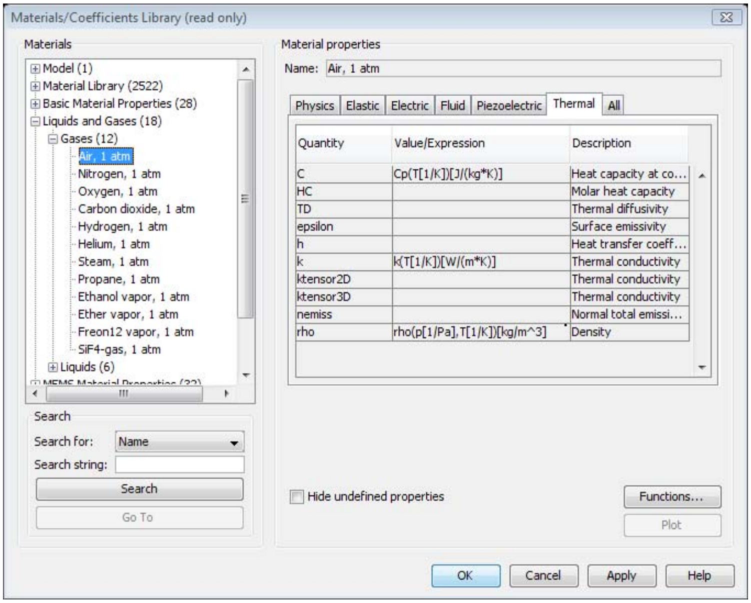
Using Material Property Functions

In many modeling situations, the material properties vary with the dependent variables describing the state (temperature, pressure, concentration, potential, stress, and so on). In COMSOL Multiphysics you can describe material properties as functions of the dependent variables. The following discussion exemplifies how to use a library with material property functions (in this case Liquids and Gases), and also how to edit it and create your own material property functions.

The Liquids and Gases material library contains thermal and fluid-dynamic properties for a set of common fluids. All properties are given as functions of temperature and at atmospheric pressure, except the density, which for gases is also a function of the local pressure.The functions are based on data collected from scientific publications.

Note: The data-fitted functions expect temperature and pressure arguments to be expressed in the SI units *kelvin* (K) and *pascal* (Pa), respectively, and return values in appropriate SI units. Unit expressions are automatically inserted to handle the conversions to and from the model's base unit system.

If you click the **Load** button in the **Subdomain Settings** dialog box of any application mode and select to expand **Liquids and Gases** you can choose from various fluids, both liquids and gases (see Figure 9-4).



Materials/Coefficients Library (read only)

Materials

- Model (1)
- Material Library (2522)
- Basic Material Properties (28)
- Liquids and Gases (18)
  - Gases (12)
    - Air, 1 atm
    - Nitrogen, 1 atm
    - Oxygen, 1 atm
    - Carbon dioxide, 1 atm
    - Hydrogen, 1 atm
    - Helium, 1 atm
    - Steam, 1 atm
    - Propane, 1 atm
    - Ethanol vapor, 1 atm
    - Ether vapor, 1 atm
    - Freon12 vapor, 1 atm
    - SiF4-gas, 1 atm
  - Liquids (6)

Search

Search for: Name

Search string:

