Plasma Module

User’s Guide

6.2

COMSOL

Plasma Module User's Guide

Version: COMSOL 6.2

Contact Information

Visit the Contact COMSOL page at www.comsol.com/contact to submit general inquiries or search for an address and phone number. You can also visit the Worldwide Sales Offices page at www.comsol.com/contact/offices for address and contact information.

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

- Support Center: www.comsol.com/support
- Product Download: www.comsol.com/product-download
- Product Updates: www.comsol.com/product-update
- COMSOL Blog: www.comsol.com/blogs
- Discussion Forum: www.comsol.com/forum
- Events: www.comsol.com/events
- COMSOL Video Gallery: www.comsol.com/videos
- Support Knowledge Base: www.comsol.com/support/knowledgebase

Part number: CM021401

Contents

Chapter 1: Introduction

Chapter 2: Data Required for Plasma Modeling

Chapter 3: The Boltzmann Equation, Two-Term Approximation Interface

Chapter 4: The Drift Diffusion Interface

Chapter 5: The Heavy Species Transport Interface

Chapter 6: Plasma Interfaces

Chapter 7: Equilibrium Discharges Interfaces

Chapter 8: The Charge Transport Interface

Chapter 9: The Corona Discharge Interface

Chapter 10: The Electrical Breakdown Detection Interface

Chapter 11: AC/DC Interfaces

Chapter 12: Fluid Flow Interface

1

Introduction

This guide describes the Plasma Module, an optional add-on package that extends the COMSOL Multiphysics® modeling environment with customized physics interfaces and functionality optimized for the analysis of nonequilibrium and equilibrium discharges.

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

- About the Plasma Module
- Overview of the User's Guide

About the Plasma Module

The Plasma Module provides a set of user interfaces that are used to define the physics, boundary conditions, and material properties that collectively constitute a plasma. For nonequilibrium plasmas transport equations for electrons, ions, and neutrals are solved self-consistently coupled with Poisson's equation. For equilibrium discharges the plasma is treated as an electric conductive fluid with one temperature in a MHD approach. There are also dedicated interfaces to model corona discharges and to determine if gas breakdown can occur. Additional heating mechanisms, such as induction heating or wave heating, can be included by solving Ampère's law or the wave equation.

In this section:

- How the Plasma Module Improves Your Modeling
- Plasma Module Nomenclature
- The Plasma Module Physics Interface Guide
- Common Physics Interface and Feature Settings and Nodes
- Where Do I Access the Documentation and Application Libraries?

The Physics Interfaces and Building a COMSOL Multiphysics Model in the *COMSOL Multiphysics Reference Manual*

How the Plasma Module Improves Your Modeling

Low-temperature plasmas represent the amalgamation of fluid mechanics, reaction engineering, physical kinetics, heat transfer, mass transfer, and electromagnetics. The Plasma Module is mostly focused on modeling low-temperature plasmas and offers dedicated physics interfaces for the most common types of plasma reactors as presented below.

- Inductively coupled plasmas (ICP)
- Capacitively coupled plasmas (CCP)
- ICP/CCP
- Microwave plasmas
- Light sources
- Electrical breakdown
- DC discharges
- Chemical vapor deposition (CVD)
- Plasma-enhanced chemical vapor deposition (PECVD)
- Dielectric barrier discharges (DBD)
- Electron cyclotron resonance (ECR)
- Reactive gas generators

The complexity of plasma modeling lies in the fact that it combines elements of reaction engineering, statistical physics, fluid mechanics, physical kinetics, heat transfer, mass transfer, and electromagnetics. The net result is a true multiphysics problem involving complicated coupling between the different physics. The Plasma Module is designed to simplify the process of setting up a self-consistent model of a low-temperature plasma.

The physics interfaces include all the necessary tools to model plasma discharges, beginning with a Boltzmann Equation, Two-Term Approximation solver that computes the electron transport properties and source coefficients from a set of electron impact collision cross sections. This physics interface makes it possible to determine many of the interesting characteristics of a discharge by providing input properties such as the electric field and the electron impact reactions that make up the plasma chemistry, without solving a space-dependent problem.

For space-dependent models, the reactions and species which make up the plasma chemistry are conveniently managed in the Model Builder. Reactions and species can be added manually or loaded from a file. It is possible to define electron impact reactions, heavy species reactions, and surface reactions. Electron impact reactions can be specified by electron impact cross sections. In this case, rate constants are obtained by suitable integration of the cross section over the electron energy distribution function which can be an analytic function or can come from the solution of the Boltzmann Equation, Two-Term Approximation.

When the fluid velocity and gas temperature are of interest, there are physics interfaces available for laminar flow and heat transfer. There are several options available when coupling the charged species transport to the electromagnetic fields. Poisson's equation for the electrostatic potential is always solved. Modeling inductively coupled plasmas where induction currents are responsible for sustaining the plasma requires the AC/DC Module. For wave-heated discharges (microwave plasmas), the RF Module is required.

Modeling the interaction between the plasma and an external circuit is an important part of understanding the overall characteristics of a discharge. This module has tools to add circuit elements directly to a 1D, 2D, or 3D model Component node, or import an existing SPICE netlist into the model.

Plasma Module Nomenclature

The Plasma Module has numerous variables and expressions used throughout the documentation and in COMSOL Multiphysics. Table 1-1 lists the variable, its description and its SI unit.

TABLE 1-1: NOMENCLATURE USED THROUGHOUT THE PLASMA MODULE DOCUMENTATION.

VARIABLE	DESCRIPTION	SI UNIT
ρ	Gas density	kg/m^3
p	Pressure	Pa
T	Temperature	K
M_k	Molecular weight, species k	kg/mol
M_n	Mean molar mass	kg/mol
w_k	Mass fraction, species k	1
W_k	Log of the mass fraction of species k	-
x_k	Mole fraction, species k	1
$\mathbf{V}_k$	Multicomponent diffusion velocity, species k	m/s
$\mathbf{j}_k$	Diffusive flux vector, species k	kg/(m^2·s)
D_{kj}	Binary diffusion coefficient	m^2/s
$D_{k,m}$	Mixture averaged diffusion coefficient	m^2/s
D_k^T	Multicomponent thermal diffusion coefficient	kg/(m·s)
$D_{k,f}$	Diffusion coefficient for species k	m^2/s
Z_k	Charge, species k	1
$\mu_{k,m}$	Mixture averaged mobility, species k	m^2/(V·s)
V	Electric potential	V
$\mathbf{E}$	Electric field	V/m
R	Universal gas constant	J/(mol·K)

TABLE 1-1: NOMENCLATURE USED THROUGHOUT THE PLASMA MODULE DOCUMENTATION.

VARIABLE	DESCRIPTION	SI UNIT
c_k	Molar concentration, species k	mol/m^3
r_j	Reaction rate, reaction j	mol/(m^{3}s)
k_j	Rate coefficient, reaction j	m^3/(mol·s)
R_k	Rate expression, species k	kg/(m^3·s)
n_k	Number density, species k	1/m^3
ν'_{kj}	Stoichiometric matrix, forward component	1
ν''_{kj}	Stoichiometric matrix, reverse component	1
ν_{kj}	Total stoichiometric matrix	1
q	Unit charge	C
k_B	Boltzmann constant	J/K
$R_{\mathrm{surf},k}$	Surface rate expression for species k	mol/(m^2·s)
γ_f	Forward sticking coefficient	1
γ_i	Secondary emission coefficient	1
ε_i	Mean energy of secondary electrons	V
ρ_k	Mass concentration, species k	kg/m^3
$\mathbf{u}$	Mass averaged fluid velocity vector	m/s
F	Faraday's constant	s·A/mol
N_A	Avogadro constant	1/mol
$\mathbf{v}^T$	Thermal drift velocity	m/s
u	Mass averaged velocity field, x/r-component	m/s
v	Mass averaged velocity field, y/z-component	m/s
w	Mass averaged velocity field, z-component	m/s
σ	Surface charge density	C/m^2
Γ_e	Electron flux	1/(m^2·s)
q_j	Surface reaction rate, reaction j	mol/(m^2·s)
A_j	Frequency factor, reaction j	m^3/(mol·s)
β_j	Temperature exponent, reaction j	1
E_j	Activation energy, reaction j	J/mol
$f(\varepsilon)$	Electron energy distribution function (EEDF)	V$^{-3/2}$
ε	Electron energy	V
$\bar{\varepsilon}$, φ	Mean electron energy	V

TABLE 1-1: NOMENCLATURE USED THROUGHOUT THE PLASMA MODULE DOCUMENTATION.

VARIABLE	DESCRIPTION	SI UNIT
μ_e	Electron mobility	$m^2/(V \cdot s)$
N_n	Neutral number density	$1/m^3$
$\mu_e N_n$	Reduced electron mobility	$1/(V \cdot m \cdot s)$
θ	Out-of-plane direction	rad
I_p	Plasma current	A

The Plasma Module Physics Interface Guide

The Plasma Module extends the functionality of the physics interfaces of the base package for COMSOL Multiphysics. The details of the physics interfaces and study types for the Plasma Module are listed in the table. The functionality of the COMSOL Multiphysics base package is given in the *COMSOL Multiphysics Reference Manual*.

In the *COMSOL Multiphysics Reference Manual*:

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

PHYSICS INTERFACE	ICON	TAG	SPACE DIMENSION	AVAILABLE STUDY TYPE
AC/DC				
Electrical Circuit		cir	Not space dependent	stationary; frequency domain; time dependent; frequency domain; eigenfrequency
Electrostatics[1]		es	all dimensions	stationary; time dependent; stationary source sweep

PHYSICS INTERFACE	ICON	TAG	SPACE DIMENSION	AVAILABLE STUDY TYPE
Fluid Flow				
Single-Phase Flow				
Laminar Flow[1]		spf	3D, 2D, 2D axisymmetric	stationary; time dependent
Plasma				
Boltzmann Equation, Two-Term Approximation		be	0D	mean energies; reduced electric fields; time dependent
Plasma, Time Periodic		ptp	1D, 2D axisymmetric	time periodic; time periodic to time dependent
Plasma		plas	all dimensions	time dependent; stationary
Inductively Coupled Plasma[2,4]		—	3D, 2D, 2D axisymmetric.	frequency-transient; frequency-stationary
Inductively Coupled Plasma with RF Bias[2,4]		—	2D, 2D axisymmetric.	Frequency-Time periodic; time periodic to time dependent
Microwave Plasma[3,4]		—	3D, 2D, 2D axisymmetric	frequency-transient
Electric Discharges				
Corona Discharge[4]		—	all dimensions	stationary
Electrical Breakdown Detection		ebd	3D, 2D, 2D axisymmetric	time dependent
Equilibrium Discharges				
Equilibrium Discharges, Out-of-Plane Currents[2,4]		—	2D, 2D axisymmetric	stationary; time dependent; frequency-transient; frequency-stationary

PHYSICS INTERFACE	ICON	TAG	SPACE DIMENSION	AVAILABLE STUDY TYPE
Equilibrium Discharges, In-Plane Currents[2,4]		—	2D, 2D axisymmetric	stationary; time dependent; frequency-transient; frequency-stationary
Equilibrium Discharges[2,4]		—	3D	stationary; time dependent; frequency-transient; frequency-stationary
Species Transport				
Drift Diffusion		dd	all dimensions	time dependent
Heavy Species Transport		hs	all dimensions	time dependent
Charge Transport		ct	all dimensions	stationary

[1] This physics interface is included with the core COMSOL package but has added functionality for this module.

[2] Requires the addition of the AC/DC Module.

[3] Requires the addition of the RF Module.

[4] This physics interface is a predefined multiphysics coupling that automatically adds all the physics interfaces and coupling features required.

Common Physics Interface and Feature Settings and Nodes

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

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

!	In the *COMSOL Multiphysics Reference Manual* see Table 2-4 for links to common sections and Table 2-5 to common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

Where Do I Access the Documentation and Application Libraries?

A number of online resources have more information about COMSOL, including licensing and technical information. The electronic documentation, topic-based (or context-based) help, and the Application Libraries are all accessed through the COMSOL Desktop.

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

THE DOCUMENTATION AND ONLINE HELP

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

Opening Topic-Based Help

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

Win

To open the **Help** window:

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

Mac

Linux

To open the **Help** window:

- In the **Model Builder** or **Physics Builder**, click a node or window and then press F1.
- In the main toolbar, click the **Help** (?) button.
- From the main menu, select **Help>Help**.

Opening the Documentation Window

Mac

Linux

To open the **Documentation** window:

- Press Ctrl+F1.
- In the main toolbar, click the **Documentation** (📘) button.
- From the main menu, select **Help>Documentation**.

THE APPLICATION LIBRARIES WINDOW

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

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

Opening the Application Libraries Window

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

CONTACTING COMSOL BY EMAIL

For general product information, contact COMSOL at info@comsol.com.

COMSOL ACCESS AND TECHNICAL SUPPORT

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

COMSOL ONLINE RESOURCES

COMSOL website	www.comsol.com
Contact COMSOL	www.comsol.com/contact
COMSOL Access	www.comsol.com/access
Support Center	www.comsol.com/support
Product Download	www.comsol.com/product-download
Product Updates	www.comsol.com/product-update
COMSOL Blog	www.comsol.com/blogs
Discussion Forum	www.comsol.com/forum
Events	www.comsol.com/events
COMSOL Application Gallery	www.comsol.com/models
COMSOL Video Gallery	www.comsol.com/videos
Support Knowledge Base	www.comsol.com/support/knowledgebase

Overview of the User's Guide

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

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

TABLE OF CONTENTS, GLOSSARY, AND INDEX

To help you navigate through this guide, see the Contents, Glossary of Terms, and Index. See also Plasma Module Nomenclature in this chapter.

DATA REQUIREMENTS FOR PLASMA MODELING

The Data Required for Plasma Modeling chapter describes the Data Requirements to model low-temperature plasmas. It gives an overview of the data you need to assemble before attempting to model a plasma. It also includes the section Importing Collision Cross-Section Data.

AC/DC INTERFACES

AC/DC Interfaces chapter describes the two physics interfaces available with this module under the AC/DC branch when adding a physics interface. This module enhances the physics interface included with the basic COMSOL Multiphysics license. Use the Electrical Circuit interface to add an external electrical circuit to the plasma model. Also see The Magnetic Fields Interface in the *COMSOL Multiphysics Reference Manual* for information about this physics interface and its feature node settings.

FLUID FLOW INTERFACES

Fluid Flow Interface describes the Laminar Flow interface, which has a few additional features available for this module.

BOLTZMANN EQUATION, TWO-TERM APPROXIMATION INTERFACE

The Boltzmann Equation, Two-Term Approximation Interface chapter describes the physics interface, which computes the electron energy distribution function (EEDF) from a set of collision cross sections for some mean discharge conditions. Additionally, electron source coefficients and transport properties can be computed.

DRIFT DIFFUSION INTERFACE

The Drift Diffusion Interface chapter describes the underlying electron transport theory for the Drift Diffusion interface and details the available features. Use the Drift Diffusion interface to compute the electron density and mean electron energy. A wide range of boundary conditions are available to handle secondary emission, thermionic emission, and wall losses.

THE HEAVY SPECIES INTERFACE

The Heavy Species Transport Interface chapter describes the physics interface, which is a mass balance interface for all nonelectron species. This includes charged, neutral, and excited species. The physics interface also allows you to add electron impact reactions, chemical reactions, surface reactions, and species via the Model Builder.

THE PLASMA INTERFACES

Plasma Interfaces chapter describes the following multiphysics interfaces.

The Plasma Interface can be used to study discharges that are sustained by time-varying electrostatic fields or static electric fields in conjunction with secondary electron emission. Rate or Townsend coefficients can be used to define the collisional processes occurring in the plasma. When rate coefficients are used, cross section data can be imported, and the rate coefficient will be automatically computed.

The Plasma, Time Periodic Interface is used for discharges sustained by a periodic excitation, as is found in capacitively coupled plasmas. The applied frequency should typically be in the MHz range.

The Inductively Coupled Plasma Interface is for studying discharges that are sustained by induction currents. The induction currents are solved for in the frequency domain and all other variables in the time domain. The electron heating due to the induction currents is handled with two multiphysics features. The 3D components require the AC/DC Module.

The Inductively Coupled Plasma with RF Bias Interface is for studying discharges that are sustained by induction currents and have a periodic RF-biased electrode.

The Microwave Plasma Interface requires the RF Module, and is used to study discharges that are sustained by electromagnetic waves (wave-heated discharge). Heating of the electrons due to their interaction with the electromagnetic waves is also handled using multiphysics features. The electromagnetic waves are solved for in the frequency domain.

THE EQUILIBRIUM DISCHARGES INTERFACES

Equilibrium Discharges Interfaces chapter describes physics interfaces that can be used to model plasma in or in close local thermodynamic equilibrium (LTE), that is, where the electrons and heavy particles temperature are approximately equal. Close to LTE conditions are usually observed at high pressure or under small values of electrical fields.

THE ELECTRIC DISCHARGES INTERFACES

The Electrical Breakdown Detection Interface uses an approximate method to determine if electric breakdown occurs in a given system by integrating Townsend coefficients along electric field lines.

The Corona Discharge Interface uses a simplified charge transport model coupled with electrostatics to provide an approximate method of computing the charge density and the electrostatic potential in corona discharges.

2

Data Required for Plasma Modeling

Modeling of low-temperature plasmas is not only difficult because of the large number of physical processes occurring but also because of the amount of data that needs to be supplied to a model. This chapter gives an overview of the data needed to assemble before attempting to model a plasma.

In this chapter:

Data Requirements

This section gives an overview of the data to be assembled before attempting to model a plasma.

One of the main difficulties is to find a complete and physically correct chemical mechanism for the plasma of interest. This can involve just a handful of reactions and species (as for argon), or in the case of molecular gases there can be hundreds of reactions and tens of hundreds of species. Often the chemical mechanism can be found from a literature search, but if it is a unique plasma chemistry then the chemical mechanism is likely unknown. The only option is to find a plasma chemistry that exhibits similar properties to your specific application and use that as a reference.

The following sections is a guide for the data need for each of the features which make up a plasma model.

- Electron Impact Reactions
- Reaction
- Surface Reaction
- Species

Electron Impact Reactions

For each ***electron impact reaction***, the reaction formula is the first piece of information that is required. Most published papers involving the numerical modeling of plasmas include a table of the reactions which make up the plasma chemistry.

Once the formula is entered the dependency of the reaction rate on the electron energy needs to be supplied. The most common way of accomplishing this is to specify ***cross-section data*** for each of the electron impact reactions. The cross-section data can be hard to find or not even exist. Cross-section data is available for most of the common gases. The LXCat Plasma Data Exchange Project (www.lxcat.net) is an excellent source to obtain electron impact cross-sections. When using data from the LXCat database it is encouraged to refer to the data as suggested in the LXCat website. Many models in COMSOL use data from LXCat. Any reutilization of these data should also be reference following LXCat norms. It is also encouraged to download the data directly from the LXCat database.

! Cross-section data must be in a specific format in order to be imported into COMSOL Multiphysics. See Importing Collision Cross-Section Data.

If cross-section data is not available then the rate coefficient can be specified with a constant value, an Arrhenius expression, or by a lookup table. In the case of an Arrhenius expression, the Arrhenius coefficients must be specified. If it is preferred to specify the rate coefficient using a lookup table the table of rate or Townsend coefficient versus mean electron energy need to be loaded.

Depending on the type of electron impact collision, the following data is also required:

- The mass ratio of the electron to the target species for elastic collisions.
- The energy loss (in eV) for inelastic collisions.
- The ratio of statistical weights between the upper and lower levels of the heavy species if a detailed balance is required.

DETAILED BALANCE

If a detailed balance is requested the cross–section data for the reverse reaction is created automatically using the principle of the detailed balance (Ref. 1)

$$\sigma^{\mathrm{inv}}(\varepsilon) = \frac{g^{\mathrm{low}}}{g^{\mathrm{up}}}\frac{\varepsilon+\Delta\varepsilon}{\varepsilon}\sigma(\varepsilon+\Delta\varepsilon)$$

where σ is the cross section for the direct reaction, ε is electron energy, $\Delta\varepsilon$ is the excitation energy, and g^{low} and g^{up} are the statistical weights of the lower and upper levels of the heavy species.

In practice, if an electron impact reaction created from a cross-section import file is defined as

```
e+Ar=>e+Ars
```

and the reverse reaction is requested the following reaction is created automatically

```
e+Ars=>e+Ar
```

with the cross section being defined by the principle of detail balance.

When a reverse reaction is requested the ratio of the statics weights needs to be provided. For atomic species the static weight of a level is defined as

$$g = 2J + 1$$

where J is the total electronic angular momentum. The values for J can be obtained from Ref. 2. As an example, if the first four 4s levels of argon are to be used in a plasma model as a lumped level, the statistical weight of the excited lumped state is 12, which results from the sum of the statistical weights of the individual levels Table 2-1.

TABLE 2-1: ATOMIC DATA FOR THE 4S LEVELS OF ARGON Ref. 2.

TERM	J	Level (eV)	g
$^2[3/2]^o$	2	11.548	5
$^2[3/2]^o$	1	11.624	3
$^2[1/2]^o$	0	11.723	1
$^2[1/2]^o$	1	11.828	3

For more details on the application of the principled of detail balancing and on the computation of statistical weights for molecules the reader is referred to Ref. 3 (page 267).

Reaction

For gas phase reactions, the forward rate constant using either Arrhenius coefficients or numeric value data is required. If the reaction is reversible then enter the reverse rate constant in the same way.

Surface Reaction

For surface reactions the following data is required:

- The forward sticking or rate coefficient, which can either be a numeric value or specified in terms of Arrhenius parameters as a function of the surface temperature.
- For first-order reactions, the total surface site concentration is required.
- For surface reactions that result in emission of secondary electrons, the secondary emission coefficient and mean energy of the secondary electrons is required.

Species

For each of the heavy species the following information is required:

- The molecular weight of the species
- The potential characteristic length of the species; required to compute the correct diffusivity and mobility
- The potential energy minimum of the species; required to compute the correct diffusivity and mobility

In cases where the neutral gas temperature is solved for, the following additional information is required:

- The species molar enthalpy, entropy, and specific heat, which can be entered directly as a function of temperature or by using *NASA polynomials*
- In the case of electronically excited species or ions, the properties of the ground state species can be entered, and an additional contribution to the species enthalpy can also be entered

The Plasma Module provides predefined transport and thermodynamic data for the following species:

TABLE 2-2: TABLE OF GASES WITH PREDEFINED TRANSPORT AND THERMODYNAMIC DATA.

GAS	FORMULA
Helium	He
Neon	Ne
Argon	Ar
Krypton	Kr
Xenon	Xe
Carbon	C
Carbon hydride	CH
Methylene	CH2
Methyl radical	CH3
Methane	CH4
Carbon monoxide	CO
Carbon dioxide	CO2
Atomic fluorine	F
Fluorine	F2

TABLE 2-2: TABLE OF GASES WITH PREDEFINED TRANSPORT AND THERMODYNAMIC DATA.

GAS	FORMULA
Sulfur hexafluoride	SF6
Silicon tetrafluoride	SiF4
Atomic hydrogen	H
Hydrogen	H2
Water	H2O
Hydrogen peroxide	H2O2
Nitroxyl	HNO
Hydroperoxyl	HO2
Hydroxyl	OH
Atomic nitrogen	N
Nitrogen	N2
Nitrous oxide	N2O
Amino radical	NH
Amino radical	NH2
Ammonia	NH3
Nitric oxide	NO
Nitrogen dioxide	NO2
Atomic oxygen	O
Oxygen	O3
Ozone	O3
Silane	SiH4
Silane radical	SiH2
Disilane	Si2H6

The following surface species properties are also predefined:

TABLE 2-3: TABLE OF SURFACE SPECIES WITH PREDEFINED PHYSICAL AND THERMODYNAMIC DATA.

NAME	FORMULA
Silicon	Si
Silicon hydride	SiH
Silicon dioxide	SiO2

Thermodynamic and transport properties for many other species can be found at the following links:

- Thermodynamic properties: https://burcat.technion.ac.il/
- Transport properties: https://uigelz.eecs.umich.edu/pub/data/e_reactions.pdf
- Additional transport properties: http://combustion.berkeley.edu/gri-mech/version30/files30/transport.dat

Importing Collision Cross-Section Data

To facilitate the modeling process, all multiphysics interfaces and the Boltzmann Equation, Two-Term Approximation interface can import sets of collision cross sections from file. This section details the file format required and has these topics:

- Cross-Section Data File Format
- The File Format
- References for the Plasma Module Cross-Section Data Requirements

Cross-Section Data File Format

IMPORTING CROSS SECTIONS

The file format used to import cross-section data is almost the same as the format proposed by Hagelaar in Ref. 4. The best way to get the cross-section data in the correct format is to modify the file `Ar_xsecs.txt` which is available in the Plasma Module Application Library folder.

For each electron impact reaction, the *lookup table* of electron energy (eV) versus collision cross section (m^2) allows electron transport properties and source coefficients to be computed. As an example, a set of collision cross sections for oxygen are shown in Figure 2-1.

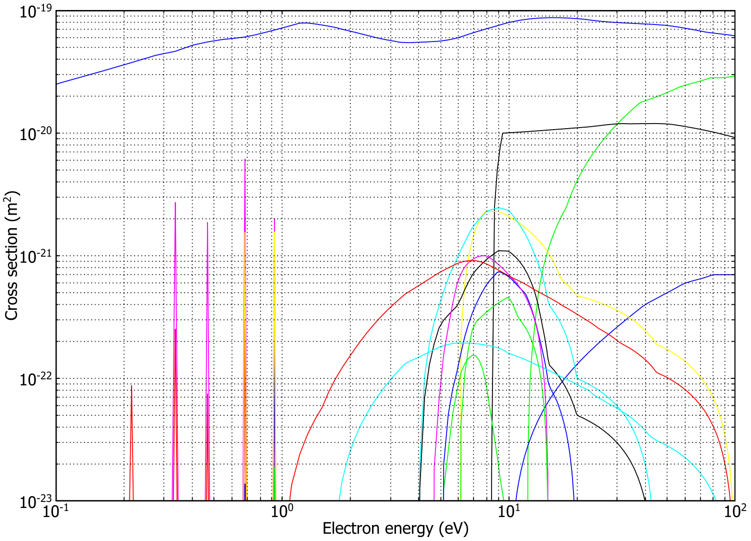


Figure 2-1: Plot of a set of collision cross sections for molecular oxygen. There are 16 collisions in total.

The File Format

Cross-section data must be specified using the following format:

```
1st line
2nd line
3rd line
4th line

---------------------------
energy data (eV) Collision cross section (m2)
---------------------------
```

1 1st line: Keyword in capitals indicating the type of the collision. Possible collision types are ELASTIC, EXCITATION, IONIZATION, and ATTACHMENT. Note the use of capital letters.

2 2nd line: Reaction formula. This formula carries over to the COMSOL Multiphysics GUI so that whatever is entered here appears in the **Formula** field. The formula must be separated by a => sign.

3 3rd line: For elastic collisions, the ratio of the electron mass to the target particle mass. For attachment, the 3rd line is 0 (zero). For ionization, the threshold energy in eV. For excitation there are three (3) numbers separated by a space. The first number is the excitation energy (eV). The second number is the ratio of statistical weights of the final state to the initial state. The third number is a flag that determines whether or not a detailed balance is used. Use a 1 to construct a detailed balance and 0 otherwise.

4 4th line: Two unit conversion factors by which the columns of data are multiplied.

5 Table of the cross section as a function of energy. The table starts and ends by a line of dashes `'------'` (at least five), and has otherwise two numbers per line: the energy in eV and the cross section in m^2. The table does not need to follow the other lines immediately; that is, user comments such as reference information can be included above the table.

Here are some examples for the types of collisions available:

```
ELASTIC
e+02=>e+02
1.71e-5
1.0 1.0
--------------------------
0.000000e+000 3.500000e-021
.
.
.
--------------------------

ATTACHMENT
e+02=>0+0-
0
1.0 1.0
--------------------------
5.800000e-002 2.220446e-036
.
.
.
--------------------------

EXCITATION
e+02=>e+02b1s
1.6270   5 1
1.0 1.0
--------------------------
1.627000e+000 2.220446e-036
.
.
```

```
.
---------------------------

IONIZATION
e+O2=>2e+O2+
12.0600
1.0 1.0
---------------------------
1.206000e+001 2.220446e-036
.
.
.
---------------------------
```

References for the Plasma Module Cross-Section Data Requirements

1. B.M. Smirnov, *Theory of Gas Discharge Plasma*, Springer, 2015.

2. https://www.nist.gov/pml/atomic-spectra-database.

3. M.A. Lieberman and A.J. Lichtenberg, *Principles of Plasma Discharges and Materials Processing*, John Wiley & Sons, 2005.

4. G.J.M. Hagelaar and L.C. Pitchford, “Solving the Boltzmann equation to obtain electron transport coefficients and rate coefficients for fluid models”, *Plasma Sources Science and Technology*, vol. 14, pp. 722–733, 2005.

3

The Boltzmann Equation, Two-Term Approximation Interface

This chapter describes the Boltzmann Equation, Two-Term Approximation interface, found under the **Plasma** branch ( ) when adding a physics interface.

In this chapter:

The Boltzmann Equation, Two-Term Approximation Interface

The **Boltzmann Equation, Two-Term Approximation (be)** interface ( ), found under the **Plasma** branch ( ) when adding a physics interface, studies the electron energy distribution function (EEDF) and calculates it from a set of collision cross sections for some mean discharge conditions. Electron source coefficients and transport properties are also computed. The physics interface is unique because the geometry and mesh are hidden during setup of the model. The Boltzmann Equation, Two-Term Approximation interface can be used as a preprocessing step before solving space-dependent models. This physics interface is available for 0D components.

When this physics interface is added, these default nodes are also added to the **Model Builder**: **Boltzmann Model** and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement. You can also right-click **Boltzmann Equation, Two-Term Approximation** to select physics features from the context menu.

The Reduced Electric Fields and Mean Energies studies are available for this physics interface and described in the *COMSOL Multiphysics Reference Manual*.

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `be`.

ELECTRON ENERGY DISTRIBUTION FUNCTION SETTINGS

Select an **Electron energy distribution function** (EEDF) — **Maxwellian** (the default), **Druyvesteyn**, **Generalized**, **Boltzmann equation, two-term approximation (linear)**, or **Boltzmann equation, two-term approximation (quadratic)**.

- Select **Maxwellian** to assume a Maxwellian EEDF, which takes the form:

$$f(\varepsilon) = \varphi^{-3/2}\beta_1\exp(-(\varepsilon\beta_2/\varphi))$$

where

$$\beta_1 = \Gamma(5/2)^{3/2}\Gamma(3/2)^{-5/2}, \beta_2 = \Gamma(5/2)\Gamma(3/2)^{-1}$$

and φ is the mean electron energy (eV), ε is the electron energy (eV) and Γ is the upper incomplete gamma function.

- Select **Druyvesteyn** to assume a Druyvesteyn EEDF, which takes the form:

$$f(\varepsilon) = 2\varphi^{-3/2}\beta_1\exp(-(\varepsilon\beta_2/\varphi)^2)$$

where

$$\beta_1 = \Gamma(5/4)^{3/2}\Gamma(3/4)^{-5/2}, \beta_2 = \Gamma(5/4)\Gamma(3/4)^{-1}$$

- Select **Generalized** for a generalized distribution function where the EEDF is somewhere between Maxwellian and Druyvesteyn. For **Generalized** enter a value for the **Power law** g (dimensionless). The default is 1, and this number should be between 1 and 2. Mathematically, the EEDF takes the form:

  $$f(\varepsilon) = g\varphi^{-3/2}\beta_1\exp(-(\varepsilon\beta_2/\varphi)^g)$$

  where

  $$\beta_1 = \Gamma(5/2g)^{3/2}\Gamma(3/2g)^{-5/2}, \beta_2 = \Gamma(5/2g)\Gamma(3/2g)^{-1}.$$

  When $g = 1$ the distribution function is Maxwellian and when $g = 2$ it is Druyvesteyn.

- Select **Boltzmann equation, two-term approximation (linear)**, or **Boltzmann equation, two-term approximation (quadratic)** to solve a two-term approximation to the Boltzmann equation. When selected, the EEDF is computed from a partial differential equation instead of by taking an assumed function. The two-term Boltzmann equation is a complicated, nonlocal integral equation.

These following options are available when **Boltzmann equation, two-term approximation (linear)**, or **Boltzmann equation, two-term approximation (quadratic)** is selected as the **Electron energy distribution function**. These settings also enable available in the **Settings** window for the **Boltzmann Model** feature.

- Select the **Temporal behavior** — **Stationary EEDF** (the default), or **Time dependent EEDF**.
- Select **Electron-electron collisions** (off by default) if the ionization degree of the discharge is high. Electron-electron collisions tend to make the distribution function more Maxwellian and they also increase the complexity of the problem.
- Select **Equal secondary electron energy sharing** (on by default) to describe how the energy is split between two electrons when an ionization collision occurs. If selected, then both electrons take an equal energy after the collision. If not selected, the secondary electron created in an ionization collision has zero energy and the ionizing electron carries all the excess energy.
- Select **Oscillating field** if the reduced angular frequency of the discharge is high, which is typically only true for microwave discharges. The reduced angular frequency is the ratio of the angular frequency and the neutral number density, ω/N.
- If the **Oscillating field** property is active, enter a **Reduced angular frequency** ω/N (SI unit: m^3/s). The default is 10^{-13} m^3/s. If the reduced angular frequency is high, the proportion of electrons with high energies is substantially increased for the same mean electron energy. This is because in DC fields, collisional momentum transfer impedes electrons acquiring higher energies but high frequency fields have the opposite effect.
- Enter the **Number of elements in eedf extra dimension** N (SI unit: dimensionless) to specify the number of mesh elements to use to discretize the underlying energy space. The default is 100, but models with complex gas mixtures may require more.
- Enter the **Element ratio in eedf extra dimension** R (SI unit: dimensionless) to specify the rate at which the mesh coarsens away from the zero energy coordinate. The default is 10, and this does not usually need to be changed. Higher values mean that the mesh will be finer closer to the zero energy, and coarser at higher energies.
- Select **Compute maximum energy** to have the software automatically compute the maximum energy coordinate based on certain assumptions about the EEDF. This is a powerful feature, in that the maximum energy coordinate does not need to be specified, but it does make the problem more nonlinear and thus difficult to solve.
- Enter the **Maximum energy** ε_{max} (SI unit: V) to specify the maximum coordinate in energy space on which we are computing the EEDF. When computing the EEDF

at high mean energies, this value may need increasing from its default value of 100 V. This option is only available if **Compute maximum energy** is not selected.

- Enter the **Maximum energy multiplication factor** χ_ε (SI unit: dimensionless) to specify how much the maximum energy coordinate should be scaled. This option is only available if **Compute maximum energy** is selected and is only needed for the **Reduced electric fields** study type.
- Enter the **EEDF minimum value** f_{min} (SI unit: dimensionless) to specify the minimum value that the EEDF should take at the maximum energy coordinate. In order to avoid divide by zero problems, the value should be small and positive. The default value of 1E-15 rarely needs to be changed.

DEPENDENT VARIABLES

The dependent variable (field variable) is the **Log of EEDF**. The name can be changed but the names of fields and dependent variables must be unique within a model.

- Global Nodes for the Boltzmann Equation, Two-Term Approximation Interface
- Theory for the Boltzmann Equation, Two-Term Approximation Interface

Argon Boltzmann Analysis: Application Library path **Plasma_Module/ Two-Term_Boltzmann_Equation/boltzmann_argon**

Global Nodes for the Boltzmann Equation, Two-Term Approximation Interface

The Boltzmann Equation, Two-Term Approximation Interface has only global nodes, listed in alphabetical order, available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or right-click to access the context menu (all users).

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

- Boltzmann Model
- Cross Section Import
- Electron Impact Reaction
- Initial Values

! In the *COMSOL Multiphysics Reference Manual* see Table 2-4 for links to common sections and Table 2-5 to common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

Boltzmann Model

In the **Boltzmann Model** node, values for specific discharge parameters are set. The number of parameters needed depends on the property settings for the physics interface.

BOLTZMANN SETTINGS

- Enter a **Gas temperature** T_g (SI unit: K). The default is 300 K. The gas temperature influences the amount of energy loss that an electron suffers during an elastic collision. The higher the gas temperature, the lower the energy loss for an electron during an elastic collision.
- Enter an **Absolute pressure** P_A (SI unit: Pa). The default is 1 atm. The absolute pressure is only available when **Temporal behavior** is set to **Time dependent EEDF**. The **Absolute pressure** is used to compute the gas number density.
- If the **Electron-electron collisions** property is active in the physics interface Electron Energy Distribution Function settings section, enter an estimate for the **Electron density** n_e (SI unit: $1/m^3$).

  There is a weak dependence on the electron density in Equation 3-1 which accounts for the screening of the Coulomb potential by space charge effects (Ref. 1).
- If the **Electron-electron collisions** property is active in the physics interface Electron Energy Distribution Function settings section, enter an **Ionization degree** β (dimensionless). The ionization degree can have a significant effect on the shape of the electron energy distribution function. When the ionization degree is very low, the tail of the distribution function tends to decrease very rapidly. As the ionization degree increases, the distribution function tends to become more Maxwellian, resulting in a greater proportion of electrons with higher energies. This can have a profound effect on the rate coefficients for ionization and excitation collisions because of their high energy threshold.

- Enter a **Reduced electric field** E/N (SI unit: V/m^2). The default is 10 Td. The reduced electric field is used as an input to the Boltzmann equation and is responsible to heat the electrons. The **Reduced electric field** is only available when **Temporal behavior** is set to **Time dependent EEDF**. The pressure is used to compute the gas number density.

MOLE FRACTION SETTINGS

Specify the mole fraction of each of the target species in the set of collision cross sections.

Since the sum of all the mole fractions must be equal to one, select one of the species from the **Mole constrained species** list. The mole fraction of the selected species is based on the mole fractions of all the other species.

Mole fraction. There are N text fields available for setting the mole fractions of each species, where N is the total number of target species. Enter the **Mole fraction** expected to be observed in the discharge for each target species. The mole fraction of electronically excited species can have a strong effect on the shape of the electron energy distribution function.

RESULTS

It can be a little bit tricky or time consuming to generate plots of certain output data if there are many species present in the model. Therefore, the **Results** section allows for many common quantities of interest to be plotted by default. Note that if a solution is first computed, then certain values in the **Results** section are changed, the plots are not automatically updated. The solver configuration should be deleted, and the study recomputed.

Initial Values

The **Initial Values** node allows you to set an initial guess for the mean electron energy, the reduced electric field, and the initial condition for a Lagrange multiplier.

INITIAL VALUES

- Enter an **Initial mean electron energy** ε_0 (SI unit: V). The default is 2 V.

  This value should be equal to the mean electron energy of the first parameter in the parametric sweep. COMSOL Multiphysics constructs an initial guess at the EEDF by assuming a Maxwellian distribution function with the specified mean electron energy.

- Enter an **Initial reduced electric field** E/N_0 (SI unit: $\text{V}\cdot\text{m}^2$). The default is 1 Td and this rarely needs to be changed. This is the initial reduced electric field used to compute the distribution function.
- Enter an **Initial condition for lam** λ_0 (SI unit: m^3/s). The default is `1e-21` and this rarely needs to be changed. This value is an initial guess for a Lagrange multiplier, which is used to enforce a normalization constraint on the EEDF. Physically, the Lagrange multiplier represents the reduced net ionization frequency and is obtained such that equation Equation 3-2 is satisfied.

Cross Section Import

Browse to the file to be imported. The file must follow the format documented in Importing Collision Cross-Section Data.

Electron Impact Reaction

Use the **Electron Impact Reaction** node to add collisions to a model by entering a formula and some properties of the collision. The cross-section data can be entered manually or by loading it from a file.

💡	Load in multiple cross sections from a file in the **Boltzmann Equation, Two-Term Approximation** settings. In this case, the collision features are automatically generated.

COLLISION

Enter a **Formula** for the collision.

Select a **Collision type**: **Elastic** (the default), **Excitation**, **Attachment**, or **Ionization**.

- For **Elastic** enter a value for the **Electron mass ratio** m_r (dimensionless) (the ratio of the mass of the electron to the target species).
- Select **Excitation** for excitation reactions where the collision is inelastic. Enter the **Energy loss** $\Delta\varepsilon$ (SI unit: V) for the reaction. This is also known as threshold energy for the reaction. Select this option for electronic excitation, vibrational and rotational excitation, and superelastic collisions. For superelastic collisions, the energy loss is negative.

- Select **Attachment** for collisions which result in the electron attaching to the target species, forming a negative ion. No user input is required.
- Select **Ionization** for electron impact reactions which result in secondary electrons being formed. Enter the **Energy loss** $\Delta\varepsilon$ (SI unit: V) for the reaction. This is also known as threshold energy for the reaction.

CROSS SECTION DATA

Enter the cross section data in two columns. Enter the **Electron energy** (SI unit: eV) and the **Collision cross section data** (SI unit: m^2). Also click the **Load from file** button (📂) to import data from a text file. The file must contain the two columns of data (**Electron energy** (SI unit: eV) in the first column and **Collision cross section data** (SI unit: m^2) in the second column).

Theory for the Boltzmann Equation, Two-Term Approximation Interface

The Boltzmann Equation, Two-Term Approximation Interface theory is described in this section:

- The Electron Energy Distribution Function
- Boltzmann Equation, Two-Term Approximation
- References for the Boltzmann Equation, Two-Term Approximation Interface

The Electron Energy Distribution Function

For computing the electron density and electron energy according to Equation 4-5 and Equation 4-6 transport properties and source coefficients are required. These coefficients can be calculated from collision cross-section data and the (unknown) *electron energy distribution function* (EEDF). The equation formulation used in the physics interface depends on whether the EEDF is an assumed function (Maxwellian, Druyvesteyn, or Generalized) or is to be explicitly computed by the Boltzmann equation, using the two-term approximation.

In order to remain consistent with Ref. 1, the unit of electron energy and mean electron energy is specified in volts, rather than electron volts. The two quantities can be converted to each other via the unit charge, which is the built-in variable `e_const` in COMSOL. So, the mean electron energy (in volts) is equal to the mean electron energy (in units of eV) divided by the unit charge. This makes the equations easier to follow while remaining consistent with the way units are interpreted in COMSOL.

Obtaining a physical solution for a fluid model of a plasma largely depends on how self-consistently the transport properties and source coefficients are calculated. By solving the two-term Boltzmann equation, maximum consistency is obtained between the kinetic and fluid description of the electron dynamics.

Solving the *Boltzmann equation* also makes it possible to calculate the drift velocity. The drift velocity is an important quantity because it depends on all the collisions which make up the plasma chemistry. The drift velocity is easy to measure, and comparing the experimental and simulated drift velocity is therefore straightforward. Indeed, collision cross sections are often inferred from drift velocity data.

Verifying that the experimental drift velocity agrees with the computed drift velocity is a good indicator that a given set of cross sections are correct. This physics interface makes it easy to compare the computed drift velocity to experimental data. A treatise on experimental measurements of the drift velocity is available in Ref. 2.

Consider the following expression

$$f(\varepsilon) = g\varphi^{-3/2}\beta_1\exp(-(\varepsilon\beta_2/\varphi)^g)$$

where the following definitions apply

$$\beta_1 = \Gamma(5/2g)^{3/2}\Gamma(3/2g)^{-5/2}, \beta_2 = \Gamma(5/2g)\Gamma(3/2g)^{-1}, 1 \le g \le 2$$

and φ is the mean electron energy. Depending on the value of the variable, g, the distribution function is either *Maxwellian* ($g = 1$), *Druyvesteyn* ($g = 2$), or *Generalized* ($1 < g < 2$). Mathematically, the distribution functions are defined as:

MAXWELLIAN EEDF

$$f(\varepsilon) = \varphi^{-3/2}\beta_1\exp(-(\varepsilon\beta_2/\varphi))$$

$$\beta_1 = \Gamma(5/2)^{3/2}\Gamma(3/2)^{-5/2}, \beta_2 = \Gamma(5/2)\Gamma(3/2)^{-1}$$

DRUYVESTEYN EEDF

$$f(\varepsilon) = 2\varphi^{-3/2}\beta_1\exp(-(\varepsilon\beta_2/\varphi)^2)$$

$$\beta_1 = \Gamma(5/4)^{3/2}\Gamma(3/4)^{-5/2}, \beta_2 = \Gamma(5/4)\Gamma(3/4)^{-1}$$

GENERALIZED EEDF

$$f(\varepsilon) = g\varphi^{-3/2}\beta_1\exp(-(\varepsilon\beta_2/\varphi)^g)$$

$$\beta_1 = \Gamma(5/2g)^{3/2}\Gamma(3/2g)^{-5/2}, \beta_2 = \Gamma(5/2g)\Gamma(3/2g)^{-1}, 1 \le g \le 2$$

Boltzmann Equation, Two-Term Approximation

The Boltzmann equation describes the evolution of a distribution function, f, in six-dimensional phase space:

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla f - \frac{e}{m}(\mathbf{E} \cdot \nabla_{\mathbf{v}} f) = C[f] ,$$

where **v** is the velocity coordinates, m is the electron mass, **E** is the electric field, $\nabla_{\mathbf{v}}$ is the velocity gradient operator, and $C[f]$ is the rate of change in f due to collisions. To be able to solve the Boltzmann equation and thus compute the EEDF, drastic simplifications are necessary. It is assumed that the electric field and the collision probabilities are spatially uniform. The Boltzmann equation is then written in terms of spherical coordinates in the velocity space and f is expanded in spherical harmonics. The series is truncated after the second term and the so-called two-term approximation of f is

$$f(v, \cos\theta, z, t) = f_0(v, z, t) + f_1(v, z, t)\cos\theta$$

where f_0 is the isotropic part of f, f_1 is an anisotropic perturbation, v is the magnitude of the velocity, θ is the angle between the velocity and the field direction, and z is the position along this direction.

The last piece of simplification consists of separating the energy dependence of the EEDF from its time and space dependence. For steady state cases the separation can be formally written as

$$f_{0,1}(\varepsilon, z, t) = \frac{1}{2\pi\gamma^3} F_{0,1}(\varepsilon) n(z, t)$$

It is also possible to solve for the temporal evolution of F_0 in which case

$$f_0(\varepsilon, z, t) = \frac{1}{2\pi\gamma^3} F_0(\varepsilon, t) n(z, t)$$

For both stationary and time-dependent cases, F_1 can be assumed to follow the electric field instantaneously or to be in the limit of a high-frequency oscillating electric field. $F_{0,1}$ is an energy distribution function that satisfies the following normalization

$$\int_0^\infty \varepsilon^{1/2} F_0 d\varepsilon = 1$$

Outside this section, for simplicity, f is used to refer to any EEDF. It is important to keep in mind that when the Boltzmann equation in the two term approximation is solved the EEDF is F_0 as defined in this section.

Using the abovementioned approximations and after some manipulations, the Boltzmann equation can be written in the form of a 1D convection-diffusion-reaction equation

$$\frac{\sqrt{\varepsilon}}{N}\frac{\partial F_0}{\partial t} + \frac{\partial}{\partial \varepsilon}\left(WF_0 - D\frac{\partial F_0}{\partial \varepsilon}\right) = S \ .$$

For more details, see Ref. 1 and Ref. 5. This equation is somewhat special because the source term is nonlocal and the convection and diffusion coefficients depend on the integral of the solution. The different terms, including the effects of electron-electron collisions, are presented below:

$$W = -\gamma\varepsilon^2\sigma_\varepsilon - 3a\left(\frac{n_e}{N_n}\right)A_1$$

$$D = \frac{\gamma}{3}\left(\frac{E}{N_n}\right)^2\left(\frac{\varepsilon}{\tilde{Q}}\right) + \frac{\gamma k_b T}{q}\varepsilon^2\sigma_\varepsilon + 2a\left(\frac{n_e}{N_n}\right)(A_2 + \varepsilon^{3/2}A_3)$$

The following definitions apply

$$a = \frac{q^2\gamma}{24\pi\varepsilon_0^2}\ln\Lambda$$

$$\Lambda = \frac{12\pi(2\varepsilon_0 e\bar{\varepsilon}/3)^{3/2}}{q^3 n_e^{1/2}} \tag{3-1}$$

$$\sigma_m = \sum_{k = \text{all}} x_k\sigma_k$$

$$\sigma_\varepsilon = \sum_{k = \text{elastic}} 2\left(\frac{m_e}{M}\right)x_k\sigma_k$$

$$\tilde{\sigma}_m = \sigma_m + \frac{\lambda}{\varepsilon^{1/2}}$$

$$A_1 = \int_0^\varepsilon u^{1/2}F_0(u)du$$

$$A_2 = \int_0^{\varepsilon} u^{3/2} F_0(u) du$$

$$A_3 = \int_{\varepsilon}^{\infty} F_0(u) du .$$

$$Q = \tilde{\sigma}_m$$

The approach presented until now is valid in situations where the electric field is constant on the time scale of the collisions (referred as the DC limit). For high-frequency oscillating electric fields, in conditions such that the characteristic frequency for energy transfer is much lower than the field frequency (referred to as the HF limit), the same approach can be used with the following modification for Q:

$$Q = 2\frac{\tilde{\sigma}_m^2 + q_w{}^2}{\tilde{\sigma}_m^2}$$

where q_w is the equivalent cross section for AC field oscillations

$$q_w = \frac{\omega}{N_n \gamma \varepsilon^{1/2}} \frac{1}{} .$$

Here:

- $\gamma = (2q/m_e)^{1/2}$ (SI unit: $C^{1/2}/kg^{1/2}$)
- m_e is the electron mass (SI unit: kg)
- $\varepsilon = (v/\gamma)^2$ is energy (SI unit: V)
- σ_ε is the total elastic collision cross section (SI unit: m^2)
- σ_m is the total momentum collision cross section (SI unit: m^2)
- $\tilde{\sigma}_m$ is the normalized total momentum collision cross section (SI unit: m^2)
- q is the electron charge (SI unit: C)
- ε_0 is the permittivity of free space (SI unit: F/m)
- T is the temperature of the background gas (SI unit: K)
- k_b is the Boltzmann constant (SI unit: J/K)
- n_e is the electron density (SI unit: $1/m^3$)
- N_n is the background gas density (SI unit: $1/m^3$)

- Λ is the Coulomb logarithm, and
- M is the mass of the target species (SI unit: kg).

The source term, S represents energy loss due to inelastic collisions. Because the energy loss due to an inelastic collision is quantized, the source term is nonlocal in energy space. The source term can be decomposed into four parts where the following definitions apply:

$$S = \sum_{k\,=\,\text{inelastic}} C_k - \gamma\lambda\varepsilon^{1/2}F_0$$

$$C_{k\,=\,\text{excitation}} = -\gamma x_k[\varepsilon\sigma_k(\varepsilon)F_0(\varepsilon) - (\varepsilon+\Delta\varepsilon_k)\sigma_k(\varepsilon+\Delta\varepsilon_k)F_0(\varepsilon+\Delta\varepsilon_k)]$$

$$C_{k\,=\,\text{att}} = -\gamma x_k\varepsilon\sigma_k(\varepsilon)F_0(\varepsilon)$$

$$C_{\text{ion}} = -\gamma x_k[\varepsilon\sigma_k(\varepsilon)F_0(\varepsilon) - (\varepsilon+\Delta\varepsilon_k)\sigma_k(\varepsilon+\Delta\varepsilon_k)F_0(\varepsilon+\Delta\varepsilon_k)]$$
$$+\delta\gamma x_k\int_0^{\infty} u\sigma_k(u)F_0(u)du$$

where x_k is the mole fraction of the target species for reaction k, σ_k is the collision cross section for reaction k, $\Delta\varepsilon_k$ is the energy loss from collision k, and δ is the delta function at $\varepsilon = 0$. The term C_{ion} changes slightly when equal energy sharing is used:

$$C_{\text{ion}} = -\gamma x_k[\varepsilon\sigma_k(\varepsilon)F_0(\varepsilon) - 4(2\varepsilon+\Delta\varepsilon_k)\sigma_k(2\varepsilon+\Delta\varepsilon_k)F_0(2\varepsilon+\Delta\varepsilon_k)]\,.$$

Note the factor of 4 differs from the factor of 2 used in Ref. 1, as was later corrected in Ref. 4. The term, λ is a scalar-valued renormalization factor that ensures that the EEDF has the following property:

$$\int_0^{\infty} F_0\varepsilon^{1/2}d\varepsilon = 1 \tag{3-2}$$

An ODE is implemented to solve for the value of λ such that Equation 3-2 is satisfied. The rate coefficients are computed from the EEDF by way of the following integral:

$$k_k = \gamma\int_0^{\infty}\varepsilon\sigma_k(\varepsilon)F_0(\varepsilon)d\varepsilon$$

The mean electron energy is defined by the integral

$$\bar{\varepsilon} = \int_0^\infty F_0 \varepsilon^{3/2} d\varepsilon \tag{3-3}$$

In order to set the mean electron energy to a specific setpoint, a second Lagrange multiplier is introduced to solve for the reduced electric field, such that Equation 3-3 is satisfied. The weak form of the constraint is

$$\left(\int_0^\infty F_0 \varepsilon^{3/2} d\varepsilon - \varepsilon_{sp}\right)\left(\frac{\tilde{E}}{N_n}\right) = 0$$

where tilde denotes test function.

The *reduced transport properties* associated to a DC transport are computed using the following integrals

$$\mu_e N_n = -\left(\frac{\gamma}{3}\right)\int_0^\infty \frac{\varepsilon}{\tilde{\sigma}_m}\left(\frac{\partial F_0}{\partial \varepsilon}\right) d\varepsilon$$

$$D_e N_n = \left(\frac{\gamma}{3}\right)\int_0^\infty \frac{\varepsilon}{\tilde{\sigma}_m} F_0 d\varepsilon$$

$$\mu_\varepsilon N_n = -\left(\frac{\gamma}{3\bar{\varepsilon}}\right)\int_0^\infty \frac{\varepsilon^2}{\tilde{\sigma}_m}\left(\frac{\partial F_0}{\partial \varepsilon}\right) d\varepsilon$$

$$D_\varepsilon N_n = \left(\frac{\gamma}{3\bar{\varepsilon}}\right)\int_0^\infty \frac{\varepsilon^2}{\tilde{\sigma}_m} F_0 d\varepsilon$$

In the HF limit the real and imaginary part of the mobility are available and are used to computed the power absorbed by the electrons and the plasma conductivity

$$\mu_r N_n = -\left(\frac{\gamma}{3}\right)\int_0^\infty \frac{\tilde{\sigma}_m \varepsilon}{\tilde{\sigma}_m^2 + q^2}\left(\frac{\partial F_0}{\partial \varepsilon}\right) d\varepsilon$$

$$\mu_i N_n = \left(\frac{\gamma}{3}\right)\int_0^\infty \frac{q \varepsilon}{\tilde{\sigma}_m^2 + q^2}\left(\frac{\partial F_0}{\partial \varepsilon}\right) d\varepsilon .$$

The *power absorbed* by the electrons from the electric field in the DC limit is given by

$$P/N_n = (\mu N_n)(E/N_n)^2$$

and in the HF limit by

$$P/N_n = (\mu_r N_n)(E/N_n)^2 .$$

The *elastic power loss* is defined as

$$P_{el}/N_n = \sum_{k\,=\,\text{elastic}} \gamma x_k \frac{2m_e}{M_k}\int_0^{\infty}\left[\sigma_k\left(\varepsilon^2 F_0 + \frac{k_B T}{e}\left(\frac{\partial F_0}{\partial \varepsilon}\right)\right)\right]d\varepsilon .$$

The *inelastic power loss* is defined as

$$P_{\text{inel}}/N_n = \sum_{k\,=\,\text{inelastic}} x_k k_k \Delta\varepsilon_k .$$

The *growth power* associated with the apparent energy loss due to electrons appearing and disappearing in ionization and attachment is given by

$$P_{\text{growth}}/N_n = ((\nu_{\text{ion}} - \nu_{\text{att}})/N_n)\bar{\varepsilon} ,$$

where ν_{ion} and η_{att} are the *total ionization frequency* and the *total attachment frequency* defined as:

$$\frac{\nu_{ion}}{N_n} = \gamma\int_0^{\infty} \sum_{k\,=\,\text{ionization}} \varepsilon x_k \sigma_k(\varepsilon) F_0(\varepsilon) d\varepsilon$$

$$\frac{\eta_{at}}{N_n} = \gamma\int_0^{\infty} \sum_{k\,=\,\text{attachment}} \varepsilon x_k \sigma_k(\varepsilon) F_0(\varepsilon) d\varepsilon .$$

Note that the solution of the Boltzmann equation in this approach should satisfy

$$(P/N_n) = (P_{\text{el}}/N_n) + (P_{\text{inel}}/N_n) + (P_{\text{growth}}/N_n) .$$

The *drift velocity* in the DC limit is computed from the following integral:

$$w = -\left(\frac{\gamma}{3}\right)\left|\left(\frac{E}{N_n}\right)\right|\int_0^{\infty}\frac{\varepsilon}{\tilde{\sigma}_m}\left(\frac{\partial F_0}{\partial \varepsilon}\right)d\varepsilon$$

and in the HF limit is given by

$$w = -\left(\frac{\gamma}{3}\right)\left|\left(\frac{E}{N_n}\right)\right|\int_0^{\infty}\varepsilon\frac{\tilde{\sigma}_m - iq}{\tilde{\sigma}_m^2 + q^2}\left(\frac{\partial F_0}{\partial\varepsilon}\right)d\varepsilon = -(\mu_r + i\mu_i)E$$

The *drift velocity* is an important quantity because it provides a convenient way of comparing the results of the Boltzmann equation to experimental data.

In high frequency plasmas it is common to use the following approximation of the electron momentum equation

$$\frac{\partial w}{\partial t} + \nu_{\mathrm{eff}}w = -\phi\frac{q}{m_e}E$$

where ν_{eff} is an effective collision frequency and ϕ is a factor that is needed in order to be coherent with the drift velocity obtained from the Boltzmann equation approach here used. Substituting the HF drift velocity in the electron momentum equation it is obtained

$$\nu_{eff} = -\frac{\mu_r}{\mu_i}\omega$$

$$\phi = -\frac{\mu_r^2 + \mu_i^2}{\mu_i}\frac{m_e\omega}{e}.$$

In the frequency domain it is possible to derive the plasma conductivity as

$$\sigma = \frac{q^2 n_e}{m_e(\nu_{\mathrm{eff}} + j\omega)}\phi.$$

The *Townsend coefficients* are computed using

$$\alpha_k/N_n = \frac{(E/N_n)(\nu_k/N_n)}{(P/N_n)}$$

where ν_{k} is the frequency associated with the reaction rate k_{k}. The *Townsend ionization coefficient* for a given gas mixture is given by

$$\alpha/N_n = \frac{(E/N_n)(\nu_{\mathrm{ion}}/N_n)}{(P/N_n)}$$

and the *Townsend attachment coefficient* is given by

$$\eta/N_n = \frac{(E/N_n)(\nu_{\mathrm{att}}/N_n)}{(P/N_n)}.$$

When the distribution function is assumed the electric field is computed using the power balance

$$\left(\frac{E}{N_n}\right) = \sqrt{\frac{(P_{\mathrm{el}}/N_n) + (P_{\mathrm{inel}}/N_n) + (P_{\mathrm{growth}}/N_n)}{\mu N_n}}$$

A plot of the drift velocity for different distribution functions versus the reduced electric field for oxygen is shown in Figure 3-1. Experimental data is also included in the plot.

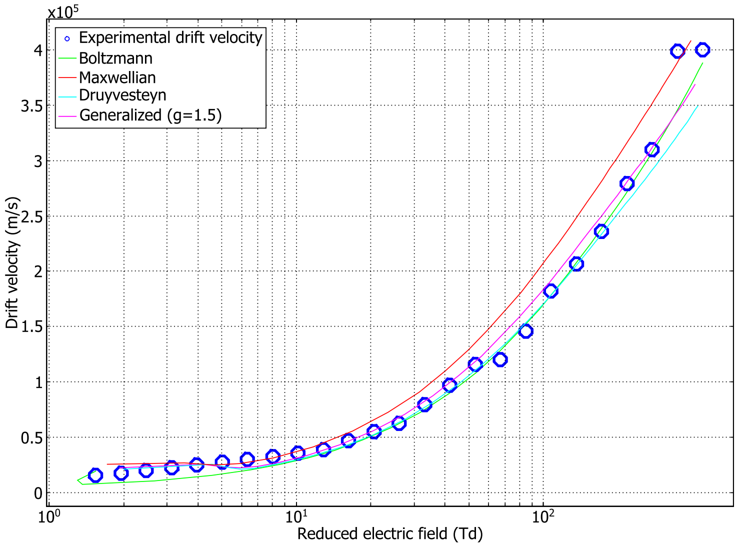


Figure 3-1: Plot of experimental and computed drift velocity for different distribution functions.

The computed transport coefficients have little dependence on the type of EEDF. However, the rate coefficients for excitation and ionization processes are highly dependent on the shape of the EEDF, due to the exponential drop off in the population of electrons at energies exceeding the activation threshold. Figure 3-2 plots the ionization rate coefficient for oxygen for the types of distribution function.

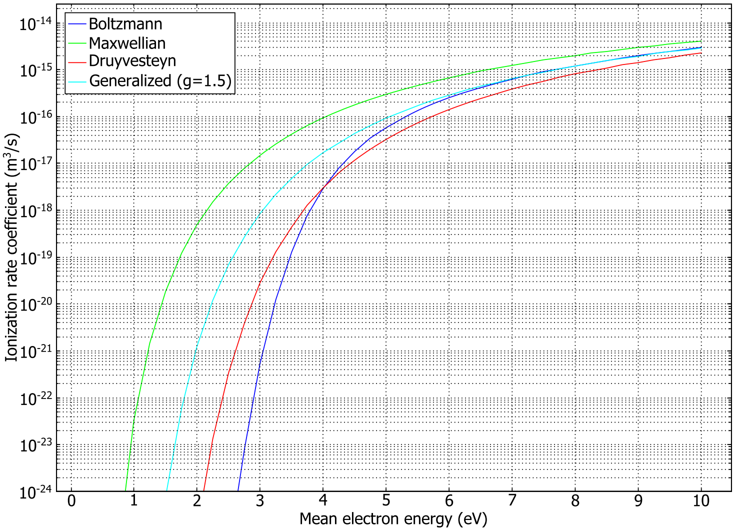


Figure 3-2: Plot of ionization coefficient vs. mean electron energy for different distribution functions.