Material properties

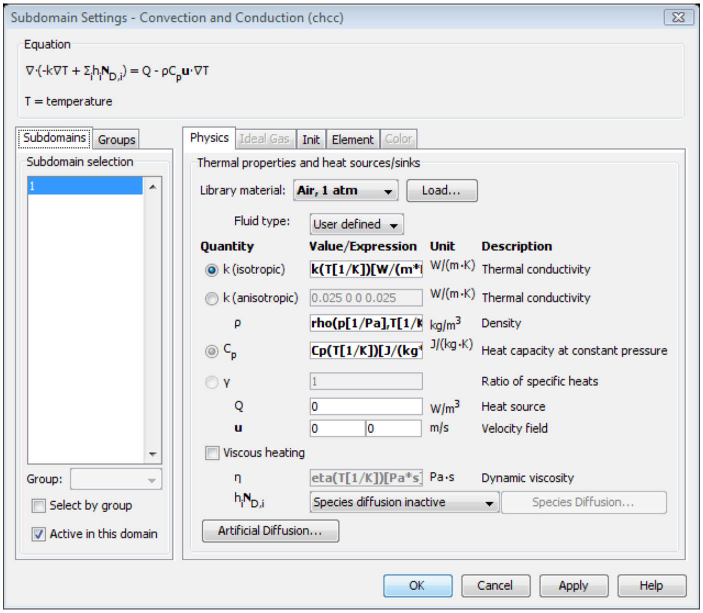
Name: Air, 1 atm

Quantity	Value/Expression	Description
C	Cp(T[1/K])[J/(kg*K)]	Heat capacity at co...
HC		Molar heat capacity
TD		Thermal diffusivity
epsilon		Surface emissivity
h		Heat transfer coeff...
k	k(T[1/K])[W/(m*K)]	Thermal conductivity
ktensor2D		Thermal conductivity
ktensor3D		Thermal conductivity
nemiss		Normal total emissi...
rho	rho(p[1/Pa],T[1/K])[kg/m^3]	Density

□ Hide undefined properties

Figure 9-4: The Materials/Coefficients Library dialog box, with Liquids and Gases expanded.

If you load a material that uses a function such as those in the Liquids and Gases library, the **Subdomain Settings** dialog box looks like Figure 9-5, with function calls in the edit fields for the material properties.



Subdomain Settings - Convection and Conduction (chcc)

Equation

$\nabla\cdot(-k\nabla T + \Sigma h_i \mathbf{N}_{D,i}) = Q - \rho C_p \mathbf{u}\cdot\nabla T$

T = temperature

Subdomains | Groups

Subdomain selection: 1

Group:

☐ Select by group

☑ Active in this domain

Physics | Ideal Gas | Init | Element | Color

Thermal properties and heat sources/sinks

Library material: Air, 1 atm

Fluid type: User defined

Quantity	Value/Expression	Unit	Description
● k (isotropic)	k(T[1/K])[W/(m*l	W/(m·K)	Thermal conductivity
○ k (anisotropic)	0.025 0 0 0.025	W/(m·K)	Thermal conductivity
ρ	rho(p[1/Pa],T[1/K	kg/m^3	Density
○ C_p	Cp(T[1/K])[J/(kg'	J/(kg·K)	Heat capacity at constant pressure
○ γ	1		Ratio of specific heats
Q	0	W/m^3	Heat source
u	0 0	m/s	Velocity field

☐ Viscous heating

η	eta(T[1/K])[Pa*s]	Pa·s	Dynamic viscosity
$h_i\mathbf{N}_{D,i}$	Species diffusion inactive		

Figure 9-5: The Subdomain Settings dialog box after loading a material that uses function calls, for example, the materials from the fluid library.

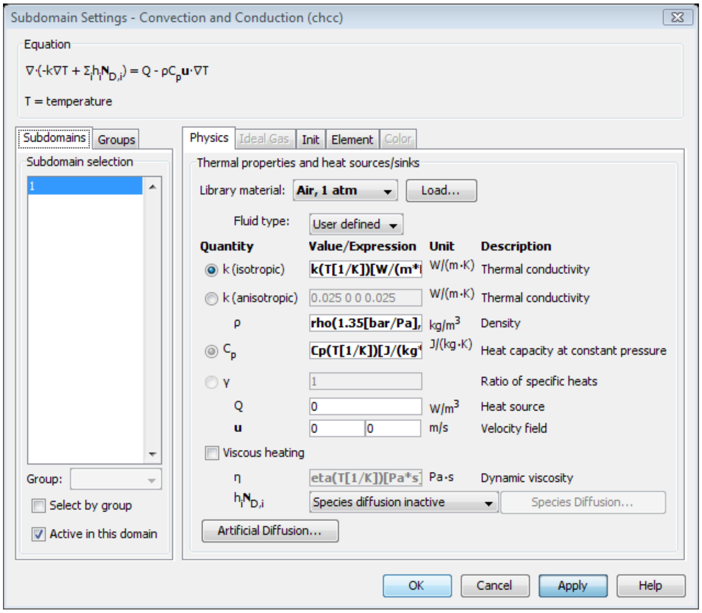
In this example, the software specifies the material property for density with the function call

```
rho(p[1/Pa],T[1/K])[kg/m^3]
```

which is a function call to the material loaded, in this case **Air**. The function uses two inputs: pressure, `p`, and temperature, `T`. The default settings are based on the assumption that the temperature variable in the model is `T` and that there is a pressure variable named `p`. Being dependent variables, these are expressed in the model's selected base unit system. The unit expressions inside the function calls convert the values from the model's unit system to nondimensional numbers corresponding to SI units, while the expression between the last brackets makes sure the returned SI value is interpreted correctly in the model's unit system.