References for the Boltzmann Equation, Two-Term Approximation Interface

1. G.J.M. Hagelaar and L. C. Pitchford, “Solving the Boltzmann equation to obtain electron transport coefficients and rate coefficients for fluid models”, *Plasma Sources Science and Technology*, vol. 14, pp. 722–733, 2005.

2. J. Dutton, “A Survey of Electron Swarm Data”, *J. Phys. Chem. Ref. Data*, vol. 4, pp. 577–866, 1975.

3. H.B. Milloy and R.O. Watts, “On the Validity of the Two-term Approximation in the Solution of Boltzmann’s Equation for Electron Motion”, *Aust. J. Phys.*, vol 30, pp 73, 1977.

4. G. J. M. Hagelaar, “Coulomb collisions in the Boltzmann equation for electrons in low-temperature gas discharge plasmas”, *Plasma Sources Science and Technology*, vol. 254, no. 1, 2015.

5. Z. Donko and N. Dyatko, “First-Principles Particle Simulation and Boltzmann Equation of Negative Differential Conductivity and Transient Negative Effects in Xenon”, *Eur.Phys. J. D* vol. 70, pp. 135–146, 2016.

4

The Drift Diffusion Interface

This chapter describes the underlying electron transport theory for the Drift Diffusion interface, found under the **Plasma>Species Transport** branch ( ).

In this chapter:

The Drift Diffusion Interface

The **Drift Diffusion (dd)** interface ( ), found under the **Plasma>Species Transport** branch ( ), solves for the electron density and mean electron energy for any type of plasma. A wide range of boundary conditions are available to handle secondary emission, thermionic emission, and wall losses.

When this physics interface is added, these default nodes are also added to the **Model Builder** — **Drift Diffusion Model**, **Insulation** (the default boundary condition), and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions and velocity. You can also right-click **Drift Diffusion** to select physics features from the context menu.

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `dd`.

ELECTRON PROPERTIES

Select the **Use reduced electron transport properties** or **Compute tensor electron transport properties**, or **Include thermal diffusion** check boxes as needed and select the model for the **Mean electron energy** — **Local energy approximation** (default), **Local field approximation** or **Fix mean electron energy**.

Use Reduced Electron Transport Properties

Select the **Use reduced electron transport properties** check box to specify the electron mobility, diffusivity, energy mobility and energy diffusivity in reduced form. The neutral number density is then specified in the **Drift Diffusion Model** node. The electron transport properties are computed from the reduced transport properties using:

$$\mu_e = \frac{\mu_{\mathrm{red}}}{N_n}$$

where N_n is the user-defined neutral number density.

It is sometimes more convenient to use reduced transport properties, since they can be output from The Boltzmann Equation, Two-Term Approximation Interface.

Compute Tensor Electron Transport Properties

Select the **Compute tensor electron transport properties** check box to automatically compute the tensor form of the electron mobility, diffusivity, energy mobility and energy diffusivity when a static magnetic field is present. In this case the magnetic flux density is specified in the Drift Diffusion Model feature.

Include Thermal Diffusion

Select the **Include thermal diffusion** check box to add an additional term to the definition of the electron current due to gradients in the electron diffusivity. If the diffusivity is a constant then including this does not affect the solution. It is only necessary to include this term if the electron diffusivity is a function of the electron temperature, and there are significant gradients in the electron temperature.

Mean Electron Energy

Select **Local energy approximation** (default) to solve the mean electron energy equation self-consistently with the continuity, momentum, and Poisson's equations, and to use the mean electron energy to parameterize transport and source coefficients. This is the most numerical demanding option to find the mean electron energy because of the strong coupling between the mean electron energy and the electromagnetic fields.

If **Local field approximation** is selected it is assumed that transport and source coefficients are well parameterized through the reduced electric field (E/N_n). The relation between the reduced electric field and the mean electron energy needs to be provided in the section **Mean Electron Energy Specification** in the **Plasma Model** node. When using the local field approximation the fluid equation for the mean electron energy is not solved, which reduces significantly the complexity of the numerical problem. The local field approximation is valid in a situation where the rate of electron energy gain from the electric field is locally balanced by the energy loss rate. When this condition is met the electrons are said to be in local equilibrium with the electric field and the electron mean properties can be expressed as a function of the reduced electric field.

Select **Fix mean electron energy** to fix the mean electron energy to its initial value. This can be useful in some situations because the strong coupling between the mean electron energy and the electromagnetic fields is removed. This allows for non-self-consistent models to be created quickly, since problems where the mean electron energy is fixed are easier to solve numerically.

STABILIZATION

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

If the **Formulation** is set to **Finite element, log formulation (linear shape function)** or **Finite element, log formulation (quadratic shape function)** then the solver can run into difficulties when the species mass fractions approach zero. The **Source stabilization** check box (selected by default) adds an additional source term to the rate expression for each species. In the ζ text field, enter a tuning parameter for the source stabilization. The default value is 1. This value is usually good enough. If the plasma is high pressure (atmospheric) then it can help to lower this number to somewhere in the range of 0.25–0.5. For information on stabilization see Stabilization.

DISCRETIZATION

Select **Formulation** — **Finite volume (constant shape function)**, **Finite element, log formulation (linear shape function)** (the default) to solve the equations in logarithmic form, **Finite element (linear shape function)**, **Finite element, log formulation (quadratic shape function)**, or **Finite element (quadratic shape function)**. The Log formulation solves for the log of the dependent variables, ensuring that the mass fraction of any of the species is never lower than zero. This makes it more numerically stable but increases the nonlinearity of the equation system, and as such the model might take slightly longer to solve. The linear formulation solves the equations in the original form.

DEPENDENT VARIABLES

The dependent variables (field variables) are the **Electron solution variable** and **Electron energy solution variable**. The name can be changed but the names of fields and dependent variables must be unique within a model. The physical meanings of **Electron solution variable** and **Electron energy solution variable** change depending on the formulation used. For example, when log formulations are used, **Electron solution**

variable is the logarithm of the electron density; whereas for other cases, it is the electron density

- Domain, Boundary, and Pair Nodes for the Drift Diffusion Interface
- Stabilization
- Theory for the Drift Diffusion Interface

Domain, Boundary, and Pair Nodes for the Drift Diffusion Interface

The Drift Diffusion Interface has these domain, boundary, and pair nodes, listed in alphabetical order, available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or right-click to access the context menu (all users).

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

In the *COMSOL Multiphysics Reference Manual* see Table 2-4 for links to common sections and Table 2-5 to common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

The boundary conditions are used for the electron density and mean electron energy, which make it easy to specify how the electrons interact with the wall. Secondary emission, thermionic emission and reflection can all be specified.

It is also possible to impose the specific value of the electron density or mean electron energy on a boundary. This, however is not recommended as imposing such a value on a solid surface tends not to fit the physics of the interaction between a solid surface and electrons.

- Drift Diffusion Model
- Initial Values
- Electron Density and Energy
- Electron Outlet
- Electron Production Rate
- Flux
- General Power Deposition
- Inductive Power Deposition
- Insulation
- Microwave Power Deposition
- Velocity
- Wall

Drift Diffusion Model

The **Drift Diffusion Model** adds the equations for electron transport in a plasma. Specify expressions for the electron diffusivity, mobility, electron energy diffusivity and electron energy mobility. There are also model inputs for the electric potential and collisional power loss.

MODEL INPUTS

This section contain fields and values that are inputs to expressions that define material properties. If such user-defined property groups are added, the model inputs are listed here.

- **Electric potential** V (SI unit: V). Used to compute the electric field which results in migration of the electrons.
- **Collisional power loss** S_{en} (SI unit: $\mathrm{W/m^3}$). Represents the energy loss due to elastic and inelastic collisions.
- When the Use Reduced Electron Transport Properties check box is selected, choose **Neutral number density** N_n (SI unit: $1/\mathrm{m}^3$) to compute the electron transport.
- When the Compute Tensor Electron Transport Properties check box is selected, choose **Magnetic flux density** B (SI unit: T) to compute the tensor form of the electron transport properties when a static magnetic field is present.

The electric potential typically comes from an **Electrostatics** interface and the collisional power loss is typically defined by The Heavy Species Transport Interface.

ELECTRON DENSITY AND ENERGY

This section is not available if the Compute Tensor Electron Transport Properties check box is selected.

For all other choices, select an option from the **Electron transport properties** list — Specify Mobility Only (the default), Specify All, or Use Lookup Tables.

> The other transport properties are computed using Einstein's relation, see Tensor Transport Properties for more details.

Specify Mobility Only

When no check boxes, or the Include Thermal Diffusion or Mean Electron Energy check boxes are selected, and if **Specify mobility only** is selected, enter a value or expression for the **Electron mobility** μ_e (SI unit: $m^2/(V\cdot s)$). Select **Isotropic**, **Diagonal**, **Symmetric**, or **Anisotropic** based on the model. The electron diffusivity, energy mobility and energy diffusivity are automatically computed using Einstein's relation for a Maxwellian EEDF:

$$\mathbf{D}_e = \mu_e T_e,\ \mu_\varepsilon = \left(\frac{5}{3}\right)\mu_e,\ \mathbf{D}_\varepsilon = \mu_\varepsilon T_e \tag{4-1}$$

If the Use Reduced Electron Transport Properties check box is selected, enter the **Reduced electron mobility** $\mu_e N_n$ (SI unit: $1/(V\cdot m\cdot s)$). The actual electron mobility is then computed by dividing the reduced electron mobility by the neutral number density. The electron diffusivity, energy mobility and energy diffusivity are then computed as in Equation 4-1.

Specify All

For **Specify all** enter values or expressions for all of the properties, which can be either scalars or tensors. The appropriate values for the transport properties can be computed with The Boltzmann Equation, Two-Term Approximation Interface. For all of the properties, select **Isotropic**, **Diagonal**, **Symmetric**, or **Anisotropic** from the list based on the model, then enter values or expressions for all of these properties:

- **Electron mobility** μ_e (SI unit: $m^2/(V\cdot s)$)
- **Electron diffusivity** $\mathbf{D}_e$ (SI unit: m^2/s)
- **Electron energy diffusivity** $\mathbf{D}_{en}$ (SI unit: m^2/s)
- **Electron energy mobility** μ_{en} (SI unit: $m^2/(V\cdot s)$)

If the Use Reduced Electron Transport Properties check box is selected, then enter:

- **Reduced electron mobility** $\mu_e N_n$ (SI unit: $1/(V\cdot m\cdot s)$)
- **Reduced electron diffusivity** $D_e N_n$ (SI unit: $1/(m\cdot s)$)

- **Reduced electron energy diffusivity** $\mathbf{D}_{\mathrm{en}}N_n$ (SI unit: 1/(m·s))
- **Reduced electron energy diffusivity** $\mu_{\mathrm{en}}N_n$ (SI unit: 1/(V·m·s))

Use Lookup Tables

For **Use lookup tables** enter or load a lookup table with the transport properties as listed above versus mean electron energy (eV).

> If the Use Reduced Electron Transport Properties check box is selected, the appropriate values for the transport properties can be computed with The Boltzmann Equation, Two-Term Approximation Interface and written out to a text file. The text file can then be loaded in for each electron transport property.

MEAN ELECTRON ENERGY SPECIFICATION

This section is available when the **Mean electron energy** is set to **Local field approximation**. Select how to provide the relation between the reduce electric field and the mean electron energy from the **Specify using** list — **Expression** or **Use lookup table**.

DC ELECTRON MOBILITY

This section is available if the Compute Tensor Electron Transport Properties check box is selected. Enter a value for the **DC electron mobility**, μ_{dc} (SI unit: m^2/(V·s)).

The section is also available when both the Compute Tensor Electron Transport Properties and the Use Reduced Electron Transport Properties check boxes are chosen. Then enter a value or expression for the **Reduced DC electron mobility**, $\mu_{\mathrm{dc}}N_n$ (SI unit: 1/(V·m·s)).

Initial Values

The **Initial Values** node adds initial values for the electron density and mean electron energy.

INITIAL VALUES

Enter values or expressions for the initial value of the **Initial electron density** $n_{\mathrm{e},0}$ (SI unit: $1/\mathrm{m}^3$) and **Initial mean electron energy** e_0 (SI unit: V). Enter an approximate guess for the mean electron energy in the reactor. COMSOL Multiphysics automatically computes the exact mean electron energy after solving but the closer the guess can be set to the expected value the easier it is to find a solution.

Electron Production Rate

Use the **Electron Production Rate** node to add sources and sinks to the number density of electrons. Ionization processes lead to an increase in the number density of electrons. Similarly, attachment leads to a decrease in the number density of electrons. A production rate for new electrons can be specified with this feature.

ELECTRON PRODUCTION RATE

Select or enter a value for a **Production rate** R_e (SI unit: $1/m^3{\cdot}s$) of new electrons. The value typically comes from The Heavy Species Transport Interface.

General Power Deposition

Use the **General Power Deposition** to add an arbitrary heat source to the electrons.

GENERAL POWER DEPOSITION

Enter a value or expression for the **General heat source** Q (SI unit: W/m^3). This could be a constant number or come from another physics interface.

Velocity

Use the **Velocity** node to include the effect of convection due to a moving gas. In nearly all practical situations, convection of electrons due to gas movement is negligible.

VELOCITY

The **Velocity field** μ (SI unit: m/s) value or expression can be user-defined or come from another physics interface, typically **Laminar Flow**.

Inductive Power Deposition

The **Inductive Power Deposition** node requires a license for the AC/DC Module. Use it to add a heat source to the electrons from induction currents. This is typically used in inductively coupled plasmas.

INDUCTIVE POWER DEPOSITION

Enter a **Heat source** Q (SI unit: W/m^3) for the electrons due to induction currents. Typically this comes from a **Magnetic Fields** interface.

Microwave Power Deposition

The **Microwave Power Deposition** node requires a license for the RF Module. Use it to add microwave heating to the electrons. This is typically used in wave heated discharges.

MICROWAVE POWER DEPOSITION

Enter a **Heat source** Q (SI unit: W/m^3) for the electrons due to electromagnetic waves. This typically comes from an **Electromagnetic Waves** interface.

Insulation

Use the **Insulation** boundary condition for boundaries that correspond to inlets or outlets for the background gas and it is the default boundary condition for the **Drift Diffusion** interface. It is also available on the **Pairs** menu. This boundary condition requires no user input. The boundary condition sets the normal component of the electron and electron energy flux to zero:

$$-\mathbf{n}\cdot\Gamma_e = 0 \text{ and} -\mathbf{n}\cdot\Gamma_\varepsilon = 0$$

Electron Density and Energy

The **Electron Density and Energy** boundary condition fixes the electron density to a specific number density or the mean electron energy to a specific energy. It is also available on the **Pairs** menu. This boundary condition should, in general, be avoided in favor of the **Wall** boundary condition.

$$n_e = n_{ew} \text{ and} \bar{\varepsilon} = \bar{\varepsilon}_0$$

Due to the fact that the log of the electron density is solved for, specifying an electron density of zero is not allowed.

For background theory see The Electron Density and Energy Boundary Condition.

ELECTRON DENSITY AND ENERGY

Select the **Fix electron density** check box to enter a value for the **Electron density at the wall** n_{ew} (SI unit: $1/\mathrm{m}^3$).

Select the **Fix mean electron energy** check box to enter a value for the **Mean electron energy at the wall** e (SI unit: V).

Wall

Use the **Wall** boundary condition when the plasma is in contact with a solid surface. This also allows for secondary and thermionic emission effects from the solid surfaces. Use the **Wall** boundary condition to describe how the electrons interact with the wall

GENERAL WALL SETTINGS

Enter a value for the **Reflection coefficient** r_{e} (dimensionless) on the selected boundaries. This must be a number between 0 and 1.

The **Include migration effects** check box should be checked if the electron flux to the wall is due to the electric field. This is usually not the case because the plasma potential is positive with respect to the walls, so the electron flux to the wall due to the electric field is zero. In models where the electrostatic field changes very rapidly (like with dielectric barrier discharges for example), the electron flux to the wall can be due to the electric field, in which case this option should be activated.

ELECTRON DENSITY WALL SETTINGS

The **Use wall for electron density** check box is selected by default and deactivates the wall boundary condition for the electron density only. When activated, the flux of electrons lost to the wall is self-consistently computed based on the electron temperature and electron density adjacent to the wall. When this is deactivated, the flux term is no longer applied for the corresponding equation, and the software will impose a zero flux boundary condition for that dependent variable.

Select or enter a value for the **Secondary emission flux** (SI unit: $1/\mathrm{m}^2\cdot\mathrm{s}$)). The secondary emission flux can either be a user-defined expression or come from another physics interface, typically The Heavy Species Transport Interface. When using this boundary condition in the **Plasma** interface, this setting is not editable, and the secondary emission flux comes from the surface reactions.

Select or enter a value for the **Thermal emission flux** (SI unit: $1/\mathrm{m}^2\cdot\mathrm{s}$)). If electrons are being emitted from the wall due to thermal emission then enter an expression for the thermionic flux. Typically this only occurs when the wall is at a very high temperature.

ELECTRON ENERGY WALL SETTINGS

The **Use wall for electron energy** check box is selected by default and deactivates the wall boundary condition for the electron energy. This is useful in order to use the wall condition for the electron density but impose the value of the mean electron energy on the same boundaries.

Select or enter a value for the **Secondary emission energy flux**, which can either be a user-defined expression or come from another physics interface, typically The Heavy Species Transport Interface. When using this boundary condition in the **Plasma** interface, this setting is not editable, and the secondary emission energy flux comes from the surface reactions.

Select or enter a value for the **Mean thermionic energy**. If electrons are being emitted from the wall due to thermionic emission then enter an expression for the mean energy of the emitted electrons. The mean electron energy of the emitted electrons is usually a function of the work function of the surface.

For background theory, see The Wall Boundary Condition.

Flux

Use the **Flux** node to select whether the boundary condition is imposed for electron density or electron energy or both. The wall boundary condition should be used in preference to the flux boundary conditions for walls. A user-defined electron **Flux** can be specified by the following:

$$-\mathbf{n}\cdot\Gamma_e = \Gamma_{e,\,\mathrm{in}} \text{ and} -\mathbf{n}\cdot\Gamma_\varepsilon = \Gamma_{\varepsilon,\,\mathrm{in}}$$

FLUX

Select the **Use flux for electron density** check box to activate the boundary condition for electron density. Then enter a value for the **Electron density influx** r_{e}(SI unit: $1/(\mathrm{m}^2{\cdot}\mathrm{s})$).

Select the **Use flux for electron energy density** to activate the boundary condition for electron energy. Then enter a value for the **Electron energy density influx** r_{en} (dimensionless).

Electron Outlet

Use the **Electron Outlet** boundary condition for boundaries which correspond to outlets for the background gas. This boundary condition requires no user input. The boundary condition sets the normal component of the electron and electron energy flux to zero:

$$-\mathbf{n}\cdot\Gamma_e = 0 \text{ and} -\mathbf{n}\cdot\Gamma_\varepsilon = 0$$

Theory for the Drift Diffusion Interface

The Drift Diffusion Interface theory is described in this section:

Introduction to Drift Diffusion Theory

In general, electron transport is described by the Boltzmann equation, which is a nonlocal continuity equation in phase space ($\mathbf{r}$, $\mathbf{u}$). The Boltzmann equation is an extremely complicated integrodifferential equation and solving it in an efficient manner is not currently possible. The Boltzmann equation can be approximated by fluid equations by multiplying by a weighting function and then integrating over velocity space. This reduces the governing equations to a three-dimensional, time-dependent problem. The fluid equations describe the electron number density, the mean electron momentum and the mean electron energy as a function of configuration space and time. The rate of change of the electron density is described by:

$$\frac{\partial}{\partial t}(n_e) + \nabla \cdot \Gamma_e = R_e \qquad (4\text{-}2)$$

where n_e is the electron density, Γ_e is the electron flux vector and R_e is either a source or a sink of electrons. The rate of change of the electron momentum is described by:

$$\frac{\partial}{\partial t}(n_e m_e \mathbf{u}_e) + \nabla \cdot n_e m_e \mathbf{u}_e \mathbf{u}_e^T = -(\nabla \cdot \mathbf{p}_e) + q n_e \mathbf{E} - n_e m_e \mathbf{u}_e \nu_m \qquad (4\text{-}3)$$

where m_e is the electron mass (SI unit: kg), $\mathbf{u}_e$ is the drift velocity of the electrons (SI unit: m/s), $\mathbf{p}_e$ is the electron pressure tensor (SI unit: Pa), q is the electron change (SI unit: s A), $\mathbf{E}$ is the electric field (SI unit: V/m) and ν_m is the momentum transfer frequency (SI unit: 1/s). The rate of change of the electron energy density is described by:

$$\frac{\partial}{\partial t}(n_\varepsilon) + \nabla \cdot \Gamma_\varepsilon + \mathbf{E} \cdot \Gamma_e = S_{\mathrm{en}} \tag{4-4}$$

where n_ε is the electron energy density (V/m^3) and S_{en} is the energy loss or gain due to inelastic collisions (V/m^3 s).

The expression for the flux, Γ_e in Equation 4-2 is derived from the momentum conservation, Equation 4-3. Under the assumptions that the ionization and attachment frequencies and also the angular frequency are much less than the momentum transfer frequency, the first term on the left hand side of Equation 4-3 can be neglected. Under the assumption that the electron drift velocity is smaller than the thermal velocity, the second term on the left side of Equation 4-3 can also be neglected. For a Maxwellian distribution the pressure term, $\mathbf{p}_e$ can be replaced using the equation of state:

$$\mathbf{p}_e = n_e k_B T_e \mathbf{I}$$

where $\mathbf{I}$ is the identity matrix and T_e is the electron temperature. With these assumptions an expression for the electron drift velocity can be derived:

$$\mathbf{u}_e = -\frac{k_B}{m_e \nu_m}\nabla T_e - \frac{k_B T_e}{n_e m_e \nu_m}\nabla n_e + \frac{q}{m_e \nu_m}\mathbf{E}\,.$$

Defining the electron flux as:

$$\Gamma_e = n_e \mathbf{u}_e = -(\mu_e \bullet \mathbf{E}) n_e - \nabla(\mathbf{D}_e n_e)$$

where the electron mobility, μ_e (SI unit: m2/(V s)) is defined as:

$$\mu_e = \frac{q}{m_e \nu_m}$$

and the electron diffusivity, D_e (SI unit: m^2/s) is defined as:

$$D_e = \frac{k_B T_e}{m_e \nu_m}$$

The drift-diffusion approximation is suitable when the gas pressure is above about 10 mtorr (1.33 Pa) and the reduced electric field is not too high (typically less than 500 Townsends). Furthermore, the number density of charged species should be much less than the number density of the background gas. In other words, the discharge must be *weakly ionized*. The plasma must also be *collisional* which means that the mean free path between electrons and the background gas must be much less than the characteristic dimension of the system. This means the drift diffusion approximation is not suitable for modeling fusion plasmas.

The Drift Diffusion interface solves one equation for the electron density and one for the electron energy density.

The Drift Diffusion interface requires specification of the electron transport properties which are typically computed using The Boltzmann Equation, Two-Term Approximation Interface.

Electron Transport Theory

The equation for the electron density is given by

$$\frac{\partial}{\partial t}(n_e) + \nabla \cdot \Gamma_e = R_e \tag{4-5}$$

$$\Gamma_e = -(\mu_e \bullet \mathbf{E})n_e - \nabla(\mathbf{D}_e n_e)$$

Here (the subscript e refers to an electron:

- n_e denotes the electron density (SI unit: $1/\mathrm{m}^3$).
- R_e is the electron rate expression (SI unit: $1/(\mathrm{m}^3 \cdot \mathrm{s})$).
- μ_e is the electron mobility which is either a scalar or tensor (SI unit: $\mathrm{m}^2/(\mathrm{V} \cdot \mathrm{s})$).
- $\mathbf{E}$ is the electric field (SI unit: V/m).
- D_e is the electron diffusivity which is either a scalar or a tensor (SI unit: m^2/s).

The *electron energy density* is given by the following equation

$$\frac{\partial}{\partial t}(n_\varepsilon) + \nabla \cdot \Gamma_\varepsilon + \mathbf{E} \cdot \Gamma_e = S_{\mathrm{en}} + (Q + Q_{\mathrm{gen}})/q \tag{4-6}$$

$$\Gamma_\varepsilon = -(\mu_\varepsilon \bullet \mathbf{E})n_\varepsilon - \nabla(\mathbf{D}_\varepsilon n_\varepsilon)$$

where (the subscript ε refers to electron energy):

- n_ε is the electron energy density (SI unit: V/m^3).
- S_{en} is the energy loss/gain due to inelastic collisions (SI unit: V/(m^3·s)).
- Q is an external heat source (SI unit: W/m^3).
- Q_{gen} is a generalized heat source (SI unit: W/m^3).
- μ_ε is the electron energy mobility which is either a scalar or a tensor (SI unit: m^2/(V·s)).
- $\mathbf{E}$ is the electric field (SI unit: V/m).
- D_ε is the electron energy diffusivity (SI unit: m^2/s).

The mean electron energy, $\bar{\varepsilon}$ (SI unit: V) is computed through the expression

$$\bar{\varepsilon} = \frac{n_\varepsilon}{n_e} \tag{4-7}$$

Because of the high degree of nonlinearity inherent in the drift diffusion equation, the electron number density can span 10 orders of magnitude over a very small distance. In this region (the *plasma sheath*), the difference in the mobility and diffusivity between the ions and electrons creates a separation of space charge. This in turn produces a large electric field which can lead to a substantial increase in the mean electron energy. The best way of handling with this from a numerical point of view is to solve for the log of the electron number and energy density. This also prevents a divide by zero from occurring when equation Equation 4-7 is evaluated. So, letting $N_{\mathrm{e}} = \ln n_e$, equation Equation 4-5 becomes

$$e^{N_e}\frac{\partial}{\partial t}(N_e) + \nabla \cdot [-e^{N_e}(\mu_e \bullet \mathbf{E}) - e^{N_e}(\nabla(\mathbf{D}_e N_e))] = R_e$$

or

$$n_e\frac{\partial}{\partial t}(N_e) + \nabla \cdot [-n_e(\mu_e \bullet \mathbf{E}) - n_e(\mathbf{D}_e \bullet \nabla N_e) - n_e(\nabla \bullet \mathbf{D}_e)] = R_e$$

similarly, for the electron energy density:

$$n_\varepsilon\frac{\partial}{\partial t}(E_n) + \nabla \cdot [-n_\varepsilon(\mu_\varepsilon \bullet \mathbf{E}) - n_\varepsilon(\mathbf{D}_\varepsilon \bullet \nabla E_n) - n_\varepsilon(\nabla \bullet \mathbf{D}_\varepsilon)] + \mathbf{E} \cdot \Gamma_e = S_{\mathrm{en}}$$

The resulting equation system is inherently more stable than the original equation system. A useful quantity for results and analysis is the electron “temperature” (SI unit: V) which is defined as:

$$T_e = \left(\frac{2}{3}\right)\bar{\varepsilon}$$

Source Coefficients

The source coefficients in the equations in the Electron Transport Theory section are determined by the plasma chemistry and are written using either rate or Townsend coefficients. Suppose that there are M reactions which contribute to the growth or decay of electron density and P inelastic electron-neutral collisions. In general, $P >> M$. In the case of *rate coefficients*, the electron source term is given by

$$R_e = \sum_{j=1}^{M} x_j k_j N_n n_e$$

where x_j is the mole fraction of the target species for reaction j, k_j is the rate coefficient for reaction j (SI unit: m^3/s) and N_n is the total neutral number density (SI unit: 1/m^3). When *Townsend coefficients* are used, the source term becomes

$$R_e = \sum_{j=1}^{M} x_j \alpha_j N_n |\Gamma_e|$$

where α_j is the Townsend coefficient for reaction j (SI unit: m^2) and Γ_e is the electron flux as defined above (SI unit: 1/(m^2·s)). Townsend coefficients can increase the stability of the numerical scheme when the electron flux is field driven as is the case with DC discharges.

The electron energy loss is obtained by summing the collisional energy loss over all reactions:

$$S_{\mathrm{en}} = \sum_{j=1}^{P} x_j k_j N_n n_e \Delta\varepsilon_j$$

where $\Delta\varepsilon_j$ is the energy loss from reaction j (SI unit: V). In the case of Townsend coefficients, the energy loss is given by

$$S_{\mathrm{en}} = \sum_{j=1}^{P} x_j \alpha_j N_n |\Gamma_e| \Delta\varepsilon_j$$

The energy loss due to *inelastic collisions* is the most important input parameter in the Drift Diffusion interface. If this is not correctly defined, severe numerical problems result.

The rate and Townsend coefficients k_j and α_j depend exponentially on the mean electron energy, $\bar{\varepsilon}$. When a Maxwellian EEDF is assumed, the rate and Townsend coefficients can be fitted with a function of the form:

$$k_j = A\bar{\varepsilon}^{\beta}e^{-E/\bar{\varepsilon}}$$

The physics interface does not provide a way of calculating the rate or Townsend coefficients. The multiphysics interfaces for the most common types of reactor automatically compute the rate coefficients for each electron impact reaction based on plasma chemistry.

Stabilization

When the source stabilization is on, additional source terms are added to Equation 4-5 and Equation 4-6. For the electron density, the following source is added:

$$R_e = N_A\exp(-\zeta\ln(n_e))$$

where ζ is the user-defined tuning parameter. When the electron density is very low, this acts as a source term that prevents the electron density from approaching zero. As the electron density increases this term becomes exponentially smaller, eventually becoming negligible for high electron densities. A similar source is added for the electron energy density:

$$S_{\mathrm{en}} = N_A\exp(-\zeta\ln(n_\varepsilon))$$

Tensor Transport Properties

If a strong DC magnetic field is present then the electron mobility can be a tensor:

$$\mu_e^{\mathbf{-1}} = \begin{bmatrix} \frac{1}{\mu_{dc}} & -B_z & B_y \\ B_z & \frac{1}{\mu_{dc}} & -B_x \\ -B_y & B_x & \frac{1}{\mu_{dc}} \end{bmatrix}$$

where the inverse of the mobility has been used because the actual expression for the electron mobility cannot be written in a compact form. The quantity μ_{dc} is the electron mobility in the absence of a magnetic field. The electron diffusivity, energy mobility, and energy diffusivity are then calculated using:

$$\mathbf{D}_e = \mu_e T_e, \mu_\varepsilon = \left(\frac{5}{3}\right)\mu_e, \mathbf{D}_\varepsilon = \mu_\varepsilon T_e$$

These equations are exact for a Maxwellian EEDF, a constant momentum transfer frequency and a constant kinetic pressure.