In many cases you must change these function inputs. For example, if you model only heat transfer, there is no variable for pressure. In that case you must either specify the pressure directly in the function input or set up a constant or expression for the variable p. This constant or expression variable must have the dimension of pressure, which you achieve by adding a pressure unit to the expression, such as `135[kPa]`.

It is easy to alter the function input values and variables: simply click inside the parentheses delimiting the function argument and replace the default symbol with the desired value or variable. In the following figure you can see such a modification where a numerical value replaces the pressure variable.



Subdomain Settings - Convection and Conduction (chcc)

Equation

$\nabla\cdot(-k\nabla T + \Sigma_i h_i \mathbf{N}_{D,i}) = Q - \rho C_p \mathbf{u}\cdot\nabla T$

T = temperature

Subdomain selection: 1

Group:

☐ Select by group

☑ Active in this domain

Thermal properties and heat sources/sinks

Library material: Air, 1 atm

Fluid type: User defined

Quantity	Value/Expression	Unit	Description
● k (isotropic)	k(T[1/K])[W/(m*I	W/(m·K)	Thermal conductivity
○ k (anisotropic)	0.025 0 0 0.025	W/(m·K)	Thermal conductivity
ρ	rho(1.35[bar/Pa],	kg/m^3	Density
○ C_p	Cp(T[1/K])[J/(kg*	J/(kg·K)	Heat capacity at constant pressure
○ γ	1		Ratio of specific heats
Q	0	W/m^3	Heat source
u	0 0	m/s	Velocity field

☐ Viscous heating

η	eta(T[1/K])[Pa*s]	Pa·s	Dynamic viscosity
$h_i\mathbf{N}_{D,i}$	Species diffusion inactive		

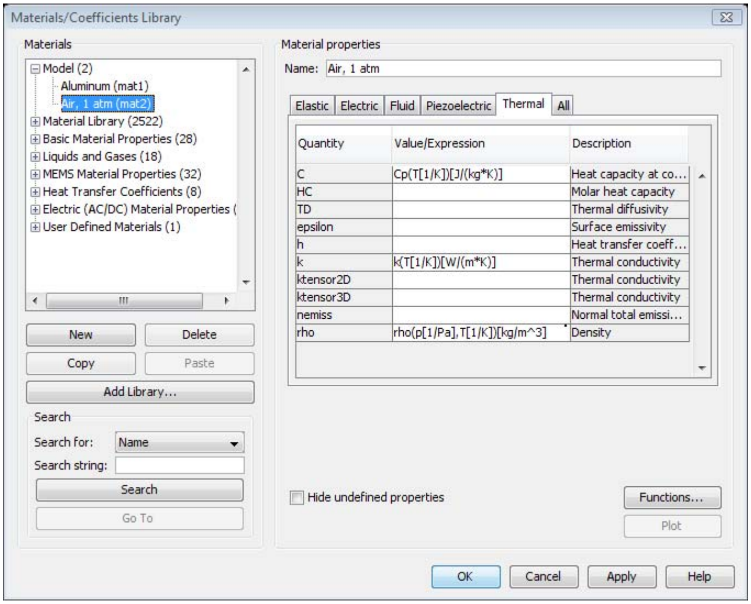
Figure 9-6: Modifying the function inputs. Notice how the unit syntax can convert argument values from any pressure unit to pascal.

In this case the function output is the density for air at the specified pressure and at the temperature given by the variable T.

Note: If you override one property defined by a library material by changing anything outside the function arguments, all other property expressions loose their connection to the material library. In particular, material functions appearing in other edit fields (now in plain text instead of in bold face) stop working. The proper way to edit one property of a loaded material is to change the material functions, as described later in this section, rather than editing the edit field in the **Subdomain Settings** dialog box.