The Wall Boundary Condition

At the Wall, the exchange of electrons occurs through the following mechanisms:

- Loss of electrons due to a net flux of electrons from the *plasma bulk* to the wall.
- Loss of electrons due to the random motion of electrons within the mean free path of the wall.
- Gain of electrons due to secondary emission; a wall emits an electron with some probability when it is struck by an ion or other excited species.
- Gain of electrons due to *thermionic emission*.

The resulting equation for the normal component of the electron flux at the wall is given by:

$$\mathbf{n} \cdot \Gamma_e = \frac{1-r_e}{1+r_e}\left(\frac{1}{2}v_{e,\,\mathrm{th}} n_e\right) - \left[\sum_i \gamma_i (\Gamma_i \cdot \mathbf{n}) + \Gamma_t \cdot \mathbf{n}\right] \qquad (4\text{-}8)$$

and for the normal component of the electron energy density:

$$\mathbf{n} \cdot \Gamma_\varepsilon = \frac{1-r_e}{1+r_e}\left(\frac{5}{6}v_{e,\,\mathrm{th}} n_\varepsilon\right) - \left[\sum_i \gamma_i \bar{\varepsilon}_i (\Gamma_i \cdot \mathbf{n}) + \bar{\varepsilon}_t \Gamma_t \cdot \mathbf{n}\right] \qquad (4\text{-}9)$$

here:

- r_e is the reflection coefficient (usually 0).
- $v_{e,\mathrm{th}}$ is the thermal velocity (SI unit: m/s).
- γ_i is the secondary emission coefficient from the *i*th positive ion species.
- Γ_i is the ion flux of the *i*th positive ion species at the wall (SI unit: $1/(\mathrm{m}^2 \cdot \mathrm{s})$).
- Γ_t is the thermal emission flux (SI unit: $1/(\mathrm{m}^2 \cdot \mathrm{s})$).

- $\bar{\varepsilon}_i$ is the mean energy of the ith species of secondary electrons (SI unit: V).
- $\bar{\varepsilon}_t$ is the mean energy of thermally emitted electrons (SI unit: V).
- $\mathbf{n}$ is the outward normal.

The thermal velocity is defined as:

$$\nu_{e,\text{th}} = \sqrt{\frac{8k_bT_e}{\pi m_e}}.$$

When the **Include migration effects** is activated the first term on the right-hand side of Equation 4-8 becomes:

$$\frac{1-r_e}{1+r_e}\left(\frac{1}{2}\nu_{e,\text{th}}n_e + n_e(\mu_e\cdot\mathbf{E})\cdot\mathbf{n}\right)$$

and the first term on the right-hand side of Equation 4-9 becomes:

$$\frac{1-r_e}{1+r_e}\left(\frac{5}{6}\nu_{e,\text{th}}n_\varepsilon + n_\varepsilon(\mu_\varepsilon\cdot\mathbf{E})\cdot\mathbf{n}\right)$$

where $\mathbf{E}$ is the electric field on the wall.

The Insulation Boundary Condition

The Insulation boundary condition simply sets the normal component of the electron and electron energy flux to zero:

$$\mathbf{n}\cdot\Gamma_e = 0 \text{ and} \mathbf{n}\cdot\Gamma_\varepsilon = 0$$

The Electron Density and Energy Boundary Condition

The Electron Density and Energy boundary condition fixes the electron density to a specific number density or the mean electron energy to a specific energy. This boundary condition should, in general, be avoided in favor of the Wall boundary condition.

$$n_e = n_{ew} \text{ and} \bar{\varepsilon} = \bar{\varepsilon}_0$$

Due to the fact that the log of the electron density is solved for, specifying an electron density of zero is not allowed.

Reference for the Drift Diffusion Interface

1. G.J.M. Hagelaar and L. C. Pitchford, “Solving the Boltzmann equation to obtain electron transport coefficients and rate coefficients for fluid models”, *Plasma Sources Science and Technology*, vol. 14, pp. 722–733, (2005).

2. E. Gogolides and H.H. Sawin, “Continuum modeling of radio-frequency glow discharges. I. Theory and results for electropositive and electronegative gases”, *J. Appl. Phys.*, vol. 72, 3971, 1992.

5

The Heavy Species Transport Interface

This chapter describes the Heavy Species Transport interface, which is a mass balance interface for all nonelectron species. This physics interface also makes it possible to add electron impact reactions, chemical reactions, surface reactions, and species, including charged, neutral, and electronically excited species. The physics interface is found under the **Plasma>Species Transport** branch ( ).

In this chapter:

- The Heavy Species Transport Interface
- Theory for the Heavy Species Transport Interface

The Heavy Species Transport Interface

The **Heavy Species Transport (hs)** interface ( ), found under the **Plasma >Species Transport** branch ( ), adds electron impact reactions, gas phase reactions, and species and surface reactions to plasma models. The most convenient way to do this is to load a set of collision cross sections from a file in the **Cross Section Import** section from the **Settings** window for **Heavy Species Transport**.

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `hs`.

OUT-OF-PLANE THICKNESS (2D, 1D AXISYMMETRIC, AND 1D)

- For 2D components, enter a **Thickness** d (SI unit: m). The default is 1 m.
- For 1D axisymmetric components, enter a **Vertical height** d_z (SI unit: m). The default is 0.01 m.
- For 1D components, enter a **Cross-section area** A (SI unit: m^2). The default is 0.01 m^2.

DIFFUSION MODEL

Select a **Diffusion model** — **Mixture-averaged** (the default), **Fick's law**, or **Global**. When using the **Mixture-averaged** or **Global** models, the mixture averaged diffusion coefficients are automatically computed based on the data specified for each species.

TRANSPORT SETTINGS

The **Migration in electric field** check box is selected by default. The **Migration** term is part of the total mass flux vector. Select the check boxes for other transport mechanisms to **Include** — **Convection**, **Calculate thermodynamic properties**, **Full expression for diffusivity**, or **Compute tensor ion transport properties**. The selection changes the number of **Model Inputs** requiring values on the **Settings** window for **Diffusion, Migration**. Note the following:

- **Convection** of heavy species present in a plasma can often be neglected due to the low operating pressure.
- For **Calculate thermodynamic properties** select the thermodynamic properties of each reaction and species are computed automatically based on the thermodynamic properties of each species.
- For **Full expression for diffusivity** it computes a more accurate expression for the binary diffusion coefficients. Often the additional correction terms (the collision term, Ω_D given by Equation 5-19 and used in Equation 5-18) are negligible in which case the expressions are much simpler and the time taken to assemble the Jacobian matrix is reduced.
- For **Mixture diffusion correction** additional terms are included in the definition of the mass flux vector to ensure that the same solution is obtained regardless of the choice of the species which comes from the mass constraint. This option makes the problem more nonlinear and strongly coupled, and is only necessary when the molecular weights of the species differ substantially (such as a mixture of sulfur hexafluoride and hydrogen).
- For **Compute tensor ion transport properties** the tensor form of the ion transport properties when a static magnetic field is present are computed. This option only needs to be activated when a strong DC magnetic field exists and the operating pressure is very low (on the order of millitorr). When this option is activated an expression must be provided for the magnetic flux density which would typically be computed by another physics interface. This is set in the **Convection, Migration, Diffusion** feature.

When the **Diffusion model** is set to **Global** only the properties **Calculate thermodynamic properties** and **Full expression for diffusivity** are available.

REACTOR

This section is available when the **Diffusion model** is set to **Global**. Select a **Reactor type** from the list — **Closed reactor** (the default), **Constant mass**, or **Constant pressure**.

- **Closed reactor** solves a closed system where mass and pressure can change, for example, as the result of surface reactions and volume reactions of the associative/dissociative type.

- **Constant mass** solves a system with mass-flow feed and outlet. The mass-flow outlet is set to keep the mass-density constant.
- **Constant pressure** solves a system with mass-flow feed and outlet. The pressure is kept constant by adjusting the system mass-density if needed.

ELECTRON ENERGY DISTRIBUTION FUNCTION SETTINGS

If cross-section data is used to define source coefficients in the model then an electron energy distribution function (EEDF) needs to be selected. These options are available.

For more information on the electron energy distribution function (EEDF), see the Theory for the Boltzmann Equation, Two-Term Approximation Interface.

- **Maxwellian**. This option assumes a Maxwellian EEDF which takes the form

$$f(\varepsilon) = \varphi^{-3/2}\beta_1\exp(-(\varepsilon\beta_2/\varphi))$$

where

$$\beta_1 = \Gamma(5/2)^{3/2}\Gamma(3/2)^{-5/2}, \beta_2 = \Gamma(5/2)\Gamma(3/2)^{-1}$$

where φ is the mean electron energy (eV), ε is the electron energy (eV) and Γ is the incomplete gamma function.

- **Druyvesteyn**. This option assumes a Druyvesteyn EEDF which takes the form

$$f(\varepsilon) = 2\varphi^{-3/2}\beta_1\exp(-(\varepsilon\beta_2/\varphi)^2)$$

where

$$\beta_1 = \Gamma(5/4)^{3/2}\Gamma(3/4)^{-5/2}, \beta_2 = \Gamma(5/4)\Gamma(3/4)^{-1}$$

- For **Generalized**, it is a generalized distribution function where the EEDF is somewhere between Maxwellian and Druyvesteyn. Specify a power law; the number must be between 1 and 2. Mathematically, the EEDF takes the form

$$f(\varepsilon) = g\varphi^{-3/2}\beta_1\exp(-(\varepsilon\beta_2/\varphi)^g)$$

  where

$$\beta_1 = \Gamma(5/2g)^{3/2}\Gamma(3/2g)^{-5/2}, \beta_2 = \Gamma(5/2g)\Gamma(3/2g)^{-1}, 1 \le g \le 2$$

- **Function**. If a two-dimensional interpolation function has been added to the model, it can be used for the EEDF. In this case, the *x*-data should be the electron energy (eV) and the *y*-data should be the mean electron energy (eV).

The two-dimensional interpolation function can be computed using a parametric sweep in The Boltzmann Equation, Two-Term Approximation Interface. This allows for modeling of discharges where the EEDF is far from Maxwellian. For step-by-step instructions on how to do this, refer to this blog entry: https://www.comsol.com/blogs/the-boltzmann-equation-two-term-approximation-interface/

In all these cases the rate constants in the model are automatically computed based on the selected EEDF using the formula

$$k_{\mathrm{f}} = \gamma\int_0^\infty \varepsilon\sigma_k(\varepsilon)f(\varepsilon)d\varepsilon \tag{5-1}$$

The rate coefficients when computed using cross section data are a highly nonlinear function of the mean electron energy. COMSOL Multiphysics automatically computes the integral in Equation 5-1 and makes the result available for evaluation of the rate coefficient. The variation of the rate coefficient for any particular model can be plotted using `<name>.kf_<reaction_number>`. For example, for reaction number 3 in a Heavy Species interface, with name `hs`, the rate coefficient would be plotted using `hs.kf_3`.

STABILIZATION

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

If the **Formulation** is set to **Finite element, log formulation (linear shape function)** or **Finite element, log formulation (quadratic shape function)** then the solver can run into difficulties when the species mass fractions approach zero. The **Reaction source stabilization** check box (selected by default) adds an additional source term to the rate expression for each species. In the ι text field, enter a tuning parameter for the source stabilization. The default value is 1. This value is usually good enough. If the plasma is high pressure (atmospheric) then it can help to lower this number to somewhere in the range of 0.25–0.5.

SURFACE CHEMISTRY

Use this to specify the **Total surface site concentration** (SI unit: mol/m^2) which will be added on all boundaries. This setting only affects models which have surface species included in the surface chemical mechanism.

DISCRETIZATION

Select **Formulation** — **Finite volume (constant shape function)**, **Finite element, log formulation (linear shape function)** (the default) to solve the equations in logarithmic form, **Finite element (linear shape function)**, **Finite element, log formulation (quadratic shape function)**, or **Finite element (quadratic shape function)**. The option with log formulation solve for the log of the dependent variables, ensuring that the mass fraction of any of the species is never lower than zero. This makes it more numerically stable but increases the nonlinearity of the equation system, and as such the model might take slightly longer to solve. The linear formulation solves the equations in the original form.

When the **Diffusion model** is set to **Global** the sections **Stabilization** and **Discretization** are not available.

- Stabilization
- Domain, Boundary, and Pair Nodes for the Heavy Species Transport Interface
- Theory for the Heavy Species Transport Interface
- Interpolation in the *COMSOL Multiphysics Reference Manual*

Surface Chemistry Tutorial Using the Plasma Module: Application Library path **Plasma_Module/Chemical_Vapor_Deposition/surface_chemistry_tutorial**

Domain, Boundary, and Pair Nodes for the Heavy Species Transport Interface

The Heavy Species Transport Interface has these domain, boundary, and pair nodes available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or right-click to access the context menu (all users).

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

The boundary conditions are available for both charged and neutral species. Surface reactions can be added to the model — COMSOL Multiphysics then automatically generates the appropriate expressions for the fluxes for each species on the boundaries. Boundary conditions do not need to be applied for a species that comes from the mass constraint condition.

- Cross Section Import
- Diffusion and Migration
- Electron Impact Reaction
- Flux
- Inlet
- Outflow
- Reaction
- Reaction Group
- Species
- Surface Reaction
- Surface Species
- Symmetry

Due to the overwhelming number of species that can exist in a plasma, the **Flux**, **Inlet**, **Outflow**, and **Symmetry** boundary condition subnodes are available from the context menu (right-click the **Species** parent node) or from the **Physics** toolbar, **Attributes** menu. This simplifies and orders the process of assigning boundary conditions for a large number of species in a model.

In the *COMSOL Multiphysics Reference Manual* see Table 2-4 for links to common sections and Table 2-5 to common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

Cross Section Import

The **Cross Section Import** node allows you to import collision cross section data into the model. The cross section data must be in a specific format. Browse to a file containing a set of collision cross sections.

The file must follow the format documented in Importing Collision Cross-Section Data.

Diffusion and Migration

The **Diffusion and Migration** node sets up the model inputs for the physics interface. Choose how to define the density of the gas mixture.

The node name changes to **Diffusion**, **Diffusion and Migration**, **Convection and Diffusion**, or **Convection, Diffusion, and Migration** based on the check boxes selected in the **Transport Settings** section. If **Global** is selected in **Diffusion model** the node name changes to **Global Model**.

MODEL INPUTS

This section has fields and values that are inputs to expressions that define material properties. If such user-defined property groups are added, the model inputs display here. Initially, this section is empty. The input depends on which **Transport Settings** are selected on The Heavy Species Transport Interface.

- Enter a **Temperature** T (SI unit: K) for the gas mixture used to compute the gas density and species transport properties.
- Enter an **Absolute pressure** p_A (SI unit: Pa) to compute the gas density and the binary diffusion coefficients.

- When the default **Migration in electric field** check box is selected, enter an **Electric potential** V (SI unit: V). This is used to compute the electric field that results in migration of any charged species.
- If the **Convection** check box is selected, enter coordinates for the **Velocity field** μ (SI unit: m/s).
- If the **Compute tensor electron transport properties** check box is selected, enter coordinates for the **Magnetic flux density** $\mathbf{B}$ (SI unit: T). This usually comes from a magnetostatic model and is used to compute anisotropic transport properties for electrons.

If **Global** is selected in **Diffusion model** this section has the following fields:

- Enter the temperature of the background gas in **Temperature** T (SI unit: K).
- If the **Reactor type** is set to **Closed reactor** or **Constant mass** enter an **Initial pressure** p_0 (SI unit: Pa). If the **Reactor type** is set to **Constant pressure** enter an **Absolute pressure** p_A (SI unit: Pa).
- Enter an **Absolute electron Flux** $|\Gamma_e|$ (SI unit: $1/(\mathrm{m}^2{\cdot}\mathrm{s})$) if there are reaction rate constants defined using **Townsend coefficient**.

DENSITY

Select an option from the **Mixture density** list — **Ideal gas** (the default) or **User defined**. For **User defined** enter a value for ρ (SI unit: $\mathrm{kg/m}^3$).

For **Ideal gas** it computes the expression for the gas density automatically using the ideal gas law, based on the composition of the mixture. The density is a function of the mean molecular weight

$$\rho = \frac{p_A M_n}{RT}$$

where p_A is the absolute pressure, T is the temperature, R is the universal gas constant, and M_n is the mean molar mass given by

$$\frac{1}{M_n} = \sum_{k=1}^{Q} \frac{w_k}{M_k}$$

TOTAL MASS FLOW

If **Global** is selected in **Diffusion model** and the **Reactor type** is set to **Constant mass** or **Constant pressure** enter the **Number of SCCM units** for the reactor feed. The **Feed mole fraction** of individual species can be set in the **Species** node ( ).

MEAN ELECTRON ENERGY

If **Global** is selected in **Diffusion model** enter the **Mean electron energy** (unit: V) to compute the Bohm velocity:

$$V_{\mathrm{Bohm},\,k} = \sqrt{\frac{eN_aT_e}{M_k}}\ .$$

In the **Surface Reaction** node ( ) in the section **Reaction formula** the ion forward rate constant is set equal to the Bohm velocity if **Bohm velocity** is selected in **Specify reaction using**.

Electron Impact Reaction

The **Electron Impact Reaction** node ( ) sets up the reaction rate for a reaction involving an electron and a target species.

REACTION FORMULA

Enter chemical reaction formulas in the **Formula** text field. The physics interface examines the species taking part in the reaction and automatically adds the associated **Species** features to the **Model Builder**.

Electron impact reactions are always assumed to be irreversible. However, the reverse reaction is automatically created if it is specified that a detailed balance should be constructed in the cross section data file. See Importing Collision Cross-Section Data.

Entering a Formula

The letter “e” is reserved to represent an electron. Therefore, if a formula such as `e+Ar=>e+Ar+` is entered, the species `e` is automatically assigned as an electron.

Valid species names consist of a case-sensitive combination of letters, numbers, and "+" or "-" signs. With the exception of the "+" and "-" signs, which are used for ions, special characters are not allowed as species names or part of species names (for example, (), [], *, #, and _). In addition, the first character in a species name must be a letter to avoid confusion with the stoichiometric coefficients in the reaction formula. Ions are labeled with one plus or minus sign per species charge added directly after the species name. For example, O++ represents an oxygen atom with a twofold positive charge; O2+ represents molecular oxygen with a single positive charge; and 2O+ represents two oxygen atoms, each with a single positive charge. This labeling automatically fills in the charge of the species ion the **Charge** text field in the associated Species node.

COLLISION TYPE

This section is available when a valid formula for the electron impact reaction is entered.

Select a **Collision type** — **Elastic** (the default), **Excitation**, **Attachment**, or **Ionization**.

- For **Elastic** enter a value for the **Electron mass ratio** m_r (dimensionless) (the ratio of the mass of the electron to the target species).
- Select **Excitation** for excitation reactions where the collision is inelastic. Enter the **Energy loss** $\Delta\varepsilon$ (SI unit: V) for the reaction. This is also known as threshold energy for the reaction. Select this option for electronic excitation, vibrational and rotational excitation, and superelastic collisions. For superelastic collisions, the activation energy is negative.
- Select **Attachment** for collisions which result in the electron attaching to the target species, forming a negative ion. No user input is required.
- Select **Ionization** for electron impact reactions which result in secondary electrons being formed. Enter the **Energy loss** $\Delta\varepsilon$ (SI unit: V) for the reaction. This is also known as threshold energy for the reaction.

COLLISION

This section is available when a valid formula for the electron impact reaction is entered.

Specify how to specify the reaction rate constant. There are these options:

- For **Cross section data** enter cross-section data directly into a table for the electron impact reaction.
- For **Arrhenius parameters** use Arrhenius coefficients to specify the reactions dependence on the electron “temperature”, T_e.
- For **Rate constant** specify an arbitrary expression for the rate constant. This can be a numeric value or a complicated function of the electron “temperature”, gas temperature, or any other expression. This is the default option for specifying the reaction.
- For **Use lookup table** either enter or load in a lookup table of the source coefficient versus mean electron energy.

CROSS SECTION DATA

This section is available when a valid formula is entered for the electron impact reaction and the **Specify reaction using** option is set to **Cross section data** in the **Collision** section.

By default the electron energy distribution function is taken from the physics interface property. It can be changed to **Maxwellian**, **Druyvesteyn**, or **Generalized**. In this case, the selected distribution function is only used in the current electron impact reaction. The **Electron energy** in the **Collision cross section data** table should have units of electronvolts (eV). The units of the **Collision cross section data** has units of m^2.

If a set of collision cross sections are loaded from a file with the Cross Section Import node, the software automatically generates the electron impact collisions and enters the cross section data into the **Cross section data** table.

Electron Energy Distribution Function Settings

ARRHENIUS PARAMETERS

For **Arrhenius parameters** specify the reaction then COMSOL Multiphysics automatically generates an expression of the form

$$k_{\mathrm{f}} = A_{\mathrm{f}} T_e^{n_f} \exp(-E_{\mathrm{f}}/T_e)$$

for the rate constant. Here, A_{f} is the forward frequency factor, n_{f} is the electron temperature exponent, and E_{f} is the activation energy (eV). The quantity T_e is the electron “temperature” (eV), which is defined as

$$T_e = \frac{2}{3}\bar{\varepsilon}$$

where $\bar{\varepsilon}$ is the mean electron energy.

KINETIC EXPRESSIONS

If the **Specify reaction using** option is set to **Rate constant** then enter in an expression for the reaction forward rate constant in the **Forward rate constant** text field. The units of the rate constant depend on the order of the reaction.

TABLE 5-1: SI UNITS FOR THE RATE CONSTANT FOR DIFFERENT REACTION ORDERS.

ORDER	EXAMPLE	SI UNIT
1	O2s=>O2	1/s
2	O+O2=>O3	$m^3/(s\cdot mol)$
3	3O=>O2+O	$m^6/(s\cdot mol^2)$

SOURCE COEFFICIENT DATA

If the **Specify reaction using** option is set to **Use lookup table** then the **Source coefficient** data section appears. Choose between **Rate coefficients** and **Townsend coefficients** to specify the source coefficients. For **Rate coefficients** either enter in the lookup table or load in from file a table of mean electron energy (eV) versus rate coefficient (SI unit: m^3/s). For **Townsend coefficients** enter in or load data from file a table of mean electron energy (eV) versus Townsend coefficient (SI unit: m^2).

Reaction

Use the **Reaction** node (⚗) to enter reversible or irreversible reactions that do not involve electrons (for example, for gas phase reactions). Enter the **Formula** for the reaction into the field and then fill in the appropriate information about the reaction.

DOMAIN SELECTION

COMSOL Multiphysics automatically ensures that the **Reaction** selection is the same as the Diffusion and Migration or Plasma Model selection.

Reaction Formula

See the Reaction Formula settings for the Electron Impact Reaction node. Note that the **Reaction type** can be **Reversible** or **Irreversible** unlike the Electron Impact Reaction.

Reaction Parameters

This section is available when a valid formula for the reaction is entered.

Choose whether or not to use Arrhenius expressions to compute the reaction forward and reverse rate constants using:

$$k = AT^n \exp\left(-\frac{E}{R_g T}\right)$$

To activate the **Arrhenius Parameters** section and make the kinetic expressions use these parameters, select the **Use Arrhenius expressions** check box. In the **Arrhenius Parameters** section, A denotes the frequency factor, n the temperature exponent, E the activation energy (SI unit: J/mol), and R_g the gas constant, 8.314 J/(mol·K). The pre-exponential factor, including the frequency factor A and the temperature factor T^n, is given the units $(\mathrm{m}^3/\mathrm{mol})^{\alpha - 1}/\mathrm{s}$, where α is the order of the reaction (with respect to the concentrations). Text fields are available for each of the Arrhenius parameters in the equation above.

Modify the default expressions for the rate constant. This field is only available if the **Use Arrhenius expressions** check box is cleared. The units of the rate constant depend on the order of the reaction.

TABLE 5-2: SI UNITS FOR THE RATE CONSTANT FOR DIFFERENT REACTION ORDERS.

ORDER	EXAMPLE	SI UNIT
1	O2s=>O2	1/s
2	O+O2=>O3	$\mathrm{m}^3/(\mathrm{s}\cdot\mathrm{mol})$
3	3O=>O2+O	$\mathrm{m}^6/(\mathrm{s}\cdot\mathrm{mol}^2)$

If the **Reaction type** is **Reversible**, the **Reverse rate constant** can either be specified directly, or by using an equilibrium constant. When **Specify equilibrium constant** is not active, enter a value for the **Reverse rate constant**. The units depend on the reaction order. When **Specify equilibrium constant** is active, enter a value for the **Equilibrium constant**. When the **Calculate thermodynamic properties** property is active, the **Equilibrium constant** is computed based on the thermodynamic properties.

Reaction Group

The **Reaction Group** node ( ) allows input of a large number of reversible reactions in the same table. More than one Reaction Group can be added to a component.

DOMAIN SELECTION

COMSOL Multiphysics automatically ensures that the **Reaction Group** selection is the same as the Diffusion and Migration or Plasma Model selection.

IMPORT REACTION FROM FILE

This section allows to import reactions into the table in the **Reaction Table** section. The reaction data must be in a specific format, as exemplified below.

```
%
% Heavy species reactions
%
REACTIONGROUP
Hn2+H2=>H3++e  ! 1.0e-11[cm^3/s]*N_A_const 1 0
Hn3+H2=>H3++e  ! 1.0e-11[cm^3/s]*N_A_const 1 0
H2+H2+=>H3++H  ! 2.1e-9[cm^3/s]*N_A_const 0 0
H2+H2<=>2H+H2  ! 3.7e-16 0 -48300 8.33e-33 1 0
O2++O2-=>O2+O2 ! 2.e-13 -0.5 0
O2+O3=>O2+O2+O ! 7e-16 0.0 -600
END
```

The reactions must be contained within the `REACTIONGROUP` and `END` keywords. Each line defines a reaction, with the formula defined on the left of the exclamation mark (`!`); on the right, the Arrhenius coefficients are listed in the following sequence: **Forward frequency factor**, **Forward temperature exponent**, **Forward activation energy**, **Reverse frequency factor**, **Reverse temperature exponent**, and **Reverse activation energy**. If the formula is irreversible (defined with =>), only the forward coefficients are needed. If the formula is reversible (defined with <=>), both forward and reversible coefficients need to be provided. If a coefficient is not needed to define the reaction, it should be specified as `0`. Any line that starts with `%` is treated as a comment.

REACTION TABLE

To enter reversible or irreversible gas phase reactions (for example, for gas phase reactions). Click the **Add** button (+) and enter the **Left-hand side, reactant** and **Right-hand side, product** of the formula. Species will be automatically created and updated.

It is possible to add reactions that involve electrons and the source term for the electrons will contain this information. However, the energy lost or gained by the electron is not taken into consideration in the electron mean energy equation.

Species

Use the **Species** node ( ) to add additional species to the model even if they do not participate in any reactions.

Due to the overwhelming number of species that can exist in a plasma, the **Flux**, **Inlet**, **Outflow**, and **Symmetry** boundary condition subnodes are available from the context menu (right-click the **Species** parent node) or from the **Physics** toolbar, **Attributes** menu. This simplifies and orders the process of assigning boundary conditions for a large number of species in a model. The subnodes are available for the **Heavy Species Transport** and **Plasma** physics interfaces, but not for the **Plasma, Time Periodic** interface. The subnodes are also not available if the **Species type** is set to **Electron**.

Species are automatically added to the model when reactions are added.

DOMAIN SELECTION

COMSOL Multiphysics automatically ensures that the **Species** selection is the same as the Diffusion and Migration selection.

SPECIES FORMULA

Species Name

Enter basic information about the name and nature of the species. When a new species is created, the **Species name** field is editable. A valid name for the species must be entered, for example, o2, Ar, or Ar+. After the species name has been entered, the text field cannot be edited (as long as a valid species name is entered).

Species Type

The **Species type** is either **Neutral**, **Ion**, or **Electron**.

- Enter “e” to automatically create a species as an **Electron**. The **Species** node is renamed to include the name entered in the **Species type** field. Also enter the Electron Parameters.
- Enter a neutral species, for example, o2 to automatically create a species as **Neutral**.
- Enter a charged species, for example, Ar+ or o2- to automatically create a species as an **Ion**. The **Species** node is renamed to include the name entered in the **Species type** field. When the species created is an **Ion**, also select the **Initial value from electroneutrality constraint** check box to indicate that the initial mass fraction of this species is automatically computed based on the initial electron density specified in the electron species properties. This ensures that the solver can find consistent initial conditions when starting because the plasma is initially electrically neutral. Also enter the General Parameters described in the next section.
- If **Neutral** is selected as the **Species type**, enter the General Parameters described in the next section.

  For **Neutral** select the **From mass constraint** check box to indicate which species should be used as the mass constraint and therefore not solved for. This is typically the species with the highest mass fraction. It should not be a species whose mass fraction is a very small value (as is the case with ionic species).

GENERAL PARAMETERS

This section is available when the **Species type** under **Species Formula** selected is **Ion** or **Neutral**. Enter the physical properties of the species. Enter the following properties:

- **Preset species data**. The default value is **User defined** which means that the transport properties are computed from the values given in the text fields below. There is preset transport data available for the most common gases encountered in plasma processing. Selecting one of these gases fills in the correct values for the species transport properties automatically.
- **Molecular weight** M_w (SI unit: kg/mol) of the species. The default value for the molecular weight is 0.032. The molecular weight must be a nonzero number.
- **Charge** z (dimensionless) number of the species. The charge number is automatically computed from the formula specified via either a reaction or by manually creating a species.

- **Potential characteristic length** σ (SI unit: m). The potential characteristic length is used to compute the binary diffusion coefficients used in the Mixture-Averaged transport setting.
- **Potential energy minimum** e/k_b (SI unit: m). This is used to compute the binary diffusion coefficients.
- **Dipole moment** μ_D (SI unit: C·m). If the dipole moment is known, enter its value and obtain a more accurate expression for the binary diffusion coefficients.
- **Thermal diffusion coefficient** D^T (SI unit: m^2/s). If thermophoretic effects are important in the model then enter a value for the thermal diffusion coefficient.
- If **Neutral** is selected as the **Species type**, enter an **Initial mole fraction** x_0 (dimensionless). Set the initial mole fraction of the selected species in the gas mixture. If the **From mass constraint** check box is selected, this field is not available.
- If **Ion** is selected as the **Species type**, enter an **Initial number density** n_0 (dimensionless) to set the initial number density of the selected species in the gas mixture. If the **Initial condition from electroneutrality constraint** check box is selected, this field is not available.
- **Diffusion coefficient** D_f (SI unit: m^2/s). If the **Diffusion model** property is set to **Fick's law** then the diffusion coefficient is not computed automatically from the transport properties, its value is taken from this text field.
- If **Global** is selected in **Diffusion model**, if the **Reactor type** is set to **Constant pressure** or **Constant mass**, and if **Ion** is not selected as the **Species type** enter an **Feed mole fraction** x_feed (dimensionless) to set the mole fraction of the selected species in the feed.

ELECTRON PARAMETERS

This section is available when the **Species type** under **Species Formula** selected is **Electron** and the physics interface is **Heavy Species Transport**. These options are not available for the **Plasma**, or **Plasma, Time Periodic** interfaces. Enter the following properties:

- **Electron density** n_e (SI unit: $1/\mathrm{m}^3$).
- **Mean electron energy** e (SI unit: V).
- **Initial electron density** $n_{\mathrm{e},0}$ (SI unit: $1/\mathrm{m}^3$).