USING MATERIAL PROPERTY FUNCTION CALLS OUTSIDE THE SUBDOMAIN SETTINGS

You can also use a library material property function in a model in places other than the **Subdomain Settings** dialog box. One example might be to combine several properties in an expression in the **Scalar Expressions** dialog box. To do so, you must first load the library material into the model using the **Materials/Coefficients Library** dialog box. This dialog box opens either, as described above, from the **Subdomain Settings** dialog box (then in read only mode), or directly from the **Options** menu. The currently loaded materials and their assigned names are listed in the **Model** folder in the **Materials** tree. (Figure 9-7).



Search for: Name

Name: Air, 1 atm

Quantity	Value/Expression	Description
C	Cp(T[1/K])[J/(kg*K)]	Heat capacity at co...
HC		Molar heat capacity
TD		Thermal diffusivity
epsilon		Surface emissivity
h		Heat transfer coeff...
k	k(T[1/K])[W/(m*K)]	Thermal conductivity
ktensor2D		Thermal conductivity
ktensor3D		Thermal conductivity
nemiss		Normal total emissi...
rho	rho(p[1/Pa],T[1/K])[kg/m^3]	Density

□ Hide undefined properties

Figure 9-7: Viewing the materials loaded into a model along with their names in COMSOL Multiphysics.

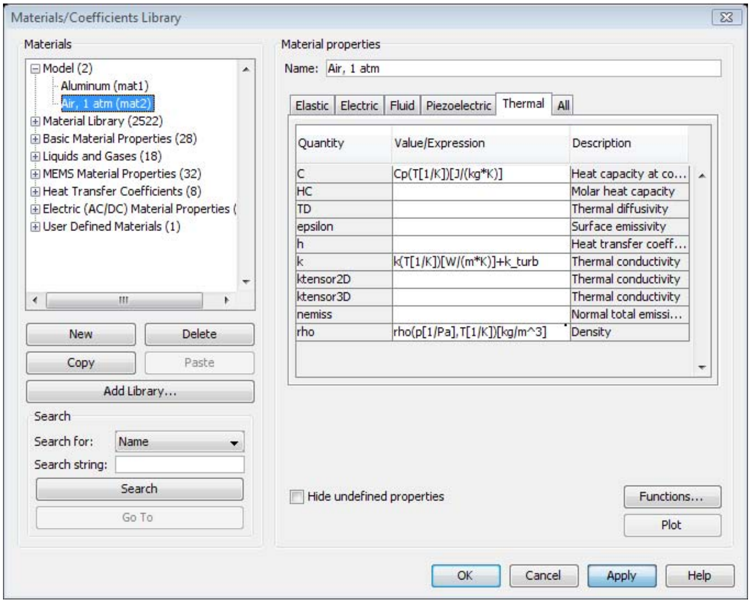
In the above figure, two materials are already loaded into the model. COMSOL Multiphysics allocates local material names on the form `matX`, where `X` is a running number assigned in the order in which materials were loaded into the model. In this case, the user loaded **Aluminum** first, so it takes the name `mat1`.

To use a particular material-property function, you can start by copying the syntax shown in the **Value/Expression** column in the **Materials/Coefficients Library** dialog box. Then you must add `matX_` in front of the function call. Thus, referring to Figure 9-7, to evaluate the function for thermal conductivity of air at 350 K, the syntax is

```
mat2_k(350)[W/m*K]
```

Editing Material Properties

To change a property of a loaded material, choose **Materials/Coefficients Library** from the **Options** menu. This opens the dialog box in edit mode. Then select the desired material in the **Materials** list, click the **Value/Expression** field of interest, and change the expression as depicted in Figure 9-8. In this example the user is adding `k_turb` to the original function for thermal conductivity, where `k_turb` is a variable that must be defined elsewhere in the model.