SPECIES THERMODYNAMIC PARAMETERS

This section is available if the **Calculate thermodynamic properties** check box is selected under Transport Settings in the Heavy Species Transport interface.

Enter thermodynamic properties for the species using the NASA polynomial format. Any coefficients for the thermodynamic polynomials are entered into the $a_{\mathrm{low,k}}$ fields and apply to the temperatures in the range **T$_{\mathrm{int}}$/lower** to **T$_{\mathrm{int}}$/midpoint**; coefficients entered into the $a_{\mathrm{hi,k}}$ fields apply to temperatures in the range **T$_{\mathrm{int}}$/midpoint** to **T$_{\mathrm{int}}$/upper** range.

- Enter the **Lower temperature limit** T_{lo}, **Middle temperature limit** T_{mid}, and **Upper temperature limit** T_{hi}. The defaults are 300 K, 1000 K, and 5000 K, respectively.
- Enter the **Polynomial coefficients** $a_{\mathrm{low,k}}$ and $a_{\mathrm{hi,k}}$.
- Enter an **Additional enthalpy contribution** Δh (SI unit: V) as required.

MOBILITY AND DIFFUSIVITY EXPRESSIONS

This section is available when the **Species type** selected under **Species Formula** is **Ion**. The default specification is to compute the diffusivity and mobility from the binary diffusion coefficients and Einstein's relation.

Select a **Specification** — **Compute mobility and diffusivity** or **Specify mobility, compute diffusivity**.

The **Mobility** is computed using Einstein's relation by default. If **Mixture averaged** is selected as the **Diffusion model** under Transport Settings in the Heavy Species Transport interface, then the mobility is computed using:

$$\mu = \frac{D_m q}{k_B T}$$

and if the Fick's law property is selected, then the mobility is defined as:

$$\mu = \frac{D_f q}{k_B T}$$

Select an **Ion temperature** — **Use gas temperature** or **Specify ion temperature**. For **Specify ion temperature** enter an **Ion temperature** T_{ion} (SI unit: K). The default is 300 K.

MOBILITY SPECIFICATION

This section is available if **Specify mobility, compute diffusivity** is selected under **Mobility and Diffusivity Expressions**.

Select an option from the **Specify using** list — **Constant value**, **Dalgarno**, **High field**, or **Lookup table**.

- For **Constant value** enter a value or expression for the **Mobility** u_m (SI unit: $m^2/(V{\cdot}s)$). This can be a function of any of the variables in the model.
- For **Dalgarno** enter a value or expression for the **Polarizability** α (SI unit: m^3). The default values is 1.64 $Å^3$.
- For **High Field** enter a value or expression for the **Cross section** σ (SI unit: m^2). The default value is $7 \cdot 10^{-19}$ m^2.
- For **Lookup table** the mobility is then a function of the electric field. Select an option from the **Mobility a function of** list — **Electric field** or **Reduced electric field**. Enter values into the table or click the **Load from file** button (📂) to import data from a text file. The file must contain the two columns of data, the **Electric field** (SI unit: V/m) or the **Reduced electric field** (SI unit: $V{\cdot}m^2$) as the x-data and the **Ion mobility** (SI unit: $m^2/(V{\cdot}s)$) or **Reduced ion mobility** (SI unit: $1/(V{\cdot}m{\cdot}s)$) as the y-data.
- For **Argon ion in argon**, the mobility automatically is defined as a function of the reduced electric field for an argon ion in a background gas of neutral argon. The data comes from Ref. 10 with data at higher reduced electric fields from Ref. 13.
- For **Helium ion in helium**, the mobility automatically is defined as a function of the reduced electric field for a helium ion in a background gas of neutral helium. The data comes from Ref. 10.

The **Diffusivity** is computed using Einstein's relation by default.

In **Ion temperature** a third option is available when **Specify mobility, compute diffusivity** is selected under **Mobility and Diffusivity Expressions**: **Use local field approximation**. Select **Use local field approximation** to automatically compute the ion temperature using the local field approximation. The local field approximation can be useful for high pressure discharges where the electric field is very high, as is the case for dielectric barrier discharges.

Surface Reaction

The **Surface Reaction** node (🧪) sets up reactions involving gas phase, surface, and bulk species on surfaces. Surface reactions play an important role in the overall characteristics of a discharge. This typically involves first-order reactions on surfaces where excited or ionic species transition back to their ground state.

REACTION FORMULA

The first thing to input after creating a new reaction is the **Reaction Formula**. The reaction can only be irreversible. Valid species names consist of a case-sensitive combination of letters, numbers, and “+” or “-” signs. With the exception of the “+” and “-” signs, which are used for ions, special characters are not allowed as species names or part of species names (for example, [], *, #, and _). In addition, the first character in a species name must be a letter to avoid confusion with the stoichiometric coefficients in the reaction formula. Ions are labeled with one plus or minus sign per species charge added directly after the species name.

For surface reactions, use the notation (s) to denote surface species or (b) to denote bulk species. (Bulk surface species is not to be confused with gas phase species.) The species is then called *`name`*`_surf` or *`name`*`_bulk` depending on the species type. For example, using `Si(s)` in the reaction formula creates a surface species with the name `Si_surf` and `Si(b)` creates a surface species with the name `Si_bulk`.

Under **Specify reaction using** select whether to specify the surface reaction using **Sticking coefficient** or **Rate coefficient**. If the **Diffusion model** property is set to **Global**, and the reaction is first order, the surface reaction can also be specified using **Sticking coefficient and diffusion**. If the reactant species has the **Species type** set to **Ion**, the surface reaction can also be specified using **Bohm velocity**. Sticking coefficients are typically easy to estimate intuitively and must be between 0 and 1. Rate coefficients require specification of a rate constant, which can be obtained in the appropriate literature.

Under **Motz-Wise correction** select whether to include this correction in the expression for the rate constant. This option is only available when the reaction is specified using sticking coefficients. When the **Motz-Wise correction** is set to **On**, Equation 5-21 is used to compute the rate constant. When set to **Off**, Equation 5-22 is used.

Only irreversible surface reactions are supported.

Arrhenius Parameters

This section is available when a valid formula for the surface reaction is entered.

Choose whether or not to use Arrhenius expressions to compute the sticking coefficients. When this option is activated, the expression for the sticking coefficient is automatically computed using:

$$\gamma_f = A_s^f T^{n_s^f} \exp\left(-\frac{E_s^f}{R_g T}\right)$$

where A_s^f is the frequency factor, n_s^f is the temperature exponent, and E_s^f is the activation energy (SI unit: J/mol).

REACTION PARAMETERS

The options available in this section depends on the **Diffusion model** property. If it is set to **Fick's law** or **Mixture Averaged**, the following options are available.

When **Specify reaction using** is set to **Sticking coefficient** and the **Use Arrhenius expressions** check box is not selected, enter a **Forward sticking coefficient** for the surface reaction. This value should be between 0 and 1. When the value is zero, the surface reaction rate is zero, and the surface reaction feature is not necessary. For neutral radicals, a typical value is 1E-3. For highly reactive species, like ions or electronically excited atoms or molecules, a value of 1 is more appropriate. If the **Use Arrhenius expressions** check box is selected, enter the following:

- A value or expression for the **Forward sticking coefficient frequency factor** $A^f{}_s$ (SI unit: 1). This should typically be a number between 0 and 1, since it is used to compute a sticking coefficient.
- A value or expression for the **Forward sticking coefficient exponent** $n^f{}_s$ (SI unit: 1). A nonzero value is only necessary if the sticking coefficient varies with temperature.
- A value or expression for the **Forward sticking coefficient activation energy** $E^f{}_s$ (SI unit: J/mol). A nonzero value is only necessary if the sticking coefficient varies with temperature.

When **Specify reaction using** is set to **Rate coefficient** and the **Use Arrhenius expressions** check box is not selected, enter a **Forward rate constant** for the surface reaction. The unit changes depending on the order of the reaction. The **Forward rate constant** could be dramatically different depending on the surface reaction. Entering unphysical values can lead to numerical instabilities when solving, so only well known values from the literature should be used. If the **Use Arrhenius expressions** check box is selected, enter the following:

- A value or expression for the **Forward surface frequency factor** a^f (SI unit: m/s for first order reactions). This unit will change depending on reaction order.

- A value or expression for the **Forward surface exponent** b^f (SI unit: 1). A nonzero value is only necessary if the rate constant varies with temperature.
- A value or expression for the **Forward surface activation energy** e^f (SI unit: J/mol). A nonzero value is only necessary if the rate constant varies with temperature.

If the **Diffusion model** property is set to **Global**, the following options are available in addition to the ones listed above.

- A value or expression for the **Correction factor** h_l (SI unit: 1). A value other than one only needs to be specified when the reactant is an ion. In that case, typical values range from 0.1–0.3. This can be thought of as a correction to the surface area where the reaction occurs. Values for specific reactions can be found in the literature.
- When **Specify reaction using** is set to **Sticking coefficient and diffusion**, enter an **Effective diffusion length** Λ_{eff} (SI unit: m). A smaller value will increase the surface reaction rate.

SECONDARY EMISSION PARAMETERS

If the **Heavy Species Transport** interface is coupled to the **Drift Diffusion** interface, or the **Plasma** interface is being used, it may be necessary to include secondary electron emission in the model. This is typically only needed when ions are the reactant.

- Enter a value for the **Secondary emission coefficient** γ_i (SI unit: 1). The value should be between 0 (no secondary emission) and around 0.45, for ions and surface which produce a high yield of secondary electrons. Values for specific pairs of ions and surfaces are given in the literature.
- Enter a value for the **Mean energy of secondary electron** ε_i (SI unit: V). This value may be hard to find in the literature, so what is often used is the value for the ionization energy of the reactant minus twice the work function of the surface. So, for an Argon ion, with ionization energy of 15.8 V and a surface with a work function of 5 V, the mean energy for the secondary electron would be 5.8 V.

Surface Species

The **Surface Species** () node adds detailed surface chemistry to the model. The species can either be surface species (for example, Si(s)) or bulk species (for example, Si(b)). Surface species are not available directly from the context menu, they can only added via a surface reaction. By entering a surface reaction formula containing surface species, the surface species will appear in the Model Builder. The selection for the surface species is locked, and automatically derived from any participating surface reactions. In addition, when all participating surface reactions are deleted in the Model Builder, the corresponding surface species will also be deleted.

Surfaces or bulk species are not supported for stationary studies. If a model contains a surface or a bulk species and an attempt is made to use a stationary study an error message appears. Surfaces or bulk species will have to be disabled to be able to use a stationary study.

SPECIES FORMULA

Enter a formula for the surface species. The symbols (s) and (b) are reserved to denote surface species or bulk species, respectively. So, to create a silicon surface species simply type `Si(s)`. To create a silicon bulk species type `Si(b)`. COMSOL Multiphysics automatically renames the species to a valid name, for example `Si_surf` or `Si_bulk`.

GENERAL PARAMETERS

Enter the molecular weight (SI unit: kg/mol) of the surface or bulk species. The default value is **User defined**, which means that the surface species properties are computed from the values given in the text fields below. There is preset transport data available for some of the typical surface species encountered in plasma processing. Selecting one of these fills in the correct values for the molecular weight and thermodynamic properties automatically.

SURFACE SPECIES PARAMETERS

If the species is a surface species then this section is available.

- Enter the **Site occupancy number** σ (dimensionless) of the species. The default value is 1. This accounts for the fact that larger molecules can occupy more than one active site on a surface.
- The **Initial site fraction** $Z_{k,0}$ (dimensionless) is the initial fraction of the total sites which is occupied by the species. This must be a number between 0 and 1.

BULK SPECIES PARAMETERS

If the species is a bulk species then enter the density (SI unit: kg/m^3). The default value is 2329 kg/m^3 corresponding to the density of silicon.

Flux

Due to the overwhelming number of species which can exist in a plasma, the **Flux** boundary condition subnode is available from the context menu (right-click the Species parent node) or from the **Physics** toolbar, **Attributes** menu. This simplifies and orders the process of assigning boundary conditions for a large number of species in a model.

The **Flux** subnode prescribes the mass flux of the selected species on selected boundaries. A user-defined normal flux is available via the equation:

$$-\mathbf{n}\cdot\rho w_k(\mathbf{V}_k+\mathbf{u}) = \Gamma_k$$

Inlet

The **Inlet** boundary condition subnode is available from the context menu (right-click the Species parent node) or from the **Physics** toolbar, **Attributes** menu. This simplifies and orders the process of assigning boundary conditions for a large number of species in a model.

The **Inlet** subnode constrains one of the following quantities:

- The mass fraction: $\omega = \omega_0$.
- The mole fraction: $x = x_0$.
- The molar concentration (SI unit: mol/m^3) $c = c_0$.
- The number density, which describes the number of particles per volume (SI unit: 1/m^3): $n = n_0$.

Constraints of a quantity other than the mass fraction can only be used when all species are defined, as in the Inlet condition. The other quantities are composition dependent and therefore unambiguous only when all species are defined.

INLET

Select the type of input from the **Mixture specification** list:

- Select **Mass fraction** (the default) to enter mass fractions ($\omega_{0,\omega1}$, for example).
- Select **Mole fraction** to enter mole fractions ($x_{0,\omega1}$, for example).
- Select **Molar concentration** to enter molar concentrations ($c_{0,\omega1}$, for example).
- Select **Number density** to enter number densities ($n_{0,\omega1}$, for example).

Type the value or expression in the text fields for each species except the one computed from the mass constraint.

When using the Logarithmic form of the equations, setting a zero value for any of the options listed above results in an error when attempting to solve. To avoid the solver taking `log(0)` a positive, nonzero value must be entered.

Outflow

The **Outflow** boundary condition subnode is available from the context menu (right-click the Species parent node) or from the **Physics** toolbar, **Attributes** menu.

The **Outflow** subnode is used on boundaries where the species mass is to be transported out of the model domain.

If an imaginary slice is drawn through the cross section of a geometry, then there is no net flux across the slice due to diffusion, since diffusion corresponds to random motion of the atoms and molecules. There can, however, be a net flux due to convection, migration, or thermophoresis, but outlet boundary conditions for temperature and electric field also impose zero gradient. An appropriate “open” boundary condition is to set the normal component of the diffusive flux to zero, which implies that diffusive transport of flux across the imaginary slice is zero. The boundary condition becomes:

$$-\mathbf{n}\cdot\mathbf{j}_k = 0$$

which assumes that all mass passing through the boundary is due to convection.

Symmetry

Due to the overwhelming number of species that can exist in a plasma, the **Symmetry** boundary condition subnode is available from the context menu (right-click the Species parent node) or from the **Physics** toolbar, **Attributes** menu. This simplifies and orders the process of assigning boundary conditions for a large number of species in a model.

The Symmetry subnode is used on all boundaries where the selected species mass fraction is symmetric. This is identical to the No Flux boundary condition. The symmetry boundary imposes a zero flux boundary condition.

Theory for the Heavy Species Transport Interface

The Heavy Species Transport Interface theory is described in this section:

About the Transport Equations

There are a number of different formulations available for modeling mass transport, ranging from a simple reaction, convection, diffusion equation to the Maxwell–Stefan (sometimes called Stefan-Maxwell) equations. The Maxwell–Stefan equations are the only formulation that conserves the total mass in the system and also satisfies a number of auxiliary constraints. The main drawback of the Maxwell–Stefan equations is the amount of computational resources needed to solve them. In many situations, simplified forms of the Maxwell Stefan equations can be applied, which do not rigorously satisfy the laws of multicomponent diffusion but are much easier to implement and consume less computational resources.

Neutral and excited species in a plasma require a suitable transport equation. Typically the concentration of excited species is very low and they can be considered dilute. This is true with atomic gases, where the concentration of ions and electronically excited species is much lower than the concentration of the ground state atoms. Molecular gases can dissociate into stable neutral fragments via *electron impact dissociation*. These fragments can have a long lifetime and accumulate inside a reactor. In this case, the dilute approximation is no longer valid and a more rigorous description of the mass transport is necessary. Conservation of mass dictates that the sum of the mass fractions of all the species must equal one. The most accurate technique is to solve the Maxwell–Stefan equations which correctly take into account diffusive transport due to mole fraction, pressure, and temperature gradients. The Maxwell–Stefan equations quickly become computationally very expensive when the number of species becomes large (>6). In a plasma, there might be 20+ neutral and excited species, so solving the full Maxwell–Stefan equations is not practical.

The Heavy Species Transport interface provides two different Diffusion models to choose from, Mixture averaged and Fick's law. The Mixture averaged option fulfills all the criteria for mass conservation. It is much less computationally expensive to solve than the full Maxwell–Stefan equations, which are not available in the Heavy Species Transport interface. The Fick's law option uses an even simpler diffusion model, which is less accurate than the Mixture averaged model, but is computationally less expensive.

Multicomponent Diffusion Equations

Suppose a reacting flow consists of $k = 1, \ldots, Q$ species and $j = 1, \ldots, N$ reactions. The equation for the first $Q - 1$ species is given by

$$\rho\frac{\partial}{\partial t}(w_k) + \rho(\mathbf{u}\cdot\nabla)w_k = \nabla\cdot\mathbf{j}_k + R_k \tag{5-2}$$

where:

- $\mathbf{j}_k$ is the diffusive flux vector
- R_k is the rate expression for species k (SI unit: kg/(m^3·s))
- $\mathbf{u}$ is the mass averaged fluid velocity vector (SI unit: m/s)
- ρ denotes the density of the mixture (SI unit: kg/m^3), and
- w_k is the mass fraction of the kth species (1).

The diffusive flux vector is defined as:

$$\mathbf{j}_k = \rho w_k \mathbf{V}_k$$

where and $\mathbf{V}_k$ is the multicomponent diffusion velocity for species k. The definition of $\mathbf{V}_k$ depends on the option chosen for the Diffusion Model property. If Mixture Averaged is chosen then:

$$\mathbf{V}_k = D_{k,m}\frac{\nabla w_k}{w_k} + D_{k,m}\frac{\nabla M_n}{M_n} + D_k^T\frac{\nabla T}{T} - z_k\mu_{k,m}\mathbf{E} \tag{5-3}$$

where

- $D_{k,m}$ is the mixture averaged diffusion coefficient (SI unit: m^2/s)
- M_n is the mean molar mass of the mixture (SI unit: kg/mol)
- T is the gas temperature (SI unit: K)
- D_k^T is the thermal diffusion coefficient for species k (SI unit: kg/(m·s))
- z_k is the charge number for species k (SI unit: dimensionless)
- $\mu_{k,m}$ is the mixture averaged mobility for species k (SI unit: m^2/(V·s))
- $\mathbf{E}$ is the electric field (SI unit: V/m)

The mixture averaged diffusion coefficient, $D_{k,m}$ for species k is defined as:

$$D_{k,m} = \frac{1 - w_k}{\sum\limits_{j \neq k}^{Q} x_j / D_{kj}} \tag{5-4}$$

The mean molar mass, M_n is computed using the sum of the mass fractions and molecular weights of all species:

$$\frac{1}{M_n} = \sum_{k=1}^{Q}\frac{w_k}{M_k}.$$

The mole fraction for species k, which is required in Equation 5-4 can be computed from the mass fraction of species k and the mean molar mass:

$$x_k = \frac{w_k}{M_k}M_n.$$

The quantity, D_{kj} is the binary diffusion coefficient between species k and j and are rather complicated functions of the physical properties of the species. They are defined later, in Equation 5-14.

By default, the mixture averaged mobility, $\mu_{k,m}$ is computed using *Einstein's relation*:

$$\mu_{k,m} = \frac{qD_{k,m}}{k_BT}$$

where q is the unit charge (SI unit: s A) and k_B is Boltzmann's constant (SI unit: J/K).

If the Mixture diffusion correction is checked, an additional term is added to the right hand side of Equation 5-3

$$\mathbf{V}_k = D_{k,m}\frac{\nabla w_k}{w_k} + D_{k,m}\frac{\nabla M_n}{M_n} + D_k{}^T\frac{\nabla T}{T} - z_k\mu_{k,m}\mathbf{E} + \sum_j \frac{M_j}{M_n} D_{j,m}\nabla x_j$$

This option makes the problem more nonlinear and strongly coupled, and is only necessary when the molecular weights of the species differ substantially (such as a mixture of sulfur hexafluoride and hydrogen).

If the Diffusion Model is changed to Fick's law then the diffusion coefficient must be given as an input for each species. In this case, the diffusion velocity is defined as:

$$\mathbf{V}_k = D_{k,f}\frac{\nabla w_k}{w_k} + D_{k,f}\frac{\nabla M_n}{M_n} + D_k{}^T\frac{\nabla T}{T} - z_k\mu_{k,f}\mathbf{E}$$

where $D_{k,f}$ is the user-defined diffusion coefficient for species k. The mobility for species k is then given by:

$$\mu_{k,f} = \frac{qD_{k,f}}{k_B T}.$$

This approach is much less computationally expensive than the Mixture averaged model since the diffusion coefficients are not computed from the complicated expression in Equation 5-4.

For ions, the mobility can be specified in Mobility Specification through an arbitrary expression, a lookup table as a function of the electric field, or through the Dalgarno or High field mobility models.

When Dalgarno Ref. 8 is selected in the Mobility Specification section the ion mobility is computed as:

$$\mu_{m,f} = \frac{0.0438}{\sqrt{\alpha m_r}}\frac{N_0}{N_n}$$

where α is the background atom polarizability in Å^3, $N_0 = 2.69 \cdot 10^{25}$ m^{-3} is the gas number density at 273.15°C and 760 Torr, and m_r is the reduced mass

$$m_r = \frac{M_n M_k}{M_n + M_k}.$$

Dalgarno's mobility corresponds to the polarizability limit of the classical Langevin theory Ref. 9. This mobility model gives reasonable values for alkali ions in Ar, Kr, Xe, N_2, and H_2. This model should not be used in the case resonant charge exchange contributes significantly to the ion transport as it occurs in ions drifting in the parent gas. Moreover, Dalgarno's mobility is only valid for small electric field strengths that ensure that the ion thermal velocity is much larger than the ion drift velocity. Note that the ion mobility and ion drift velocity are, respectively, independent and directly proportional to the electric field.

When High field is selected in the Mobility Specification section the ion mobility is computed as:

$$\mu_{m,f} = \sqrt{\frac{2q|z|}{\pi\frac{M_k}{N_A}N_n\sigma|\mathbf{E}|}}$$

where σ is a cross section characterizing the ion-atom interaction (SI unit: m^2). This model is valid when the electric field is high enough to ensures that the ion drift velocity is much larger than the ion thermal velocity Ref. 8. With this model the ion drift velocity is proportional to $\sqrt{|\mathbf{E}|/N_n}$ as is frequently found in experiments in the high field limit.

By default, the mixture averaged diffusivity $D_{k,m}$ (and Fick's law diffusivity $D_{k,f}$) are computed using *Einstein's relation*:

$$D_{k,m} = \frac{k_B T}{q}\mu_{k,m}.$$

There is also the option to use the *Local Field Approximation*. The local field approximation can be used to compute the ion temperature and thus the diffusivity through Einstein's relation, see Ref. 7 and Ref. 10. The ion temperature is computed using:

$$T_i = T + \left(\frac{M_k + M_n}{5M_k + 3M_n}\right)\frac{M_n}{k_b}(\mu\cdot\mathbf{E})\cdot(\mu\cdot\mathbf{E})$$

where M_k is the ion mass (SI unit: kg), T is the background gas temperature (SI unit: K), M_n is the mean mass of the mixture (SI unit: kg), and $\mathbf{E}$ is the electric field (SI unit: V/m). If the High field mobility model is used the background gas temperature is neglected and the ion temperature is computed as

$$T_i = \left(\frac{M_k + M_n}{5M_k + 3M_n}\right)\frac{M_n}{k_b}(\mu \cdot \mathbf{E}) \cdot (\mu \cdot \mathbf{E}) .$$

Additional Quantities of Interest

The Heavy Species Transport interface also defines a number of additional variables for quantities which are typically of interest during results processing. The number density of species k is given by

$$n_k = \left(\frac{p}{k_B T}\right)x_k$$

where p (SI unit: Pa) is the absolute pressure of the fluid. The molar concentration of species k is given by

$$c_k = \frac{w_k \rho}{M_k};$$

The density of the mixture is given by the ideal gas law by default:

$$\rho = \frac{pM_n}{RT}$$

where R is the universal gas constant (SI unit: J/(mol·K)).

A useful quantity in systems where large thermal gradients exist (that is, chemical vapor deposition processes) is the *thermal drift velocity* (SI unit: m/s), which is defined as

$$\mathbf{v}^T = D_k{}^T \nabla \ln T$$

Auxiliary Relations

Only $Q - 1$ of the Maxwell–Stefan equations are independent and so a *closure law* is required. One way of ensuring a unique solution is the use the fact that the sum of the diffusive fluxes must be zero:

$$\sum_{k=1}^{Q} \rho w_k \mathbf{V}_k = 0 \tag{5-5}$$

An alternative is to use the fact that the sum of the mass fractions must be equal to one:

$$w_Q = 1 - \sum_{k=1}^{Q-1} w_k \tag{5-6}$$

In the Heavy Species Transport interface, Equation 5-6 is used to fulfill the mass constraint so only $Q - 1$ species are solved for.

Source Coefficients

The rate expression, R_k is determined by the stoichiometry of the system through the formula

$$R_k = M_k \sum_{j=1}^{N} \upsilon_{kj} r_j \tag{5-7}$$

where ν_{kj} is the stoichiometric matrix and r_j is the reaction rate corresponding to reaction j. The reaction rates depend on the order of the reaction and the molar concentration of each species. In compact form, the equation for the jth reaction rate is

$$r_j = k_{\mathrm{f},j} \prod_{k=1}^{Q} c_k^{\nu'_{kj}} - k_{r,j} \prod_{k=1}^{Q} c_k^{\nu''_{kj}}$$

where:

- c_k is the molar concentration of species k (SI unit: mol/m^3)
- $k_{\mathrm{f},j}$ is the rate coefficient for the jth forward reaction (SI unit: 1/s for first order reactions, m^3/(mol·s) for second order reactions or m^6/(mol^2·s) for third order reactions)
- ν'_{kj} is the stoichiometric matrix corresponding to forward reactions, and
- ν''_{kj} is the stoichiometric matrix corresponding to reverse reactions.

The molar concentration of the kth species is computed via:

$$c_k = \frac{w_k \rho}{M_k}.$$

The rate coefficients k_j depend exponentially on the gas temperature through Arrhenius law

$$k_j = A_j T^{\beta_j} \exp\left(-\frac{E_j}{RT}\right)$$

where A_j is the frequency factor for reaction j, β is the temperature exponent and E_j is the activation energy for reaction j.

Thermodynamic Properties

The species' heat capacity, C_p, the molar enthalpy, h, and the molar entropy, s, are computed using the polynomial format of Gordon and McBride (Ref. 1). This manual also refers to these expressions as *NASA polynomials*:

$$C_{p,i} = R_g(a_1 + a_2 T + a_3 T^2 + a_4 T^3 + a_5 T^4) \tag{5-8}$$

$$h_i = R_g\left(a_1 T + \frac{a_2}{2}T^2 + \frac{a_3}{3}T^3 + \frac{a_4}{4}T^4 + \frac{a_5}{5}T^5 + a_6\right) + F\Delta h \tag{5-9}$$

$$s_i = R_g\left(a_1 \ln T + a_2 T + \frac{a_3}{2}T^2 + \frac{a_4}{3}T^3 + \frac{a_5}{4}T^4 + a_7\right) \tag{5-10}$$

Here, $C_{p,i}$ denotes the species' heat capacity (SI unit: J/(mol·K)), T the temperature (SI unit: K), and R_g the ideal gas constant, 8.314 J/(mol·K). Further, h_i is the species' molar enthalpy (SI unit: J/mol), Δh is an additional enthalpy contribution (SI unit: V), F is Faraday's constant (SI unit: C/mol), and s_i represents its molar entropy (SI unit: J/(mol·K)), at standard state. The additional enthalpy should include the internal energy of the species, if it is not already included, so that the enthalpy of the reaction can account for it. This is important when computing the heat source related with electron impact reactions.

The thermodynamic properties for a given reaction are computed from the enthalpy and entropy of the participating species. The enthalpy of reaction (SI unit: J/mol) is calculated using:

$$H_j = \sum_i \nu_{ij} h_i \tag{5-11}$$

Similarly, the entropy of reaction (SI unit: J/(mol·K)) comes from the relationship

$$S_j = \sum_i \nu_{ij} s_i \tag{5-12}$$

In these equations, h_i and s_i are the species' molar enthalpy (SI unit: J/mol) and entropy (SI unit: J/(mol·K)), respectively. The heat source of reaction (SI unit: J/(m^3·s)) is given as:

$$Q_j = -H_j r_j$$

In case of inelastic electron impact reactions, the energy loss by the electron in the reaction (**Energy loss** $\Delta\varepsilon$ set in the **Collision Type** section of the **Electron Impact Reaction** feature) is taken into account in Q_j.

For elastic electron impact reactions the heat source term is given by

$$Q_{ele,j} = \left(2\frac{m_e}{m_k}\right)\frac{3}{2}\left(T_e - \frac{k_B T_{gas}}{e}\right)Fr_j \tag{5-13}$$

where m_e and m_k are the electron and heavy species mass in kg, T_e is the electron temperature in eV, and T_{gas} is the gas temperature in K. The sum of Q_j and $Q_{ele,j}$ over all reactions is the total heat source due to volume reactions:

$$Q = \sum_i Q_j + Q_{ele,j}$$

This appears as an option in the **General source** list in the **Heat Source** node in any of the **Heat Transfer** interfaces called **Heat source for gas.** This is how reactions in a plasma heat the background gas.

Species Transport Properties

The diffusion coefficients are computed from the following expression based on kinetic gas theory:

$$D_{kj} = 2.66 \cdot 10^{-2} \cdot \frac{\sqrt{T^3(M_k + M_j)/(2 \cdot 10^3 M_k M_j)}}{p\sigma_{kj}^2 \Omega_D} \tag{5-14}$$

Here:

- D_{kj} is the binary diffusion coefficient (SI unit: m^2/s)

- M equals the molecular weight (SI unit: kg/mol)
- T represents the temperature (SI unit: K)
- p is the pressure (SI unit: Pa), and
- σ equals the characteristic length (SI unit: Å) of the Lennard-Jones/Stockmayer potential.
- In addition, Ω_D is the collision integral, given by the following equation (Ref. 2 and Ref. 3):

$$\begin{aligned}\Omega_D &= A(T^*)^{-B} + C[\exp(-DT^*)] + E[\exp(-FT^*)] \\ &\quad + G[\exp(-HT^*)] + \frac{0.19\delta_{kj}^2}{T^*}\end{aligned} \tag{5-15}$$

where

$$T^* = k_b \frac{T}{\varepsilon_{kj}}$$

$$\varepsilon_{kj} = (\varepsilon_k \varepsilon_j)^{1/2}$$

$$\sigma_{kj}^2 = \left(\frac{\sigma_k + \sigma_j}{2}\right)^2$$

$$\delta = \frac{1}{2}\frac{\mu^2}{\varepsilon\sigma^3} \tag{5-16}$$

In Equation 5-16, μ is the species dipole moment (SI unit: Debye). For gases at moderate pressure, the binary diffusivity can be used for the multicomponent Maxwell–Stefan diffusivities.