Search for: Name

Search string:

Name: Air, 1 atm

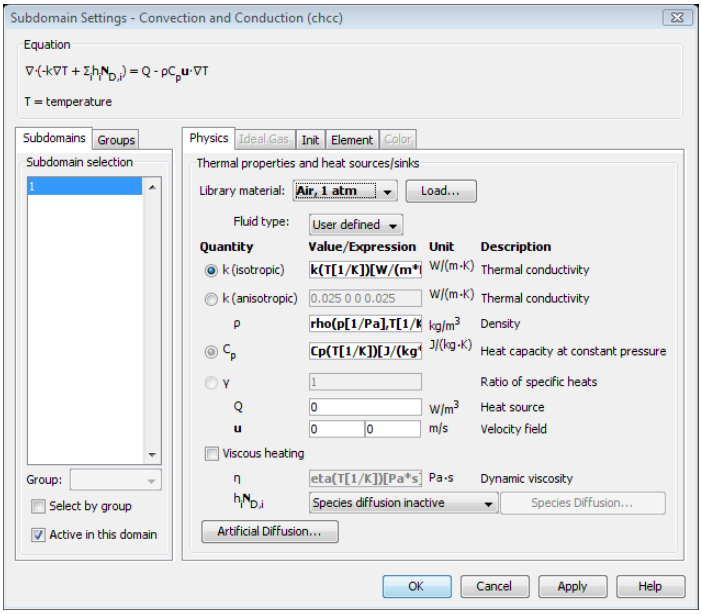
Quantity	Value/Expression	Description
C	Cp(T[1/K])[J/(kg*K)]	Heat capacity at co...
HC		Molar heat capacity
TD		Thermal diffusivity
epsilon		Surface emissivity
h		Heat transfer coeff...
k	k(T[1/K])[W/(m*K)]+k_turb	Thermal conductivity
ktensor2D		Thermal conductivity
ktensor3D		Thermal conductivity
nemiss		Normal total emissi...
rho	rho(p[1/Pa],T[1/K])[kg/m^3]	Density

☐ Hide undefined properties

Figure 9-8: Editing a material property.

If you want to edit the function describing some material property, click the **Functions** button. This opens the **Functions** dialog box, where you can view and edit any function describing the material.

After changing a material property in this way, you need to reload all subdomain settings that use the material; otherwise the function call will not work. To reload the material, simply go to **Physics>Subdomain Settings** and select the modified material from the **Library material** list. The new expression then shows up in the edit field for the corresponding material property in the **Subdomain Settings** dialog box (see Figure 9-9).



Subdomain Settings - Convection and Conduction (chcc)

Equation

$\nabla\cdot(-k\nabla T + \Sigma_i h_i \mathbf{N}_{D,i}) = Q - \rho C_p \mathbf{u}\cdot\nabla T$

T = temperature

Subdomain selection: 1

Group:

☐ Select by group

☑ Active in this domain

Thermal properties and heat sources/sinks

Library material: Air, 1 atm

Fluid type: User defined

Quantity	Value/Expression	Unit	Description
● k (isotropic)	k(T[1/K])[W/(m*I	W/(m·K)	Thermal conductivity
○ k (anisotropic)	0.025 0 0 0.025	W/(m·K)	Thermal conductivity
ρ	rho(p[1/Pa],T[1/K	kg/m^3	Density
● C_p	Cp(T[1/K])[J/(kg*	J/(kg·K)	Heat capacity at constant pressure
○ γ	1		Ratio of specific heats
Q	0	W/m^3	Heat source
u	0 0	m/s	Velocity field

☐ Viscous heating

η	eta(T[1/K])[Pa*s]	Pa·s	Dynamic viscosity
$h_i\mathbf{N}_{D,i}$	Species diffusion inactive		

Figure 9-9: The update of a material property expression in the Subdomain Settings dialog box.

Note that you can only edit materials currently in the **Model** folder in this way and that any changes are local to the current model. The original material in the library file remains intact. Within the **Materials/Coefficients Library** dialog box you can, however, change a material's name and then **Copy** and **Paste** it into one of the other libraries. Afterwards, clicking **OK** saves the new material for future use in the corresponding library text file.

COMSOL Multiphysics also allows you to set up new materials by creating a new library file, as well as change the existing files using any text editor. A detailed description of this process appears in the *COMSOL Multiphysics User's Guide*, where you also find complete documentation on the functionality of the Materials/Coefficients Library.

10

Glossary

This glossary contains application-specific terms used in the Chemical Engineering Module software and documentation. For finite element modeling terms, mathematical terms, and geometry and CAD terms, see the glossary in the *COMSOL Multiphysics User's Guide*. For references to more information about a term, see the index.