The predefined expression for the dynamic viscosity is given by the kinetic gas theory (Ref. 2 and Ref. 3) as

$$\mu_i = 2.669 \cdot 10^{-6} \frac{\sqrt{T(M_i \cdot 10^3)}}{\sigma_i^2 \Omega_v} \tag{5-17}$$

Here μ represents the dynamic viscosity (SI unit: N·s/m^2), and Ω_v is the collision integral, given by

$$\Omega_v = A(T^*)^{-B} + C[\exp(-DT^*)] + E[\exp(-FT^*)] + \frac{0.20\delta^2}{T^*}$$

with

$$T^* = k\frac{T}{\varepsilon}$$

$$\delta = \frac{1}{2}\frac{\mu^2}{\varepsilon\sigma^3} \qquad (5\text{-}18)$$

In Equation 5-18, μ is the species dipole moment (SI unit: Debye). The predefined expression for thermal conductivity comes from the Stiel-Thodos equation (Ref. 4)

$$k = \frac{\eta}{M}(1.15C_p + 0.88R_g) \qquad (5\text{-}19)$$

where the thermal conductivity k is in (SI unit: W/(m·K)). Here C_p denotes the heat capacity (SI unit: J/(mol·K)).

Mixture Transport Properties

The transport properties of the mixture are computed using the following for specific heat capacity (SI unit: J/(kg·K))

$$c_{p\mathrm{mix}} = \frac{\sum_i x_i C_{p,\,i}}{\sum_i x_i M_i}$$

The thermal conductivity (SI unit: W/(m·K)) is computed using:

$$k_{\mathrm{mix}} = \frac{1}{2}\left[\sum_{i=1} x_i k_i + \left(\sum_{i=1} \frac{x_i}{k_i}\right)^{-1}\right]$$

In these equations:

- T is the temperature (SI unit: K)
- x is the mole fraction
- M is the molecular weight (SI unit: kg/mol)
- c_p is the heat capacity per unit mass (SI unit: J/(kg·K))

- C_p represents the heat capacity per mole (SI unit: J/(mol·K)), and
- k is the thermal conductivity (SI unit: W/(m·K)).

The dynamic viscosity (SI unit: N·s/m^2) is computed using:

$$\mu_{\text{fluid}} = \sum_{i=1} \frac{x_i \mu_i}{\sum_{j=1} x_j \phi_{ij}}$$

where

$$\phi_{ij} = \frac{\left[1 + \left(\frac{\eta_i}{\eta_j}\right)^{0.5} \left(\frac{M_j}{M_i}\right)^{0.25}\right]^2}{\left[8\left(1 + \frac{M_i}{M_j}\right)\right]^{0.5}}$$

In these equations μ denotes the dynamic viscosity (SI unit: Ns/m^2).

Scaling

Because there can be a large variation in mass fraction of the components of a plasma, a logarithmic scaling of the equations is sometimes necessary. So, starting with the equation for a generalized mass transport template

$$\rho\frac{\partial}{\partial t}(w_k) + \rho(\mathbf{u} \cdot \nabla)w_k = \nabla \cdot \rho w_k \mathbf{V}_k + R_k \tag{5-20}$$

and letting $W_k = \ln w_k$, Equation 5-20 becomes

$$\rho w_k \frac{\partial}{\partial t}(W_k) + \rho w_k(\mathbf{u} \cdot \nabla)W_k = \nabla \cdot [\rho w_k \mathbf{V}_k] + R_k$$

The first term on the right-hand side depends on which formulation is used. For the Mixture averaged diffusion model:

$$\mathbf{V}_k = D_{k,m}\nabla W_k + D_{k,m}\nabla \ln M + D^T \nabla \ln T – Z_k \mu_{k,m}\mathbf{E}$$

and for the **Fick's law** diffusion model:

$$\mathbf{V}_k = D_{k,f}\nabla W_k + D_{k,f}\nabla \ln M + D^T \nabla \ln T – Z_k \mu_{k,f}\mathbf{E}\,.$$

You can switch between the standard and logarithmic formulations of the mass balance equation. In this module the default implementation is logarithmic.

Stabilization

When the source stabilization is on, an additional term is added to the rate expression, Equation 5-7. The expression takes the form:

$$R_{k,\,\mathrm{tot}} = R_k + \exp(-\iota \ln(n_k))$$

where ι is the user-defined tuning parameter. When the species number density is very low, this acts as a source term which prevents the mass fraction from approaching zero. As the number density increases this term becomes exponentially smaller, eventually becoming negligible for high number densities.

Surface Reactions

The equations solved for the transport for each species need to be supplemented by appropriate boundary conditions. This is often accomplished by way of a set of suitable surface reactions, which describe how species are created or consumed on surfaces. Assume a surface mechanism with I surface reactions and K chemical species

$$\sum_{k=1}^{K} \nu^{f}_{ki} c_k \Leftrightarrow \sum_{k=1}^{K} \nu^{r}_{ki} c_k$$

where ν_{ki} are the stoichiometric coefficients and c_k is the concentration of the kth species. The surface rate expression (SI unit: mol/(m^2·s)) for species k is defined as

$$R_{\mathrm{surf},\,k} = \sum_{i=1}^{I} \nu_{ki} q_i$$

where $\nu_{ki} = \nu^{\mathrm{r}}{}_{ki} - \nu^{\mathrm{f}}{}_{ki}$. The rate of progress of the ith reaction is governed by the law of mass action:

$$q_i = k_{f,\,i} \prod_{k=1}^{K} c_k^{\nu^{f}_{ki}} - k_{r,\,i} \prod_{k=1}^{K} c_k^{\nu^{r}_{ki}}$$

where the following definitions apply for c_k.

Gas Phase Species

The molar concentration is given by:

$$c_k = \frac{w_k \rho}{M_k}$$

where w_k is the mass fraction of species k, ρ is the gas density, and M_k is the molecular weight of species k. When the Motz-Wise correction option is set to On, the forward rate constant is related to the sticking coefficients via:

$$k_{\mathrm{f},\,i} = \left(\frac{\gamma_i}{1-\gamma_i/2}\right)\frac{\prod \sigma_j^{\nu_{ji}}}{(\Gamma_{\mathrm{tot}})^m}\left(\frac{1}{4}\right)\sqrt{\frac{8RT}{\pi M_k}} \qquad (5\text{-}21)$$

When the Motz-Wise correction option is set to Off, the forward rate constant is given via:

$$k_{\mathrm{f},\,i} = \gamma_i \frac{\prod \sigma_j^{\nu_{ji}}}{(\Gamma_{\mathrm{tot}})^m}\left(\frac{1}{4}\right)\sqrt{\frac{8RT}{\pi M_k}}\,. \qquad (5\text{-}22)$$

where Γ_{tot} is the total surface site concentration (SI unit: $\mathrm{mol/m^2}$). The *sticking coefficient* can be a constant or even temperature dependent through an Arrhenius expression:

$$\gamma_i = a_i T^{b_i} \exp\left(-\frac{e_i}{RT}\right)$$

where a, b, and e are constants. Since γ is a probability it must be between 0 and 1 to be physically meaningful.

Flux Matching

Flux matching must be performed for each species in order for the total molar mass in the system to be conserved. If a model also contains a physics interface to compute the fluid velocity and pressure (the Laminar Flow interface for example) and the gas temperature (the Heat Transfer in Fluids interface for example) then those physics interfaces require appropriate flux matching boundary conditions. The Heavy Species Transport interface automatically computes flux matching variables for the total mass flux and heat source of reaction on surfaces.

Surface Chemistry Tutorial Using the Plasma Module: Application Library path **Plasma_Module/Chemical_Vapor_Deposition/surface_chemistry_tutorial**

FLUX MATCHING FOR EACH SPECIES

The inward mass flux for species k on a reacting surface is given by:

$$\mathbf{n} \cdot \mathbf{j}_k = M_k R_{\text{surf}, k} \tag{5-23}$$

This is the natural boundary condition for Equation 5-4 and results in either an inward or outward mass flux for species k, depending on the surface chemical mechanism stoichiometry. If the species is an ion, then there is an additional flux contribution due to migration. In this case, Equation 5-23 becomes:

$$\mathbf{n} \cdot \mathbf{j}_k = M_k R_{\text{surf}, k} + M_k c_k \mu_{m, k} z_k (\mathbf{n} \cdot \mathbf{E})[(z_k \mathbf{n} \cdot \mathbf{E}) > 0]$$

This additional flux contribution ensures that when the normal electric field is directed to the wall, there is an outflux of ions due to migration.

FLUX MATCHING FOR THE TOTAL MASS FLUX

If a model contains the Laminar Flow interface, an expression for the mass averaged velocity is needed on reacting surfaces. The Heavy Species Transport interface creates an appropriate variable for the inward mass flux, which can be accepted as a feature input for the Mass Flow option in the Inlet boundary condition of the Laminar Flow interface.

The inward mass flux, M_f is defined as:

$$M_f = \sum_{k=1}^{K_g} M_k R_{\text{surf}, k}$$

where $R_{\text{surf},k}$ is the surface rate expression for each species which comes from summing the surface reaction rates multiplied by their stoichiometric coefficients over all surface reactions:

$$R_{\text{surf}, k} = \sum_{i=1}^{I} \nu_{ki} q_i \text{ .}$$

FLUX MATCHING FOR THE TOTAL HEAT FLUX

If a model contains the Heat Transfer in Fluids interface, an expression for the total heat source of reactions is needed. The Heavy Species Transport interface creates an appropriate variable for the total heat source of reactions, which can be accepted as a feature input in the Reaction Heat Flux boundary condition of the Heat Transfer in Fluids interface.

The total heat source due to reactions is defined as:

$$Q_b = \sum_{i=1}^{I} q_i h_i$$

where h_i is the molar enthalpy change due to surface reaction i. Depending on the surface chemical mechanism stoichiometry, this can either cause heating (exothermic) or cooling (endothermic).

About Surface Species

For each of the k Surface Species the following equations are solved:

$$\frac{dZ_k}{dt} = \frac{R_{\mathrm{surf},\, k}}{\Gamma_{\mathrm{tot}}}$$

where Z_k is the site fraction (dimensionless), $R_{\mathrm{surf},k}$ is the surface rate expression (SI unit: mol/m^2) and Γ_{tot} is the total surface site concentration, as specified in the Surface Chemistry section of the physics interface settings. The quantity Z_k is the surface equivalent of the mole fraction on the volumetric level. That is, the sum of the site fractions must always equal one:

$$\sum Z_k = 1 .$$

Empty sites are not accounted for in reaction rate coefficients, so all species which can occupy a surface must be included in the model. The surface site concentration is given by:

$$c_k = \frac{Z_k \Gamma_{\mathrm{tot}}}{\sigma_k} .$$

For each of the bulk surface species, the following equation is solved for the deposition height:

$$\frac{dh_k}{dt} = -\frac{R_{\mathrm{surf},\, k} M_k}{\rho_k}$$

where h_k is the total growth height (SI unit: m), M_k is the molecular weight (SI unit: kg/mol), and ρ is the density of the bulk species (SI unit: kg/m^3).

Limitations of the Heavy Species Transport Equations

Before continuing, some comments on the validity of the equations are necessary. Despite their relatively complicated form, the equations for transport of the heavy species are formulated under the following assumptions:

- The diffusion velocity must be much less than the thermal velocity; that is, $\mathbf{V}_k << v_k$. In the case of very low pressure CCP reactors, the ion drift velocity can approach or exceed the ion thermal velocity. In this case, the heavy species transport equations fail to fit the physics of the discharge.
- A single temperature is assumed for heavy species. In the case of reactors with high reduced electric fields, the ions can have a different temperature than the surrounding gas.

Global Model

If the **Diffusion Model** property is set to **Global**, the model equations solved are greatly simplified because the spatial information of the different quantities in the plasma reactor are treated as volume-averaged. Without the spatial derivatives the numerical solution of the equation set becomes considerably simpler and the computational time is greatly reduced. This type of model is useful when investigating a broad region of parameters with complex plasma chemistries.

To avoid unnecessary complexity in the notation the same symbols which were used to represent the space-dependent quantities are used to represent the volume-averaged quantities. In the following the different densities, mass fractions and other quantities are to be thought as volume-averaged.

For a mixture consisting of $k = 1, \ldots, Q$ species and $j = 1, \ldots, N$ reactions the mass-fraction balance equations for the first $Q - 1$ species is given by

$$\begin{aligned} V\rho\frac{u}{dt}(w_k) &= m_f w_{f,k} - m_o w_k + VR_k + \sum_l h_l A_l R_{surf,k,l} M_k \\ &\quad - w_k\left(m_f + m_o + \sum_l h_l A_l M_{f,l}\right) \end{aligned}$$

where:

- m_f is the total feed mass-flow rate (SI unit: kg/s)
- m_o is the outlet mass-flow rate (SI unit: kg/s)
- $w_{f,k}$ is the mass fraction of species kth in the feed (1)
- h_l is a correction factor of surface l (1)
- A_l is the area of surface l (SI unit: m^2)
- V is the reactor volume (SI unit: m^3),
- $R_{surf,k,l}$ is the surface rate expression of surface l (SI unit: $\mathrm{mol/(m^2{\cdot}s)}$).
- $M_{f,l}$ is the inward mass flux of surface l (SI unit: $\mathrm{kg/(m^2{\cdot}s)}$).

The sum in the last two terms is over surfaces where species are lost or created. The fourth term on the right hand side accounts for surface reactions of species kth. The last term on the right hand side is introduced because the species mass-balance equations are written in the nonconservative form and it is used the mass-continuity equation to replace for the mass density time derivative.

To take into account possible variations of the system total mass or pressure the mass-continuity equation can also be solved

$$V\frac{d\rho}{dt} = m_f - m_o + \sum_l h_l A_l M_{f,l}\,. \qquad (5\text{-}24)$$

CLOSED REACTOR

In the **Reactor** property it is possible to choose three different types of reactor models. If the **Reactor Type** is set to **Closed Reactor** the mass-flow feeds are set to zero

$$m_f = m_o = 0 \qquad (5\text{-}25)$$

and Equation 5-24 is solved to take into account possible mass changes in the system caused by surface reactions.

CONSTANT MASS

If the **Reactor Type** is set to **Constant Mass** the outlet mass-flow is set such that the feed and surface reactions cannot change the mass of the system

$$m_o = m_f + \sum_l h_l A_l M_{f,l} \qquad (5\text{-}26)$$

and the pressure in the reactor is found in order to keep the mass-density constant

$$\frac{d\rho}{dt} = 0\,. \tag{5-27}$$

CONSTANT PRESSURE

If in the **Reactor** property the **Reactor Type** is set to **Constant Pressure** Equation 5-24 is not solved and the outlet mass-flow feed is set such that the mass feed and surface reactions cannot change the mass of the system

$$m_o = m_f + \sum_l h_l A_l M_{f,l}\,. \tag{5-28}$$

This condition ensures that the feed and surface reactions cannot change the mass density. However, the mass density can still change as a result of a change in the mean molar mass of the gas mixture in order to maintain a constant pressure.

GLOBAL MODEL SURFACE REACTIONS

A large part of a successful implementation of a plasma global modal depends on how the surface losses are estimates. For positive ions the forward rate constant can be estimate to be equal to the Bohm velocity

$$k_{\mathrm{f},i} = \sqrt{\frac{eN_aT_e}{M_k}} \tag{5-29}$$

where T_e is the electron temperature (V).

For neutral species the forward rate constant can be estimated by

$$\frac{1}{k_{\mathrm{f},i}} = \frac{1}{k_{\gamma,i}} + \frac{1}{k_{D,i}} \tag{5-30}$$

$$k_{D,i} = \frac{D_{k,m}}{\Lambda^2{}_{\mathrm{eff}}}\frac{V}{A_l}\,. \tag{5-31}$$

When the **Motz-Wise correction** option is set to **On**

$$k_{\gamma,i} = \left(\frac{\gamma_i}{1-\gamma_i/2}\right)\frac{\prod \sigma_j^{\nu_{ji}}}{(\Gamma_{\mathrm{tot}})^m}\left(\frac{1}{4}\right)\sqrt{\frac{8RT}{\pi M_n}} \tag{5-32}$$

and when the **Motz-Wise correction** option is set to **Off**, the forward rate constant is given via:

$$k_{\gamma,\,i} = \gamma_i \frac{\prod \sigma_j^{\nu_{ji}}}{(\Gamma_{\text{tot}})^m}\left(\frac{1}{4}\right)\sqrt{\frac{8RT}{\pi M_n}}\,. \tag{5-33}$$

Equation 5-31 is an estimation of the diffusive losses to the wall, where Λ_{eff} is the effective diffusion length, and $D_{k.m}$ is the mixture-average diffusion coefficient of species k.

Surface reactions can be adjusted using the **Correction factor** h_l. It is common practice to correct the surface ion losses by a factor that takes into account the spatial profile of the ions. Models for the ion correction factor can be found in the literature Ref. 11 and Ref. 12.

References for the Heavy Species Transport Interface

1. S. Gordon and B.J. McBride, *Computer Program for Calculation of Complex Chemical Equilibrium Compositions, Rocket Performance, Incident and Reflected Shocks, and Chapman-Jouquet Detonations*, NASA-SP-273, 1971.

2. P.D. Neufeld, A.R. Janzen, and R.A. Aziz, "Empirical Equations to Calculate 16 of the Transport Collision Integrals $\Omega^{(l,\,s)*}$ for the Lennard-Jones (12–6) Potential", *J. Chem. Phys.*, vol. 57, p. 1100, 1972.

3. R.S. Brokaw, "Predicting Transport Properties of Dilute Gases", *Ind. Eng. Process Design Develop.*, vol. 8, pp. 240–253, 1969.

4. L.I. Stiel and G. Thodos, "The Viscosity of Polar Substances in the Dense Gaseous and Liquid Regions", *AIChE J.*, vol. 10, p. 26, 1964.

5. R.J. Kee, M.E. Coltrin, and P. Glarborg, *Chemically Reacting Flow Theory and Practice*, Wiley, 2003.

6. R.B. Bird, W.E. Stewart, and E.N. Lightfoot, *Transport Phenomena*, John Wiley & Sons, 2002.

7. G.J.M. Hagelaar, *Modeling Of Microdischarges for Display Technology*, PhD thesis, Universiteitsdrukkerij Technische Universiteit Eindhoven, 2000.

8. B.M. Smirnov, *Theory of Gas Discharge Plasma*, Springer 2015.

9. E.W. McDaniel, *Collision Phenomena in Ionized Gases*, John Wiley and Sons, 1964.

10. H.W. Ellis, R.Y. Pai, E.W. McDaniel, E A. Mason, and L.A. Viehland, “Transport Properties of Gaseous Ions Over a Wide Energy Range”*, Atomic Data and Nuclear Data tables*, vol. 17, pp. 177–210, 1976.

11. M.A. Lieberman and A.J. Lichtenberg, *Principles of Plasma Discharges and Materials Processing*, John Wiley & Sons, 2005.

12. E.G. Thorsteinsson, and J.T. Gudmundsson, “A global (volume averaged) model of a chlorine discharge”, *Plasma Sources Sci. Technol.*, vol. 19, p. 015001 (15pp), 2010.

13. A.V. Phelps, “Cross Sections and Swarm Coefficients for Nitrogen Ions and Neutrals in N2 and Argon Ions and Neutrals in Ar for Energies from 0.1eV to 10keV”, *J. Phys. Chem. Ref. Data*, vol. 20, no. 3, pp. 557–573, 1991.

6

Plasma Interfaces

This chapter describes the Plasma Module interfaces found under the **Plasma** branch ( ).

In this chapter:

- Computing Versus Assuming the EEDF
- Using Stationary Studies in Plasma Models
- Electronegative Discharges

The Plasma Interface

The **Plasma (plas)** interface ( ), found under the **Plasma** branch ( ) couples the Drift Diffusion, Heavy Species Transport, and Electrostatics interfaces into an integrated multiphysics interface to model plasma discharges.

When this physics interface is added, these default nodes are also added to the **Model Builder**: **Plasma Model**, **Zero Charge**, **Insulation**, and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions and velocity. You can also right-click **Plasma** to select physics features from the context menu.

> Except where described in this section, this physics interface shares it physics nodes with The Drift Diffusion Interface, The Heavy Species Transport Interface, and The Electrostatics Interface.

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `plas`.

OUT-OF-PLANE THICKNESS (2D, 1D AXISYMMETRIC, AND 1D)

- For 2D components, enter a **Thickness** d (SI unit: m). The default is 1 m.
- For 1D axisymmetric components, enter a **Vertical height** d_z (SI unit: m). The default is 0.01 m.
- For 1D components, enter a **Cross-section area** A (SI unit: m^2). The default is 0.01 m^2.

DIFFUSION MODELS

Select a **Diffusion model** — **Mixture-averaged** (the default), **Fick's law**, or **Global**. When using the **Mixture-averaged** or **Global** models, the mixture averaged diffusion coefficients are automatically computed based on the data specified for each species.

TRANSPORT SETTINGS

Select the check boxes for which transport mechanisms to **Include** — **Convection**, **Migration in electric field**, **Calculate thermodynamic properties**, **Full expression for diffusivity**, or **Compute tensor ion transport properties**. The selection changes the number of Model Inputs requiring values on the **Plasma Model** page. Note the following:

- The **Migration** term is part of the relative mass flux vector.
- **Convection** of heavy species present in a plasma can often be neglected due to the low operating pressure.
- For **Calculate thermodynamic properties** select that the thermodynamic properties of each reaction and species are computed automatically based on the thermodynamic properties of each species.
- For **Full expression for diffusivity** it computes a more accurate expression for the Maxwell–Stefan diffusivities. Often the additional correction terms are negligible in which case the expressions are much simpler and the time taken to assemble the Jacobian matrix is reduced.
- For **Mixture diffusion correction** additional terms are included in the definition of the mass flux vector to ensure that the same solution is obtained regardless of the choice of the species which comes from the mass constraint. This option makes the problem more nonlinear and strongly coupled, and is only necessary when the molecular weights of the species differ substantially (such as a mixture of sulfur hexafluoride and hydrogen).
- For **Compute tensor ion transport properties** the tensor form of the ion transport properties when a static magnetic field is present is computed. This option only needs to be activated when a strong DC magnetic field exists and the operating pressure is very low (on the order of millitorr). When this option is activated an expression must be provided for the magnetic flux density, which typically another physics interface computes. This is set in the **Plasma Model** node.

When the **Diffusion model** is set to **Global** only the properties **Calculate thermodynamic properties** and **Full expression for diffusivity** are available.

HEAVY SPECIES ENERGY BALANCE

This section is only available when the **Diffusion model** is set to **Global.** Select the check box **Include heavy species energy conservation equation** to include the heavy species energy equation that solves for the background gas temperature.

PLASMA PROPERTIES

Select the **Compute tensor electron transport properties** or **Use reduced electron transport properties**, or **Include thermal diffusion** check boxes as needed and select the model for the **Mean electron energy** — **Local energy approximation** (default), **Local field approximation**, or **Fix mean electron energy**.

When the **Diffusion model** is set to **Global** only the check boxes **Compute tensor electron transport properties** and **Include thermal diffusion** are not available.

Compute Tensor Electron Transport Properties

Select **Compute tensor electron transport properties** to compute the tensor form of the electron mobility, electron diffusivity, energy mobility and energy diffusivity. This should only be used in cases where a strong DC magnetic field exists. Two quantities must be supplied, both of which are in the Plasma Model node. The DC mobility which is the value of the electron mobility in the absence of a DC magnetic field and the magnetic flux density which would typically be computed by another physics interface.

Use Reduced Electron Transport Properties

Select **Use reduced electron transport properties** to specify the electron mobility, diffusivity, energy mobility and energy diffusivity in reduced form. The neutral number density is then specified in the Drift Diffusion Model node. The electron transport properties are computed from the reduced transport properties using:

$$\mu_e = \frac{\mu_{\mathrm{red}}}{N_n}$$

where N_n is the user-defined neutral number density.

Include Thermal Diffusion

The **Include thermal diffusion** check box adds an additional term to the definition of the electron current due to gradients in the electron diffusivity. If the diffusivity is a constant then including this does not affect the solution. It is only necessary to include this term if the electron diffusivity is a function of the electron temperature, and there are significant gradients in the electron temperature.

Mean Electron Energy

Select **Local energy approximation** (default) to solve the mean electron energy equation self-consistently with the continuity, momentum, and Poisson’s equations, and to use the mean electron energy to parameterize transport and source coefficients. This is the most numerical demanding option to find the mean electron energy because of the strong coupling between the mean electron energy and the electromagnetic fields.

If **Local field approximation** is selected, it is assumed that transport and source coefficients are well parameterized through the reduced electric field (E/N_n). If **Boltzmann equation, two-term approximation (linear)** or **Boltzmann equation, two-term approximation (quadratic)** is selected, the computed E/N_n from the fluid plasma equations is used directly as input to the Boltzmann equation in the two–term approximation and the relation between the reduced electric field is automatically computed. If the Boltzmann equation is not solved, the relation between the reduced electric field and the mean electron energy needs to be provided in the section **Mean Electron Energy Specification** in the **Plasma Model** node.

When using the local field approximation the fluid equation for the mean electron energy is not solved, which reduces significantly the complexity of the numerical problem. The local field approximation is valid in a situation where the rate of electron energy gain from the electric field is locally balanced by the energy loss rate. When this condition is met the electrons are said to be in local equilibrium with the electric field and the electron mean properties can be expressed as a function of the reduced electric field.

Select **Fix mean electron energy** to fix the mean electron energy to its initial value. This can be useful in some situations because the strong coupling between the mean electron energy and the electromagnetic fields is removed. This allows for non-self-consistent models to be created quickly, since problems where the mean electron energy is fixed are easier to solve numerically.

REACTOR

This section is available when the **Diffusion model** is set to **Global**. Select a **Reactor type** — **Closed reactor** (the default), **Constant mass**, or **Constant pressure**.

- **Closed reactor** solves a closed system where mass and pressure can change, for example, as the result of surface reactions and volume reactions of the associative/ dissociative type.

- **Constant mass** solves a system with mass-flow feed and outlet. The mass-flow outlet is set to keep the mass-density constant.
- **Constant pressure** solves a system with mass-flow feed and outlet. The pressure is kept constant by adjusting the system mass-density if needed.

For more information on the electron energy distribution function (EEDF), see the Theory for the Boltzmann Equation, Two-Term Approximation Interface.

The options **Closed reactor** and **Constant mass reactor** are not supported for stationary studies. If a model uses those options and an attempt is made to use a stationary study an error message appears. Only the **Constant pressure** option is supported for stationary studies.

ELECTRON ENERGY DISTRIBUTION FUNCTION

If cross section data is used to define source coefficients in the model then an electron energy distribution function (EEDF) needs to be selected. Select one of these options.

- **Maxwellian**. This option assumes a Maxwellian EEDF which takes the form:

$$f(\varepsilon) = \varphi^{-3/2}\beta_1\exp(-(\varepsilon\beta_2/\varphi))$$

where

$$\beta_1 = \Gamma(5/2)^{3/2}\Gamma(3/2)^{-5/2}, \beta_2 = \Gamma(5/2)\Gamma(3/2)^{-1}$$

where φ is the mean electron energy (SI unit: eV), ε is the electron energy (SI unit: eV) and Γ is the incomplete gamma function:

$$\Gamma(s) = \int_0^\infty u^{s-1}e^{-u}du\,.$$

- **Druyvesteyn**. This option assumes a Druyvesteyn EEDF which takes the form:

$$f(\varepsilon) = 2\varphi^{-3/2}\beta_1\exp(-(\varepsilon\beta_2/\varphi)^2)$$

where

$$\beta_1 = \Gamma(5/4)^{3/2}\Gamma(3/4)^{-5/2}, \beta_2 = \Gamma(5/4)\Gamma(3/4)^{-1}$$

- **Generalized**. Use this option for a generalized distribution function where the EEDF is somewhere between Maxwellian and Druyvesteyn. For this option, specify a power law. This number must be between 1 and 2. Mathematically, the EEDF takes the form:

  $$f(\varepsilon) = g\varphi^{-3/2}\beta_1\exp(-(\varepsilon\beta_2/\varphi)^g)$$

  where

  $$\beta_1 = \Gamma(5/2g)^{3/2}\Gamma(3/2g)^{-5/2}, \beta_2 = \Gamma(5/2g)\Gamma(3/2g)^{-1}, 1 \le g \le 2$$

- **Function**. If a two-dimensional interpolation function has been added to the model, it can be used for the EEDF. In this case, the *x*-data should be the electron energy (eV) and the *y*-data should be the mean electron energy (eV).

The two-dimensional interpolation function can be computed using a parametric sweep in The Boltzmann Equation, Two-Term Approximation Interface. This allows for modeling of discharges where the EEDF is far from Maxwellian. For step-by-step instructions on how to do this, refer to this blog entry: https://www.comsol.com/blogs/the-boltzmann-equation-two-term-approximation-interface/

See Interpolation in the *COMSOL Multiphysics Reference Manual*.

- **Boltzmann equation, two-term approximation (linear)** and **Boltzmann equation, two-term approximation (quadratic)**. Use these options to solve a two-term approximation of the Boltzmann equation.

When **Boltzmann equation, two-term approximation (linear)** or **Boltzmann equation, two-term approximation (quadratic)** is selected, the EEDF is computed from a partial differential equation instead of taking an assumed function. The following options are available:

- Select the **Temporal behavior** — **Stationary EEDF** (the default), or **Time dependent EEDF**. This option is only available when **Mean electron energy** is set to **Local field approximation** and **Diffusion model** is set to **Global**.
- Select **Electron-electron collisions** (off by default) if the ionization degree of the discharge is high. The ionization degree and the electron density necessary to model the electron-electron collisions are obtained from the solution of the model

equations every time step or iteration. Including this option dramatically increases the computational demand, so it should be avoided if at all possible.

- Select **Equal secondary electron energy sharing** (on by default) to describe how the energy is split between two electrons when an ionization collision occurs. If selected, then both electrons take an equal energy after the collision. If not selected, the secondary electron created in an ionization collision has zero energy and the ionizing electron carries all the excess energy.
- Select **Oscillating field** if the reduced angular frequency of the discharge is high, which is typically only true for microwave discharges. The reduced angular frequency is the ratio of the angular frequency and the neutral number density, ω/N.
- If the **Oscillating field** property is active, enter a **Reduced angular frequency** ω/N (SI unit: m^3/s). The default is 10^{-13} m^3/s. If the reduced angular frequency is high, the proportion of electrons with high energies is substantially increased for the same mean electron energy. This is because in DC fields, collisional momentum transfer impedes electrons acquiring higher energies but high frequency fields have the opposite effect.
- Enter the **Number of elements in eedf extra dimension** N (SI unit: dimensionless) to specify the number of mesh elements to use to discretize the underlying energy space. The default is 100, but models with complex gas mixtures may require more.
- Enter the **Element ratio in eedf extra dimension** R (SI unit: dimensionless) to specify the rate at which the mesh coarsens away from the zero energy coordinate. The default is 10, and this does not usually need to be changed. Higher values mean that the mesh will be finer closer to the zero energy, and coarser at higher energies.
- Select **Compute maximum energy** to have the software automatically compute the maximum energy coordinate based on certain assumptions about the EEDF. This is a powerful feature, in that the maximum energy coordinate does not need to be specified, but it does make the problem more nonlinear and thus difficult to solve.
- Enter the **Maximum energy** ε_{max} (SI unit: V) to specify the maximum coordinate in energy space on which we are computing the EEDF. When computing the EEDF at high mean energies, this value may need increasing from its default value of 100 V. This option is only available if **Compute maximum energy** is not selected.
- Enter the **Maximum energy multiplication factor** χ_ε (SI unit: dimensionless) to specify how much the maximum energy coordinate should be scaled. This option is only

available if **Compute maximum energy** is selected and **Mean electron energy** is set to **Local field approximation**.

- Enter the **EEDF minimum value** f_{min} (SI unit: dimensionless) to specify the minimum value that the EEDF should take at the maximum energy coordinate. In order to avoid divide by zero problems, the value should be small and positive. The default value of 1E-15 rarely needs to be changed.

In all these cases the rate constants in the model are automatically computed based on the selected EEDF using the formula:

$$k_{\mathrm{f}} = \gamma\int_0^{\infty} \varepsilon\sigma_k(\varepsilon)f(\varepsilon)d\varepsilon\,. \tag{6-1}$$

The rate coefficients when computed using cross section data are a highly nonlinear function of the mean electron energy. COMSOL Multiphysics automatically computes the integral in Equation 6-1 and makes the result available for evaluation of the rate coefficient. The variation of the rate coefficient for any particular model can be plotted using `<name>.kf_<reaction number>`. For example, for reaction number 3 in the Plasma interface, with name `plas`, the rate coefficient is plotted using `plas.kf_3`.

STABILIZATION

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

If the **Equation formulation** is set to **Log** then the solver can run into difficulties as the species mass fractions approach zero. The **Reaction source stabilization** check box (selected by default) adds an additional source term to the rate expression for each species. In the ι field, enter a tuning parameter for the source stabilization. The default value is 1. This value is usually good enough. If the plasma is high pressure (atmospheric) then it can help to lower this number to somewhere in the range of 0.25–0.5.

The solver can also run into difficulties as the electron density or electron energy density approach zero. The **Source stabilization** check box (selected by default) adds an additional source term to the equation for the electron density and electron energy density. In the ζ field, enter a tuning parameter for the source stabilization. The default value is 1. This value is usually good enough. If the plasma is high pressure (atmospheric) then it can help to lower this number to somewhere in the range of 0.25–0.5.

CONSISTENT STABILIZATION

To enable this section, click the **Show More Options** button (👁) and select **Stabilization** in the **Show More Options** dialog box. This section is only available if one of the following **Finite element (linear shape function)** or **Finite element (quadratic shape function)** are used.

It is possible to add **Streamline diffusion for electrons** and **Streamline diffusion for ions** independently. Both check boxes for these methods are selected by default and should remain selected for optimal performance.

Streamline diffusion stabilization is a numerical technique to stabilize the numeric solution of convection-dominated PDEs by adding diffusion in the directions of the streamlines.

INCONSISTENT STABILIZATION

To enable this section, click the **Show More Options** button (👁) and select **Stabilization** in the **Show More Options** dialog box. This section is only available if one of the following discretizations is used: **Finite element (linear shape function)**, **Finite element (quadratic shape function)**, **Finite element, log formulation (linear shape function)**, or **Finite element, log formulation (quadratic shape function)**.

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

If required, select the **Isotropic diffusion for electrons** or **Isotropic diffusion for ions** check boxes and enter a **Tuning parameter for electrons** $\delta_{\mathrm{id,e}}$ and/or a **Tuning parameter for ions** $\delta_{\mathrm{id,i}}$. The default value is 0.25.

DISCRETIZATION

Select **Formulation** — **Finite Volume (constant shape function)**, **Finite element, log formulation (linear shape function)** (the default), **Finite element (linear shape function)**, **Finite element, log formulation (quadratic shape function)**, or **Finite element (quadratic shape function)**. The finite element options use the Galerkin method to discretize the equations whereas the **Finite Volume (constant shape function)** option creates degrees of freedom for the dependent variables which are piecewise constant within each mesh element. For the charged species, Scharfetter–Gummel upwinding is used. For charge neutral species and the electrostatic field, a centered difference scheme is used. In cases where the ion and electron flux are strongly driven by the electric field, the **Finite Volume (constant shape function)** option can be more stable. Examples where this is true include dielectric barrier and corona discharges.

The options with log formulation solve for the log of the dependent variables, ensuring that the mass fraction of any of the species is never lower than zero. This makes it more numerically stable but increases the nonlinearity of the equation system, and as such the model might take slightly longer to solve.

When the **Diffusion model** is set to **Global** the sections **Stabilization** and **Discretization** are not available.

DEPENDENT VARIABLES

The dependent variables (field variables) are the **Electron solution variable**, **Electron energy solution variable**, and **Electric potential**. The name can be changed but the names of fields and dependent variables must be unique within a model. The physical meanings of **Electron solution variable** and **Electron energy solution variable** change depending on the formulation used. For example, when log formulations are used, **Electron solution variable** is the logarithm of the electron density; whereas for other cases, it is the electron density

- Domain, Boundary, Edge, Point, and Pair Nodes for the Plasma Interface
- Plasma Reactors Theory and Capacitively Coupled Plasmas
- Stabilization
- Interpolation in the *COMSOL Multiphysics Reference Manual*

- *GEC ICP Reactor, Argon Chemistry*: Application Library path **Plasma_Module/Inductively_Coupled_Plasmas/argon_gec_icp**
- *DC Glow Discharge*: Application Library path **Plasma_Module/Direct_Current_Discharges/positive_column_2d**

Domain, Boundary, Edge, Point, and Pair Nodes for the Plasma Interface

The Plasma Interface has these domain, boundary, edge, point, and pair nodes, listed in alphabetical order, available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or right-click to access the context menu (all users).

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

In the *COMSOL Multiphysics Reference Manual* see Table 2-4 for links to common sections and Table 2-5 to common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

These nodes are described in this section:

- Plasma Model
- Initial Values
- Species Group
- Species Constraint Group

These nodes are described for the Drift Diffusion or Heavy Species Transport interface (listed in alphabetical order):

- Cross Section Import
- Electron Density and Energy
- Electron Impact Reaction
- Electron Outlet
- Electron Production Rate
- Flux
- General Power Deposition
- Insulation
- Reaction
- Reaction Group
- Species
- Surface Reaction
- Surface Species
- Velocity
- Wall

These nodes are described for the Electrostatics interface (listed in alphabetical order):

- Dielectric Contact
- Dielectric Shielding
- Distributed Capacitance
- Electric Displacement Field
- Electric Potential
- Electrostatic Point Dipole
- Floating Potential
- Ground
- Line Charge
- Metal Contact
- Periodic Condition
- Point Charge
- Surface Charge Accumulation
- Surface Charge Density
- Terminal
- Thin Low Permittivity Gap
- Zero Charge

Plasma Model

Use the **Plasma Model** node to set the electron transport properties, material properties, and model inputs.

MODEL INPUTS

This section has fields and values that are inputs to expressions that define material properties. If such user-defined property groups are added, the model inputs display here. Initially, this section is empty. The input depends on which **Transport Settings** are selected in the **Plasma** interface.

- Enter a **Temperature** T (SI unit: K) for the gas mixture used to compute the gas density and species transport properties.
- Enter an **Absolute pressure** p_A (SI unit: Pa) to compute the gas density and the Maxwell–Stefan diffusivities.
- If the **Convection** check box is selected, enter coordinates for the **Velocity field** $\mathbf{u}$ (SI unit: m/s).
- If the **Compute tensor electron transport properties** or the **Compute tensor ion transport properties** property is activated enter an expression for the **Magnetic flux density** B (SI unit: T). This usually comes from a magnetostatic model, and is used to compute anisotropic transport properties for electrons and ions.

If **Global** is selected in **Diffusion model** this section has the following fields:

- If **Include heavy species energy conservation equation** is selected enter the initial temperature of the background gas in **Initial temperature** T_0 (SI unit: K), otherwise enter the temperature of the background gas in **Temperature** T (SI unit: K).
- If the **Reactor type** is set to **Closed reactor** or **Constant mass** enter an **Initial pressure** p_0 (SI unit: Pa). If the **Reactor type** is set to **Constant pressure** enter an **Absolute pressure** p_A (SI unit: Pa).
- Enter an **Absolute electron Flux** $|\Gamma_e|$ (SI unit: $1/(\mathrm{m}^2.\mathrm{s})$) if there are reaction rate constants defined using **Townsend coefficient**.

ELECTRON DENSITY AND ENERGY

Select an option from the **Electron transport properties** list — **Specify mobility only** (the default), **From electron impact reactions**, **From electron energy distribution function**, **Specify all**, **Use lookup tables**, or **Mobility from electron energy distribution function**.

- The options available depend on whether the **Use reduced electron transport properties** physics interface property is selected and on what option is selected for **Electron energy distribution function** property. If the physics interface property **Compute tensor electron transport properties** is selected then only one field appears for the DC or reduced DC electron mobility, $\mu_\mathrm{dc}N_\mathrm{n}$. The other transport properties are computed using Einstein’s relation. The option **From electron energy distribution function** is only available if **Maxwellian**, **Druyvesteyn**, or **Generalized** are selected. The option **Mobility from electron energy distribution function** is only available when the **Boltzmann equation, two-term approximation (linear)** or **Boltzmann equation, two-term approximation (quadratic)** are selected.

Tensor Transport Properties

- For **Specify mobility only** enter a value or expression for the **Electron mobility** μ_e (SI unit: $\mathrm{m}^2/(\mathrm{V{\cdot}s})$). Select **Isotropic**, **Diagonal**, **Symmetric**, or **Anisotropic** based on the model. The electron diffusivity, energy mobility and energy diffusivity are automatically computed using Einstein’s relation for a Maxwellian EEDF:

$$\mathbf{D}_e = \mu_e T_e, \mu_\varepsilon = \left(\frac{5}{3}\right)\mu_e, \mathbf{D}_\varepsilon = \mu_\varepsilon T_e \tag{6-2}$$

For **Use reduced electron transport properties** enter the **Reduced electron mobility** $\mu_e N_n$ (SI unit: 1/(V·m·s)). The actual electron mobility is then computed by dividing the reduced electron mobility by the neutral number density.

- With **From electron impact reactions** the electron mobility is computed from

$$\mu_e = \frac{e}{m_e \nu} \tag{6-3}$$

where ν is the total electron collision frequency computed from

$$\nu = \frac{1}{c_e}\sum_{j=1}^{M} r_j \tag{6-4}$$

where r_j (unit: mol/(m^3·s)) is a reaction rate, c_e (unit: mol/m^3) is the electron molar concentration, and the sum is over all electron impact collisions. The electron diffusivity, energy mobility and energy diffusivity are automatically computed using Einstein's relation for a Maxwellian EEDF:

$$\mathbf{D}_e = \mu_e T_e, \mu_\varepsilon = \left(\frac{5}{3}\right)\mu_e, \mathbf{D}_\varepsilon = \mu_\varepsilon T_e \tag{6-5}$$

- With **From electron energy distribution function** the electron transport properties are computed directly from the EEDF using the following:

$$\mu_e N_n = -\left(\frac{\gamma}{3}\right)\int_0^\infty \frac{\varepsilon}{\sigma_m}\left(\frac{\partial f}{\partial \varepsilon}\right)d\varepsilon \tag{6-6}$$

$$D_e N_n = \left(\frac{\gamma}{3}\right)\int_0^\infty \frac{\varepsilon}{\sigma_m} f d\varepsilon \tag{6-7}$$

$$\mu_\varepsilon N_n = -\left(\frac{\gamma}{3\bar{\varepsilon}}\right)\int_0^\infty \frac{\varepsilon^2}{\sigma_m}\left(\frac{\partial f}{\partial \varepsilon}\right)d\varepsilon \tag{6-8}$$

$$D_\varepsilon N_n = \left(\frac{\gamma}{3\bar{\varepsilon}}\right)\int_0^\infty \frac{\varepsilon^2}{\sigma_m} f d\varepsilon \tag{6-9}$$

- For **Specify all** enter values or expressions for all of the properties, which can be either scalars or tensors. The appropriate values for the transport properties can be computed with The Boltzmann Equation, Two-Term Approximation Interface. For all of the properties, select **Isotropic**, **Diagonal**, **Symmetric**, or **Anisotropic** from the list based on the model, then enter values or expressions for all of these properties:

- **Electron mobility** μ_{e} (SI unit: $\mathrm{m}^2/(\mathrm{V{\cdot}s})$)
- **Electron diffusivity** D_{e} (SI unit: m^2/s)
- **Electron energy diffusivity** D_{en} (SI unit: m^2/s)
- **Electron energy mobility** μ_{en} (SI unit: $\mathrm{m}^2/(\mathrm{V{\cdot}s})$)

For **Use reduced electron transport properties** enter:

- **Reduced electron mobility** $\mu_{\mathrm{e}}N_n$ (SI unit: $1/(\mathrm{V{\cdot}m{\cdot}s})$)
- **Reduced electron diffusivity** $D_{\mathrm{e}}N_n$ (SI unit: $1/(\mathrm{m{\cdot}s})$)
- **Reduced electron energy diffusivity** $D_{\mathrm{en}}N_n$ (SI unit: $1/(\mathrm{m{\cdot}s})$)
- **Reduced electron energy diffusivity** $\mu_{\mathrm{en}}N_n$ (SI unit: $1/(\mathrm{V{\cdot}m{\cdot}s})$)

- For **Use lookup tables** enter or load a lookup table with the transport properties as listed above versus mean electron energy (eV).
- With **Mobility from electron energy distribution function** the electron mobility is computed directly from the EEDF using

$$\mu_e N_n = -\left(\frac{\gamma}{3}\right)\int_0^{\infty}\frac{\varepsilon}{\tilde{\sigma}_m}\left(\frac{\partial f}{\partial \varepsilon}\right)d\varepsilon\,. \tag{6-10}$$

The electron diffusivity, energy mobility and energy diffusivity are computed using Einstein's relation

$$\mathbf{D}_e = \mu_e T_e,\ \mu_\varepsilon = \left(\frac{5}{3}\right)\mu_e,\ \mathbf{D}_\varepsilon = \mu_\varepsilon T_e \tag{6-11}$$

See the Theory for the Boltzmann Equation, Two-Term Approximation Interface for more information.

For the **Use reduced electron transport properties** physics interface property, the appropriate values for the transport properties can be computed with The Boltzmann Equation, Two-Term Approximation Interface and written out to a text file. The text file can then be loaded in for each electron transport property.

MEAN ELECTRON ENERGY SPECIFICATION

This section is available when the **Mean electron energy** is set to **Local field approximation** and if **Maxwellian**, **Druyvesteyn**, or **Generalized** are selected from the **Electron energy distribution function** property. Select how to provide the relation between the reduce electric field and the mean electron energy from the **Specify using** list — **Expression** or **Use lookup table**.

If **Global** is selected in **Diffusion model** enter an **Electric potential** V (SI unit: V) used to compute the reduced electric field.

DENSITY

Select an option from the **Mixture density** list — **Ideal gas** or **User defined**. For **User defined** enter a value for ρ (SI unit: kg/m^3).

For **Ideal gas** it computes the expression for the gas density automatically using the ideal gas law, based on the composition of the mixture. The density is a function of the mean molecular weight:

$$\rho = \frac{p_A M}{RT}$$

where p_A is the absolute pressure, T is the temperature, R is the universal gas constant and M is the mean molecular weight given by:

$$\frac{1}{M} = \sum_{k=1}^{Q} \frac{w_k}{M_k}$$

TOTAL MASS FLOW

If **Global** is selected in **Diffusion model** and the **Reactor type** is set to **Constant mass** or **Constant pressure** enter the **Number of SCCM units** for the reactor feed. The **Feed mole fraction** of individual species can be set in the **Species** node ( ).

MEAN ELECTRON ENERGY

This section is available if **Global** is selected in **Diffusion model** and **Mean electron energy** is set to **Local energy approximation** or **Fix mean electron energy**. If **Fix mean electron energy** is selected enter the **Mean electron energy** ε (unit: V). If **Local energy approximation** is selected:

- Enter the **Power absorbed** P_{abs} (SI unit: W) by the plasma.
- The **Mean kinetic energy lost per electron lost** ε_e (unit: V).

- The **Mean kinetic energy lost per ion lost** ε_i (unit: V).
- The **Initial mean electron energy** ε_0 (unit: V).

HEAT TRANSFER TO SURFACES

This section is only available when the **Diffusion model** is set to **Global** and **Include heavy species energy conservation equation** is selected. Enter values or expressions for each of the following:

- The **Surface temperature** T_S (SI unit: K).
- The **Effective diffusion length** Λ_S (SI unit: K).

ADDITIONAL GAS HEATING

This section is only available when the **Diffusion model** is set to **Global** and **Include heavy species energy conservation equation** is selected. Enter a value or expression for the **Additional gas heat source** Q_0 (SI unit: W).

EEDF INPUTS

This section is available if **Global** is selected in **Diffusion model**, **Mean electron energy** is set to **Local field approximation**, and **Electron Energy Distribution Function** is set to **Boltzmann equation, two-term approximation (linear)** or **Boltzmann equation, two-term approximation (quadratic)**. Enter values or expressions for each of the following:

- Enter the **Reduced electric field** E/N (SI unit: V·m^2) for which the Boltzmann equation in the two-term approximation is solved.
- Enter an **EEDF initial mean electron energy** ε_0 (unit: V) to set the value of the mean electron energy of the Maxwellian EEDF used as initial condition.

Initial Values

Use the **Initial Values** node to set the initial values of the electron density, mean electron energy, and electric potential.

Initial conditions for the heavy species are specified on the individual species features.

INITIAL VALUES

Enter values or expressions for each of the following:

- **Initial electron density** $n_{e,0}$ (SI unit: $1/m^3$). This value represents some initial seed electrons in the plasma. If this number is too low then the plasma might not sustain itself. If this number is too high then numerical instabilities can arise at early time steps.
- **Initial mean electron energy** e_0 (SI unit: V). Enter an approximate guess for the mean electron energy in the reactor. COMSOL Multiphysics automatically computes the exact mean electron energy after solving but the closer the guess is set to the expected value the easier it is to find a solution.
- **Electric potential** V (SI unit: V). In nearly all cases the value should be zero. This initial value must also be consistent with any boundary conditions on the electric potential in the model.

When the **Diffusion model** is set to **Global** the initial values are set in the **Plasma Model** node.

Species Group

When using a stationary solver and when a chemical mechanism contains species made up of different elements, the default mass constraint (the sum of all mass fractions equals one) might not be good enough to ensure mass conservation over all elements. The **Species Group** feature can be used to ensure that the total elemental mass in the system is conserved. Consider a simplified Ar/O_2 plasma chemistry with an initial 70/30 ratio by mole fraction. For a closed system (which is currently the only type of system currently supported in the **Plasma, Time Periodic** interface), the sum of the mass fractions of oxygen species should equal the initial value, and the sum of the mass fractions of argon species should equal their initial values. If the following species are considered in the model: `Ar`, `Ars`, `Ar_1p`, `O2`, `O`, `O2_1p`, then ground state argon should be picked as the **From mass constraint** species, and a **Species Group** feature should be added with the following table:

TABLE 6-1: SPECIES GROUP TABLE FOR OXYGEN.

O2
O
O2_1p

Note that the first species in the table matters. The first species in the table is used to enforce mass conservation for this group of species, meaning that the total number of oxygen atoms in the system is conserved. The first species should correspond to a ground state species present in excess within this group. If additional species N2, N2A, N2_1p were added to the model, an additional **Species Group** feature would be necessary:

TABLE 6-2: SPECIES GROUP TABLE FOR NITROGEN.

N2
N2A
N2_1p

Species Constraint Group

For models with complex chemistries, obtaining a converged solution can be tricky, or not possible at all. The **Species Constraint Group** feature can be used in the following way to help in these cases:

- Add the species suspected to be causing nonconvergence to the **Species Constraint Group** feature. Instead of solving for these species, the mass fraction will be constrained to the initial value.
- Add a study and compute the solution with the troublesome species constrained to their initial values.
- Add a second study which uses the solution from the first study as the initial value. This is done by using the options in **Values of Dependent Variables** section of the study settings. In the same study settings, the Species Constraint Group feature can be disabled using the **Modify model configuration for study step** option.

This means that the troublesome species will begin to evolve based on a physically correct, converged initial solution, rather than from a “cold start”.

In the **Species Group** section, add potentially troublesome species to the **Species names** table. Note that the names must correspond to the exact name in the Species name field in the Species feature. For example, an argon ion should be entered into the table as Ar_1p and not Ar+.

The Plasma, Time Periodic Interface

The **Plasma, Time Periodic (ptp)** interface ( ), found under the **Plasma** branch ( ) is a special interface designed to model capacitively coupled plasmas (CCP).

When this physics interface is added, these default nodes are also added to the **Model Builder**: **Plasma Model**, **Zero Charge**, **Insulation**, and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions and velocity. You can also right-click **Plasma, Time Periodic** to select physics features from the context menu.

Except where described in this section, this physics interface shares it physics nodes with The Drift Diffusion Interface, The Heavy Species Transport Interface, and The Electrostatics Interface.

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `ptp`.

OUT-OF-PLANE THICKNESS (2D, 1D AXISYMMETRIC, AND 1D)

- For 2D components, enter a **Thickness** d (SI unit: m). The default is 1 m.
- For 1D axisymmetric components, enter a **Vertical height** d_z (SI unit: m). The default is 0.01 m.
- For 1D components, enter a **Cross-section area** A (SI unit: m^2). The default is 0.01 m^2.

TRANSPORT SETTINGS

Select a **Diffusion model** — **Mixture-averaged** or **Fick's law**. When using the **Mixture-averaged** model, the mixture averaged diffusion coefficients are automatically computed based on the data specified for each species.

Select the check boxes for which transport mechanisms to **Include** — **Convection**, **Migration in electric field**, **Calculate thermodynamic properties**, **Full expression for diffusivity**, or **Compute tensor ion transport properties**. The selection changes the number of Model Inputs requiring values on the **Plasma Model** page. Note the following:

- The **Migration** term is part of the relative mass flux vector.
- For **Calculate thermodynamic properties** select that the thermodynamic properties of each reaction and species are computed automatically based on the thermodynamic properties of each species.
- For **Full expression for diffusivity** it computes a more accurate expression for the Maxwell–Stefan diffusivities. Often the additional correction terms are negligible in which case the expressions are much simpler and the time taken to assemble the Jacobian matrix is reduced.
- For **Mixture diffusion correction** additional terms are included in the definition of the mass flux vector to ensure that the same solution is obtained regardless of the choice of the species which comes from the mass constraint. This option makes the problem more nonlinear and strongly coupled, and is only necessary when the molecular weights of the species differ substantially (such as a mixture of sulfur hexafluoride and hydrogen).
- For **Compute tensor ion transport properties** the tensor form of the ion transport properties when a static magnetic field is present is computed. This option only needs to be activated when a strong DC magnetic field exists and the operating pressure is very low (on the order of millitorr). When this option is activated an expression must be provided for the magnetic flux density, which typically another physics interface computes. This is set in the **Plasma Model** node.

PLASMA PROPERTIES

Select the **Compute tensor electron transport properties** or **Use reduced electron transport properties**, or **Include thermal diffusion** check boxes as needed.

Compute Tensor Electron Transport Properties

Select **Compute tensor electron transport properties** to compute the tensor form of the electron mobility, electron diffusivity, energy mobility and energy diffusivity. This should only be used in cases where a strong DC magnetic field exists. Two quantities must be supplied, both of which are in the Plasma Model node. The DC mobility which is the value of the electron mobility in the absence of a DC magnetic field and the magnetic flux density which would typically be computed by another physics interface.

Use Reduced Electron Transport Properties

Select **Use reduced electron transport properties** to specify the electron mobility, diffusivity, energy mobility and energy diffusivity in reduced form. The neutral number density is then specified in the Drift Diffusion Model node. The electron transport properties are computed from the reduced transport properties using:

$$\mu_e = \frac{\mu_{\mathrm{red}}}{N_n}$$

where N_n is the user-defined neutral number density.

Include Thermal Diffusion

The **Include thermal diffusion** check box adds an additional term to the definition of the electron current due to gradients in the electron diffusivity. If the diffusivity is a constant then including this does not affect the solution. It is only necessary to include this term if the electron diffusivity is a function of the electron temperature, and there are significant gradients in the electron temperature.

For more information on the electron energy distribution function (EEDF), see the Theory for the Boltzmann Equation, Two-Term Approximation Interface.

ELECTRON ENERGY DISTRIBUTION FUNCTION

If cross section data is used to define source coefficients in the model then an electron energy distribution function (EEDF) needs to be selected. Select one of these options.

- **Maxwellian**. This option assumes a Maxwellian EEDF which takes the form:

$$f(\varepsilon) = \varphi^{-3/2}\beta_1 \exp(-(\varepsilon\beta_2/\varphi))$$

where

$$\beta_1 = \Gamma(5/2)^{3/2}\Gamma(3/2)^{-5/2}, \beta_2 = \Gamma(5/2)\Gamma(3/2)^{-1}$$

where φ is the mean electron energy (SI unit: eV), ε is the electron energy (SI unit: eV) and Γ is the incomplete gamma function:

$$\Gamma(s) = \int_0^\infty u^{s-1}e^{-u}du\,.$$

- **Druyvesteyn** (the default). This option assumes a Druyvesteyn EEDF which takes the form:

$$f(\varepsilon) = 2\varphi^{-3/2}\beta_1\exp(-(\varepsilon\beta_2/\varphi)^2)$$

where

$$\beta_1 = \Gamma(5/4)^{3/2}\Gamma(3/4)^{-5/2},\ \beta_2 = \Gamma(5/4)\Gamma(3/4)^{-1}$$

- **Generalized**. Use this option for a generalized distribution function where the EEDF is somewhere between Maxwellian and Druyvesteyn. For this option, specify a power law. This number is typically between 1 and 2. Mathematically, the EEDF takes the form:

$$f(\varepsilon) = g\varphi^{-3/2}\beta_1\exp(-(\varepsilon\beta_2/\varphi)^g)$$

  where

$$\beta_1 = \Gamma(5/2g)^{3/2}\Gamma(3/2g)^{-5/2},\ \beta_2 = \Gamma(5/2g)\Gamma(3/2g)^{-1}$$

- **Function**. If a two-dimensional interpolation function has been added to the model, it can be used for the EEDF. In this case, the *x*-data should be the electron energy (eV) and the *y*-data should be the mean electron energy (eV).

The two-dimensional interpolation function can be computed using a parametric sweep in The Boltzmann Equation, Two-Term Approximation Interface. This allows for modeling of discharges where the EEDF is far from Maxwellian. For step-by-step instructions on how to do this, refer to this blog entry: https://www.comsol.com/blogs/the-boltzmann-equation-two-term-approximation-interface/

See Interpolation in the *COMSOL Multiphysics Reference Manual*.

In all these cases the rate constants in the model are automatically computed based on the selected EEDF using the formula:

$$k_{\mathrm{f}} = \gamma\int_0^\infty \varepsilon\sigma_k(\varepsilon)f(\varepsilon)d\varepsilon\,. \qquad (6\text{-}12)$$

The rate coefficients when computed using cross section data are a highly nonlinear function of the mean electron energy. COMSOL Multiphysics automatically computes the integral in Equation 6-1 and makes the result available for evaluation of the rate coefficient. The variation of the rate coefficient for any particular model can be plotted using `<name>.kf_<reaction number>`. For example, for reaction number 3 in the Plasma interface, with name `plas`, the rate coefficient is plotted using `plas.kf_3`.

EXTRA DIMENSION SETTINGS

Enter the **Period** P_{xd} (SI unit: s) over which the extra dimension should be constructed. The default is `1/(13.56[MHz])`, which corresponds to the time for 1 RF period at the classical operating frequency of 13.56 MHz. This corresponds to the time, in seconds, of the lower excitation frequency in the system. Some care must be taken when specifying this value if there are multiple frequencies included in the model, see Using Consistent Source Frequencies and Period Settings for more details.

Enter the **Number of elements** over which to discretize the extra dimension. The default value of 40 means that 40 mesh elements will be used to discretize the RF period in the extra dimension. In most cases, 40 should be enough for a single frequency. As for the **Period** setting, using more than one frequency typically means the **Number of elements** must be increased.

The remaining settings describe in which space the nonelectron should be computed. See Choosing the Correct Space on Which to Solve for the Heavy Species for more details. For the **Heavy species selection**, there are two options available:

- The **Product space** option means that all the nonelectron species in the model will be computed in the product space of the base geometry and the extra dimension. This option can generate a significant number of degrees of freedom and is better suited to 1D models.
- For 2D models, it is usually better to solve for the heavy species in the **Base geometry**. This keeps the number of degrees of freedom to a minimum, while sacrificing little in terms of accuracy.

If the **Heavy species selection** is set to **Product space**, there is an additional option for the **Electric field applied to ions**. This can be either **Instantaneous**, in which case the time modulating electric field is used to compute the ion migrative fluxes, or **Time averaged**, in which case the ions experience the period averaged electric field. Using the **Time**

averaged option is very computationally expensive, and is usually only used for testing simple 1D models.

SECONDARY EMISSION MODEL

To display this section, click the **Show More Options** button (👁) and select **Advanced Physics Options**.

By default the **Secondary emission model** is set to **From surfaces**. With this option the gain of electrons due to secondary emission is made through the normal component of the electron flux at the wall. See The Wall Boundary Condition for more details. If the **Secondary emission model** is set to **Uniform** the gain of electrons from secondary emission is distributed uniformly across the domain through the electron rate expression. This source of electrons is computed from

$$R_{e,\,sec} = N_{i,\,\mathrm{sec}}\frac{1}{V}\int_{\partial\Omega}\Gamma_{i,\,\mathrm{sec}}dS \tag{6-13}$$

where the number of ionization events per secondary electron emitted is obtained from

$$N_{i,\,\mathrm{sec}} = g\sum_{k} n_k\sigma_k(\varepsilon_b) \tag{6-14}$$

where the sum is over all ionization reactions by electron impact, g is the **Characteristic gap size**, n_k is the target number density, σ_k is the reaction cross section, and ε_{b} is the **Beam energy**. In Equation 6-13 the integrand is the period-averaged secondary emission flux from all possible surface reactions, the integral is over all surfaces, and V is the reactor volume. The **Uniform** model is an extremely simplistic model that tries to capture the nonlocal behavior of ionization events produced by secondary electrons that are accelerated to very high energies. This phenomenon is important at low pressures below the 10 s of millitorr range. See Validity of Fluid Models for more detail on the regime of validity.

STABILIZATION

To display this section, click the **Show More Options** button (👁) and select **Stabilization**.