Glossary of Terms

absorption Uptake of a gas into the bulk of a liquid. Gas absorption takes place for example in the liquid of a scrubber tower where an up-streaming gas is washed by a down-going flow of a scrubber solution.

adsorption Attachment of a molecule or atom to a solid surface. Adsorption involves a chemical bond between the adsorbed species and the surface.

anisotropy Variation of a transport property in different directions in a material. Is often obtained from homogenization of regular structures, for example, monolithic structures in tubular reactors.

Arrhenius rate equation Expression that relates the rate constant of a chemical reaction to the exponential of the temperature.

batch reactor Reactor characterized by its operation, which means that the reactor does reaches steady state.

bipolar plate Electrically conducting plate connected to the anode on one side and to the cathode on the other side in an electrochemical cell.

Brinkman equations Extension of Darcy's law in order to include the transport of momentum through shear in porous media flow.

Butler-Volmer equation Expression that relates the reaction rate of an electron transfer reaction on an electrode surface to the exponential of the overpotential. The equation can be derived from the Arrhenius rate equation by accounting for the contribution of the electric potential to the activation energy.

boundary layer Region in a fluid close to a solid surface. This region is characterized by large gradients in velocity and is often treated with approximative methods, because it is difficult to geometrically resolve the large gradients found there.

chemisorption See *adsorption*.

continuous reactor Reactor that operates without interruption. This type of reactor is characterized by its steady-state operation.

diffusion layer Fictive layer in a fluid close to a solid surface where a chemical reaction takes place. The flux of species perpendicular to the surface in this layer is dominated by diffusion.

Darcy's law Equation that gives the velocity vector as proportional to the pressure gradient. Often used to describe flow in porous media.

electroneutrality condition Condition that states that the sum of charges in a control volume in an electrolyte should be zero.

electrophoresis Migration of charged electrolyte ions in an electric field.

electroosmosis Onset of a flow due to the application of an external electric field or due to the formation of an electric field created by ion transport in membranes, for example.

Euler flow Flow at high velocities, where incompressibility of the fluid is of importance whereas the influence of viscous momentum transport is negligible.

Fick's law The first law relates the concentration gradients to the diffusive flux of a solute infinitely diluted in a solvent. The second law introduces the first law into a differential material balance for the solute.

fully developed laminar flow Laminar flow along a channel or pipe that only has velocity components in the main direction of the flow. The velocity profile perpendicular to the flow does not change downstream in the flow.

Hagen-Poiseuille equation See *Poiseuille's law.*

heterogeneous reaction Reaction that takes place at the interface between two phases.

homogeneous reaction Reaction that takes place in the bulk of a solution.

law of the wall See *wall function.*

Maxwell-Stefan equations Set of equations that describe the diffusion of solutes and solvent in a concentrated solution. In such a solution, the solutes interact with each other and with the solvent.

monolithic reactor Catalytic reactor made of one single piece of solid material. Incorporates a catalytic structure in its often porous structure.

Navier-Stokes equations Equations for the momentum balances coupled to the equation of continuity for a Newtonian incompressible fluid.

Nernst-Planck equation Equation that describes the flux of an ion through diffusion, convection, and migration in an electric field. The equation is valid for diluted electrolytes.

Newtonian flow Flow characterized by a constant viscosity or a viscosity that is independent of the shear rate in the fluid.

physisorption See *adsorption*.

Poiseuille's law Equation that relates the mass rate of flow in a tube as proportional to the pressure difference per unit length and to the fourth power of the tube radius. The law is valid for fully developed laminar flow.

RANS Reynolds-averaged Navier-Stokes, which implies time averaging of the velocity fluctuations in turbulent flow. The Reynolds' stresses obtained by this averaging have to be expressed with an additional set of equations. Turbulence models like the k-ε and k-ω models belong to this class.

specific surface area Internal surface area of a porous structure given in area per unit volume, which yields the unit one over unit length. Often used to characterize the structure of porous catalysts.

streamline-diffusion stabilization A numerical technique for stabilization of the numeric solution to a convection-dominated PDE by artificially adding diffusion in the direction of the streamlines.

switch function Conditional function that gives a smooth onset of a variable, for example from 0 to 1 or from 1 to 0. Often used for phase changes or saturation.

wall function Semi-empirical expression for the anisotropic flow close to a solid surface used in turbulence models. Often based on negligible variations in pressure gradient in the direction tangential to the surface.

INDEX