The solver can run into difficulties as the species mass fractions approach zero. The **Reaction source stabilization** check box (off by default) adds an additional source term to the rate expression for each species. In the ι field, enter a tuning parameter for the source stabilization. By default, no stabilization is added. However, if working with a model with no secondary electron emission included, it may be necessary to activate this. A value of 1 or higher is recommended, although if the plasma is high pressure (atmospheric) then it can help to lower this number to somewhere in the range of 0.25–0.5.

The solver can also run into difficulties as the electron density or electron energy density approach zero. The **Source stabilization** check box (off by default) adds an additional source term to the equation for the electron density and electron energy density. The same values and guidelines as above apply.

INCONSISTENT STABILIZATION

To enable this section, click the **Show More Options** button (👁) and select **Stabilization** in the **Show More Options** dialog box. This section is only available if the **Heavy species selection** in **Extra Dimension Settings** is set to **Base geometry**.

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

If required, select the **Isotropic diffusion for ions** check boxes and enter a **Tuning parameter for ions** $\delta_{\mathrm{id,i}}$. The default value is 0.25.

DISCRETIZATION

Select **Formulation** — **Finite element, log formulation (linear shape function)** (the default), or **Finite element, log formulation (quadratic shape function)**.

DEPENDENT VARIABLES

The dependent variables (field variables) are the **Electron solution variable**, and **Electron energy solution variable**, **Electric potential**. The name can be changed but the names of fields and dependent variables must be unique within a model. The dependents variables that by default end with _per are only used to map the time periodic solution into a time-dependent solution using the Time Periodic to Time Dependent study.

- Domain, Boundary, Edge, Point, and Pair Nodes for the Plasma Interface
- Plasma Reactors Theory and Capacitively Coupled Plasmas
- Stabilization
- Interpolation in the *COMSOL Multiphysics Reference Manual*

- *Alpha to Gamma Transition*: Application Library path **Plasma_Module/Capacitively_Coupled_Plasmas/alpha_to_gamma_transition**
- *Ion Energy Distribution Function in a Capacitively Coupled Plasma Reactor*: Application Library path **Plasma_Module/Capacitively_Coupled_Plasmas/ccp_ion_energy_distribution_function**
- *Computing the Plasma Impedance*: Application Library path **Plasma_Module/Capacitively_Coupled_Plasmas/computing_plasma_impedance**

Domain, Boundary, Edge, and Point for the Plasma, Time Periodic Interface

The Plasma, Time Periodic Interface has these domain, boundary, edge, and point nodes, listed in alphabetical order, available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or right-click to access the context menu (all users).

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

In the *COMSOL Multiphysics Reference Manual* see Table 2-4 for links to common sections and Table 2-5 to common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

These nodes are described in the The Plasma Interface chapter:

- Plasma Model
- Initial Values
- Species Group
- Species Constraint Group

These nodes are described in the chapters for The Drift Diffusion Interface and The Heavy Species Transport Interface (listed in alphabetical order):

- Cross Section Import
- Electron Impact Reaction
- General Power Deposition
- Reaction
- Species
- Surface Reaction
- Wall

These nodes are available to describe how to excite and ground the plasma:

- Dielectric Contact
- Ground
- Metal Contact
- Surface Charge Accumulation
- Terminal
- Zero Charge

The following node is available as a subnode for the Metal Contact and Terminal nodes:

- Periodic Excitation

The following nodes allow to set inflow and outflow boundary conditions:

- Inflow
- Outflow

Inflow

The **Inflow** feature sets the mass fraction of specified heavy species at a boundary. Use this feature to set the composition of gas mixture going into a system. This feature does not set the fluid velocity at a boundary. For that, the plasma model needs to coupled with a fluid flow interface.

When solving for plasmas with chemistries that contain more than one element (for example, Ar and O_2) in the Plasma, Time Periodic interface the mass fraction of each element is not conserved if no constraint is used. This problem is similar in nature to the one found when solving for Navier–Stokes equations in steady state without fixing the pressure somewhere. The **Inflow** boundary condition fixes the mass fraction or mole fraction of specified species and it can be used as a strategy to constrain the mass fraction of some species at a given boundary even if no fluid flow exists in the system. An important aspect to remember is that the **Inflow** feature does not apply a constraint to the species being computed **From mass constraint** and if some other species mass fractions attain important values the mass fraction of the mass constraint species can differ from the value specified. This can be avoided by setting to a small value the mass fraction of the species that attain large values at the inflow boundary.

In the **Mixture specification**: select what quantity to constrain:

- **Mole fraction** (the default) constrains the mole fraction
- **Mass fraction** constrains the mole fraction

On the table specify the names of the species to constrain and the value of the mole or mass fraction. As an example, if the inflow mixture of Ar/Cl2 has a mole fraction of 0.1/0.9 the table should be like:

TABLE 6-3: MOLE FRACTION AT THE INFLOW BOUNDARY.

Species name	Mole fraction (1)
Ar	0.1
Cl2	0.9

Note that the names must correspond to the exact name in the Species name field in the Species feature.

Assume that there is significant dissociation of Cl2 so that Cl is an important species. Also assume that Ar uses the option **From mass constraint**. In this case the mass fraction for Ar is not solved for and there is no constraint applied to set the mass fraction at 0.1. This means that only the mass fraction of Cl2 is constrained at the Inflow boundary. Since the Cl mass fraction is high everywhere the mass fraction of Ar will not be 0.1 at

Inflow boundary. This could be avoided by extending the inflow channel so that there is no plasma and no dissociation in the region where the Inflow boundary is applied. Another solution for this issue is to constraint the mole fraction of Cl to a small value as shown below.

TABLE 6-4: MOLE FRACTION AT THE INFLOW BOUNDARY.

Species name	Mole fraction (1)
Ar	0.1
Cl2	0.9
Cl	1e-4

Outflow

The Outflow node should be applied at outlets where species are transported out of the model domain. This boundary condition makes sure that transport losses by diffusion do not exist but transport losses by convection, migration in the electric field, and others can still exist. This node is available for exterior boundaries. By default the boundary condition is applied to all neutral heavy species. By selecting the option **Include ions** the boundary conditions is also applied to all ions. The following boundary condition is applied depending on the selection made on the **Diffusion Model** section:

- **Mixture-averaged**

$$-\mathbf{n}\cdot\rho D_{k,m}\nabla w_k = 0$$

- **Fick's law**

$$-\mathbf{n}\cdot\rho D_f\nabla w_k = 0$$

Species Group

When a chemical mechanism contains species made up of different elements, the default mass constraint (the sum of all mass fractions equals one) might not be good enough to ensure mass conservation over all elements. The **Species Group** feature can be used to ensure that the total elemental mass in the system is conserved. Consider a simplified $\mathrm{Ar/O_2}$ plasma chemistry with an initial 70/30 ratio by mole fraction. For a closed system (which is currently the only type of system currently supported in the **Plasma, Time Periodic** interface), the sum of the mass fractions of oxygen species should equal the initial value, and the sum of the mass fractions of argon species should equal their initial values. If the following species are considered in the model: `Ar`, `Ars`, `Ar_1p`, O2, O, O2_1p, then ground state argon should be picked as the **From mass constraint** species, and a **Species Group** feature should be added with the following table:

TABLE 6-5: SPECIES GROUP TABLE FOR OXYGEN.

O2
O
O2_1p

Note that the first species in the table matters. The first species in the table is used to enforce mass conservation for this group of species, meaning that the total number of oxygen atoms in the system is conserved. The first species should correspond to a ground state species present in excess within this group. If additional species N2, N2A, N2_1p were added to the model, an additional **Species Group** feature would be necessary:

TABLE 6-6: SPECIES GROUP TABLE FOR NITROGEN.

N2
N2A
N2_1p

Species Constraint Group

For models with complex chemistries, obtaining a converged solution can be tricky, or not possible at all. The **Species Constraint Group** feature can be used in the following way to help in these cases:

- Add the species suspected to be causing nonconvergence to the **Species Constraint Group** feature. Instead of solving for these species, the mass fraction will be constrained to the initial value.
- Add a study and compute the solution with the troublesome species constrained to their initial values.
- Add a second study which uses the solution from the first study as the initial value. This is done by using the options in **Values of Dependent Variables** section of the study settings. In the same study settings, the Species Constraint Group feature can be disabled using the **Modify model configuration for study step** option.

This means that the troublesome species will begin to evolve based on a physically correct, converged initial solution, rather than from a “cold start”.

In the **Species Group** section, add potentially troublesome species to the **Species names** table. Note that the names must correspond to the exact name in the Species name field in the Species feature. For example, an argon ion should be entered into the table as `Ar_1p` and not `Ar+`.

Dielectric Contact

Use the **Dielectric Contact** node is used for modeling what happens when a plasma is in contact with a dielectric (insulating) surface. The **Dielectric Contact** can only be applied on boundaries adjacent to the **Plasma Model** feature.

Ground

The **Ground** node implements ground (zero potential) as the boundary condition $V = 0$. Ground means that there is a zero potential on the boundary at all points in the base geometry and the extra dimension. This boundary condition is also applicable at symmetry boundaries where the potential is known to be antisymmetric with respect to the boundary.

Metal Contact

The **Metal Contact** is applied to exterior boundaries which are adjacent to the **Plasma Model** feature. This corresponds to a powered electrode, where the source voltage, power, or external circuit can be specified. The following options are available:

- For the **Terminal type**, choose either **Voltage**, **Power**, or **Circuit**. When **Circuit** is selected, refer to the Circuit Options section. In nearly all cases, the **Power** option is a more stable way of driving the plasma, as multiple physically significant solutions can exist when a voltage source is used. When driving with a fixed power, there can only be one solution to the problem.

For the **Source**, select one of the following options:

- **RF**. In this case, the excitation is periodic and the maximum and minimum amplitude of the applied voltage are equal. The following options are available.
  - Select whether the **Periodic function** should be **Cosine** or **Sine**. This simply determines whether the excitation waveform should form a cosine or sine wave over the period.
  - If the **Terminal type** is **Voltage**, enter the **Voltage amplitude** V_a (SI unit: V) at the electrode. If the **Terminal type** is **Power**, enter the **RF Power** P_{rf} (SI unit: W) to be deposited into the plasma.
  - Enter the **Frequency** f_p (SI unit: Hz) for the applied voltage or power. This value must be consistent with the value given for the **Period** property of the physics interface. See Using Consistent Source Frequencies and Period Settings for further information.
  - Enter the **Source phase** α (SI unit: dimensionless) of the generator. The default is zero and this rarely needs to be changed.
- **RF and DC**. This option allows an additional bias power or voltage to be applied, or the DC self–bias to be computed. In addition to the options for the **RF** case above, the following options are available:
  - **Compute DC self–bias**. In asymmetric discharges, where the grounded surface area of the discharge is different from the powered electrode surface area, the powered electrode will form a negative DC self–bias. This requires that the period averaged current arriving on the electrode surface is zero. By checking this option, the software will automatically compute the DC self–bias on the electrode. When this option is selected, no other options in the **DC Source** section are available.
  - When the **Terminal type** is **Voltage**, and **Compute DC self–bias** is off, enter a **DC bias voltage** $V_{\mathrm{dc,b}}$ (SI unit: V). This can be used to increase the ion or electron fluxes to the electrode which may result in a net current flow into the electrode. When the **Terminal type** is **Power**, and **Compute DC self–bias** is off, enter a **DC bias power** $P_{\mathrm{dc,b}}$ (SI unit: W). There is an additional option to specify whether the **Bias** should be **Negative** or **Positive**.
- **DC**. This option can be used to add a pure DC bias somewhere in the system, but there has to be at least one feature adding an RF excitation elsewhere. Adding a pure

DC source generally makes it more difficult to obtain a converged solution. The following options are available:

- When the **Terminal type** is **Voltage**, enter a **DC voltage** V_{dc} (SI unit: V). When the **Terminal type** is **Power**, enter a **DC power** P_{dc} (SI unit: W). There is an additional option to specify whether the **Electrode type** should be **Cathode** or **Anode**, depending on whether the power should be deposited using a negative or positive potential.

Surface Charge Accumulation

On interior boundaries which are adjacent to both a **Plasma Model** and **Charge Conservation** feature, the **Surface Charge Accumulation** node is used to determine the accumulation of surface charge on the dielectric surface due to incident ion and electron fluxes. This condition should always be applied on boundaries which are adjacent to both a **Plasma Model** and **Charge Conservation** feature, otherwise the charged species will “leak” from the modeling domain, resulting in unphysical results. Mathematically, the following boundary condition is implemented:

$$-\mathbf{n}\cdot(\mathbf{D}_1-\mathbf{D}_2) = \rho_s \tag{6-15}$$

where ρ_s is the solution of the following distributed ODE on the boundary:

$$\frac{d\rho_s}{dt} = \mathbf{n}\cdot\mathbf{J}_i+\mathbf{n}\cdot\mathbf{J}_e \tag{6-16}$$

where $\mathbf{n}\cdot\mathbf{J}_i$ is the normal component of the total ion current density on the wall and $\mathbf{n}\cdot\mathbf{J}_e$ is the normal component of the total electron current density on the wall. In addition, since the solution is assumed to be time periodic, a periodic boundary condition on the surface charge density is imposed in the extra dimension.

Terminal

The **Terminal** is applied to exterior boundaries which are adjacent to the **Charge Conservation** feature. This corresponds to a powered electrode in contact with a dielectric material, where the source voltage, power, or external circuit can be specified. This condition cannot be applied on boundaries adjacent to the **Plasma Model** feature. The following options are available:

- For the **Terminal type**, choose either **Voltage**, **Power**, or **Circuit**. When **Circuit** is selected, refer to the Circuit Options section. In nearly all cases, the **Power** option is a more stable way of driving the plasma, as multiple physically significant solutions can exist when a voltage source is used. When driving with a fixed power, there can only be one solution to the problem.

For the **Source**, select one of the following options:

- **RF**. In this case, the excitation is periodic and the maximum and minimum amplitude of the applied voltage are equal. The following options are available.
  - Select whether the **Periodic function** should be **Cosine** or **Sine**. This simply determines whether the excitation waveform should form a cosine or sine wave over the period.
  - If the **Terminal type** is **Voltage**, enter the **Voltage amplitude** V_a (SI unit: V) at the electrode. If the **Terminal type** is **Power**, enter the **RF Power** P_{rf} (SI unit: W) to be deposited into the plasma.
  - Enter the **Frequency** f_p (SI unit: Hz) for the applied voltage or power. This value must be consistent with the value given for the **Period** property of the physics interface. See Using Consistent Source Frequencies and Period Settings for further information.
  - Enter the **Source phase** α (SI unit: dimensionless) of the generator. The default is zero and this rarely needs to be changed.

Zero Charge

The Zero Charge node adds the condition that there is zero charge on the boundary so that $\mathbf{n} \cdot \mathbf{D} = 0$. This boundary condition is also applicable at symmetry boundaries where the potential is known to be symmetric with respect to the boundary. This is the default boundary condition at exterior boundaries. At interior boundaries, it means that no displacement field can penetrate the boundary and that the electric potential is discontinuous across the boundary.

Periodic Excitation

The **Periodic Excitation** node can be added as a subfeature to either the **Metal Contact** or **Terminal** features. This allows for dual frequency reactors to be modeled, although certain restrictions apply, see Using Consistent Source Frequencies and Period Settings.

The **Terminal type** setting should be the same as the parent feature (the **Metal Contact** or **Terminal**). If the **Terminal type** differs, a warning node appears in the Model Builder. The following options are available for all terminal types:

- Enter the **Frequency** f_p (SI unit: Hz) for the applied voltage or power. This value must be consistent with the value given for the **Period** property of the physics interface. See Using Consistent Source Frequencies and Period Settings for further information.
- Enter the **Source phase** α (dimensionless) of the generator. The default is zero and this rarely needs to be changed.

Depending on the **Terminal type**, the following is available:

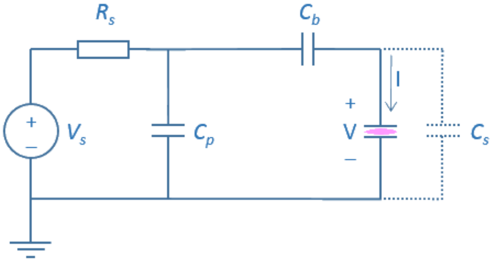
- When the **Terminal type** is **Voltage** or **Power**, select whether the **Periodic function** should be **Cosine** or **Sine**. This simply determines whether the excitation waveform should form a cosine or sine wave over the period.
- Enter values for either the **Voltage amplitude** V_a or V_{cir} (SI unit: V) or **RF Power** P_{rf} (SI unit: W) depending on the value selected for the **Terminal type**.

Circuit Options

The **Metal Contact** and **Terminal** features both allow the **Terminal type** to be **Circuit**. In this case, there are 3 options available, **Parallel RC circuit**, **L–network**, and **Reversed L–network**.

PARALLEL RC CIRCUIT

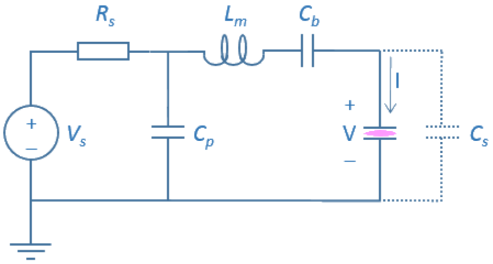
The circuit diagram for the most general parallel RC circuit is shown below:



The circuit includes an optional blocking capacitor (C_b) and stray capacitance (C_s). Both are off by default, and both make the model more nonlinear and thus more difficult to solve when active. The parallel RC circuit is often used to perform some initial investigations before moving onto designing a matching network.

L–NETWORK

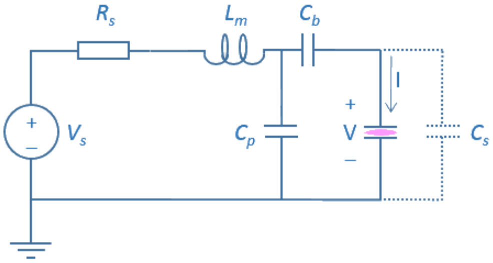
The L–network is a matching circuit designed to ensure maximum power transfer between the RF generator and the plasma. The most general form of the circuit is shown below:



As with the parallel RC circuit, the blocking capacitor and stray capacitance are optional, and off by default. The match inductance (L_m) and capacitance (C_p) can be adjusted to maximize the power transfer for a known plasma impedance. This type of match is only appropriate when the real part of the plasma impedance is below 50 Ω, which typically occurs when the discharge power is high.

REVERSED L–NETWORK

The reversed L-network is similar to the L-network, except the parallel capacitor is moved to the other side of the inductor:



As with the parallel RC circuit, the blocking capacitor and stray capacitance are optional, and off by default. This type of match is only appropriate when the real part of the plasma impedance is above 50 Ω, which typically occurs when the discharge power is low.

Enter the following settings:

- **Source type**. This determines whether the generator acts like a **Voltage source** (the default) or **Power source**. When **Voltage source** is selected:
  - Enter the **Voltage amplitude** V_{cir} (SI unit: V) of the generator.

  When **Power source** is selected:
  - Enter the **Source power** P_s (SI unit: W) of the generator.

  Using the **Power source** option is usually more stable, meaning that its more likely that the solver will converge than the **Voltage source** option.
- Enter the **Series resistance** R_s (SI unit: Ω) of the generator. This corresponds to the impedance the generator and is 50 Ω (the default value) in most practical applications.
- Enter the **Parallel capacitance** C_p (SI unit: F) which corresponds to the capacitance parallel to the plasma. When the **Circuit type** is **L–network** or **Reversed L–network**, this value can be computed to ensure proper matching.

Computing the Plasma Impedance: Application Library path **Plasma_Module/Capacitively_Coupled_Plasmas/computing_plasma_impedance**

- When the **Circuit type** is **L–network** or **Reversed L–network**, enter the **Series inductance** (SI unit: H). This value can also be computed to obtain a proper match.
- Enter the **Source frequency** f_p (SI unit: Hz) of the generator. This value must be consistent with the value given for the **Period** property of the physics interface. See Using Consistent Source Frequencies and Period Settings for further information.
- Enter the **Source phase** α (dimensionless) of the generator. The default is zero and this rarely needs to be changed.

In the Additional Circuit Elements section, there are the following options:

- **Include blocking capacitor**. When this is active, enter the **Blocking capacitance** C_b (SI unit: F). This places a blocking capacitor between the plasma and the external circuit which will shield any accumulated surface charges on the electrode.
- **Include stray capacitance**. When this is active, enter the **Stray capacitance** C_s (SI unit: F). This adds an additional parallel capacitance across the plasma corresponding to stray leads or the vacuum capacitance of the system. The presence of a stray

capacitance can lead to an impedance mismatch between the generator and plasma in a system which would otherwise be perfectly matched.

The Inductively Coupled Plasma Interface

The **Inductively Coupled Plasma** interface is used to study discharges that are sustained by induction currents. The induction currents are solved for in the frequency domain. The electron heating due to the induction currents is automatically handled by the software. This physics interface is available for 2D and 2D axisymmetric components.

With an AC/DC Module license, this physics interface is also available in 3D. This also enables additional features that are available with the AC/DC Module, and described in the *AC/DC Module User's Guide*.

When a predefined **Inductively Coupled Plasma** interface ( ) is added from the **Plasma** branch ( ) of the **Model Wizard** or **Add Physics** windows, **Plasma** and **Magnetic Fields** interfaces are added to the Model Builder. In addition, the **Multiphysics Couplings** node is added, which automatically includes the multiphysics coupling features **Plasma Conductivity Coupling** and **Electron Heat Source**.

When the **Plasma** physics interface is added, these default nodes are also added to the **Model Builder** — **Plasma Model**, **Zero Charge, Insulation**, and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, the plasma chemistry and boundary conditions.

When the **Magnetic Fields** interface is added, these default nodes are also added to the **Model Builder** — **Ampère's Law, Magnetic Insulation**, and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, a coil.

SETTINGS

The settings for the **Plasma** interface can be found in The Plasma Interface section. The settings for the **Magnetic Fields** interface can be found in The Magnetic Fields Interface in the *AC/DC Module User's Guide*.

For a description of the multiphysics coupling nodes, refer to:

- Plasma Conductivity Coupling
- Electron Heat Source
- Collisionless Heating Coupling

The Inductively Coupled Plasma with RF Bias Interface

The **Inductively Coupled Plasma with RF Bias** interface is used to study discharges that are sustained by induction currents and have an additional time periodic electric excitation. The magnetic field is solved in the frequency domain and the plasma solves for a periodic steady-state. Multiphysics couplings couple the time averaged plasma conductivity to the **Magnetic Fields** interface and couple the resulting electron heating due to the induction currents back to the **Plasma, Time Periodic** interface. This physics interface is available for 2D and 2D axisymmetric components.

A AC/DC Module license is necessary to have access to the **Coil** feature. This also enables additional features that are available with the AC/DC Module, and described in the *AC/DC Module User's Guide*.

When a predefined **Inductively Coupled Plasma with RF Bias** interface ( ) is added from the **Plasma** branch ( ) of the **Model Wizard** or **Add Physics** windows, **Plasma, Time Periodic** and **Magnetic Fields** interfaces are added to the Model Builder. In addition, the **Multiphysics Couplings** node is added, which automatically includes the multiphysics coupling features **Plasma Conductivity Coupling** and **Electron Heat Source**.

When the **Plasma, Time Periodic** physics interface is added, these default nodes are also added to the **Model Builder** — **Plasma Model**, **Zero Charge, Insulation**, and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, the plasma chemistry and boundary conditions.

When the **Magnetic Fields** interface is added, these default nodes are also added to the **Model Builder** — **Ampère's Law, Magnetic Insulation**, and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, a coil.

SETTINGS

The settings for the **Plasma, Time Periodic** interface can be found in The Plasma, Time Periodic Interface section. The settings for the **Magnetic Fields** interface can be found in The Magnetic Fields Interface in the *AC/DC Module User's Guide*.

For a description of the multiphysics coupling nodes, refer to:

- Plasma Conductivity Coupling

- Electron Heat Source

The Microwave Plasma Interface

!

This physics interface requires a RF Module license. For theory and physics interface feature descriptions, see the *RF Module User's Guide*.

The **Microwave Plasma** physics interface is suitable for studying discharges that are sustained by electromagnetic waves (wave-heated discharge). The electromagnetic waves are solved for in the frequency domain, and all plasma variables are solved for in the time-domain.

When a predefined **Microwave Plasma** interface ( ) is added from the **Plasma** branch ( ) of the **Model Wizard** or **Add Physics** windows, **Plasma** and **Electromagnetic Waves, Frequency Domain** interfaces are added to the Model Builder. In addition, the **Multiphysics Couplings** node is added, which automatically includes the multiphysics coupling features **Plasma Conductivity Coupling** and **Electron Heat Source**.

When the **Plasma** physics interface is added, these default nodes are also added to the **Model Builder**: **Plasma Model**, **Zero Charge**, **Insulation**, and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, the plasma chemistry and boundary conditions.

When the **Electromagnetic Waves, Frequency Domain** interface is added, these default nodes are also added to the **Model Builder** — **Wave Equation, Electric**; **Perfect Electric Conductor**; and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, a port.

SETTINGS

The settings for the **Plasma** interface can be found in The Plasma Interface section. The settings for the **Electromagnetic Waves, Frequency Domain** interface can be found in The Electromagnetic Waves, Frequency Domain Interface in the *RF Module User's Guide*.

For a description of the multiphysics coupling nodes, refer to:

- Plasma Conductivity Coupling
- Electron Heat Source

Multiphysics Couplings

Plasma Conductivity Coupling

The **Plasma Conductivity Coupling** multiphysics coupling node ( ) computes the electrical conductivity of the plasma based on the cold plasma approximation. The electron density is provided by the **Plasma** or **Plasma, Time Periodic** interfaces and the and collision frequency can be provided by the **Plasma** or **Plasma, Time Periodic** interfaces or set by the user. The angular frequency is provided by the **Magnetic Fields** or **Electromagnetic Waves, Frequency Domain** interfaces.

COLLISION FREQUENCY SPECIFICATION

Select how to set the collision frequency from **Specify collision frequency using** — **User defined**, **Electron mobility** (the default), or **Electron energy distribution function**. The option **Electron energy distribution function** only exists for the **Plasma** interface.

- When **User defined** is selected enter values for the **Effective collision frequency** and the **Field coefficient**. The plasma conductivity is computed using these values and the electron density provided by the **Plasma** or **Plasma, Time Periodic** interfaces. For the **Plasma, Time Periodic** interface the electron density used in the plasma conductivity is period averaged.
- When **Electron mobility** is selected the momentum collision frequency used to computed the plasma conductivity is obtain from the mobility defined in the **Plasma** interface using

$$\nu_m = \frac{e}{m_e\mu_e} \tag{6-17}$$

and for the **Plasma, Time Periodic** interface using

$$\nu_m = \frac{e}{m_e\langle\mu_e\rangle} \tag{6-18}$$

where $\langle\mu_e\rangle$ is the period averaged electron mobility.

- The option **Electron energy distribution function** is only available when the options **two-term approximation (linear)**, or **Boltzmann equation, two-term approximation (quadratic)** are selected from the property **Electron energy distribution function**. When using this option the effective collision frequency ν_{eff} and the field coefficient ϕ are automatically computed using:

$$\nu_{\text{eff}} = -\frac{\mu_r}{\mu_i}\omega \tag{6-19}$$

$$\phi = -\frac{\mu_r^2 + \mu_i^2}{\mu_i}\frac{m_e\omega}{e} \tag{6-20}$$

where μ_r and μ_i are the real and imaginary parts of the mobility defined in Boltzmann Equation, Two-Term Approximation.

COMPUTE TENSOR PLASMA CONDUCTIVITY

If a static magnetic field is present, and the pressure is suitably low, it may be necessary to use a full tensor for the plasma conductivity. This can result is a nonlinear and anisotropic plasma conductivity, as is the case in electron cyclotron resonance. When **Compute tensor plasma conductivity** is active, enter a value for the **Doppler broadening parameter**. This can smooth resonance zones where the power deposition is focused over a very small region. The lower the value entered, the more smoothing occurs. More information can be found in the Resonance Zone section.

This option is not available for the **Plasma, Time Periodic** interface.

COUPLED INTERFACES

The selection for this multiphysics coupling is locked, and corresponds to the intersection of the **Magnetic Fields** or **Electromagnetic Waves, Frequency Domain** and **Plasma** or **Plasma, Time Periodic** interfaces specified in the **Coupled Interfaces** section.

Electron Heat Source

The **Electron Heat Source** multiphysics coupling node ( ) applies the heat source from the **Magnetic Fields** or **Electromagnetic Waves, Frequency Domain** to the electrons in the **Plasma** or **Plasma, Time Periodic** interfaces. This corresponds to inductive heating or wave-heating of the electrons respectively.

When coupling to the **Plasma, Time Periodic** interface the electron heating is assumed constant along the period defined in the extra dimension.

COUPLED INTERFACES

The selection for this multiphysics coupling node is locked, and corresponds to the intersection of the **Magnetic Fields** or **Electromagnetic Waves, Frequency Domain** and **Plasma** or **Plasma, Time Periodic** interfaces specified in the **Coupled Interfaces** section.

Collisionless Heating Coupling

The **Collisionless Heating Coupling** multiphysics coupling node () includes nonlocal heating effects via an effective viscosity approximation. Typically this is only important at very low pressures (less than 2 Pa) and frequencies above 10 MHz. This multiphysics coupling node is only available in 2D axisymmetric models.

COUPLED INTERFACES

The selection for this multiphysics coupling is locked, and corresponds to the intersection of the **Magnetic Fields** and **Plasma** interfaces specified in the **Coupled Interfaces** section.

Plasma Reactors Theory

The Plasma interface couples the Drift Diffusion, Heavy Species and Electrostatic interfaces. A brief explanation of the underlying equations is given here.

For detailed information about electron transport see Theory for the Drift Diffusion Interface.

For detailed information about the transport of the nonelectron species see Theory for the Heavy Species Transport Interface.

For detailed information about electrostatics see The Electrostatics Interface.

The electron density and mean electron energy are computed by solving a pair of drift-diffusion equations for the electron density and mean electron energy. Convection of electrons due to fluid motion is neglected.

$$\frac{\partial}{\partial t}(n_e) + \nabla \cdot [-n_e(\mu_e \bullet \mathbf{E}) - \nabla(\mathbf{D}_e n_e)] = R_e$$

$$\frac{\partial}{\partial t}(n_\varepsilon) + \nabla \cdot [-n_\varepsilon(\mu_\varepsilon \bullet \mathbf{E}) - \nabla(\mathbf{D}_\varepsilon n_\varepsilon)] + \mathbf{E} \cdot \Gamma_e = R_\varepsilon$$

The electron source R_e and the energy loss due to inelastic collisions R_ε are defined later. If a strong DC magnetic field is present then the electron mobility can be a tensor:

$$\mu_e{}^{-\mathbf{1}} = \begin{bmatrix} \frac{1}{\mu_{\mathrm{dc}}} & -B_z & B_y \\ B_z & \frac{1}{\mu_{\mathrm{dc}}} & -B_x \\ -B_y & B_x & \frac{1}{\mu_{\mathrm{dc}}} \end{bmatrix}$$

where the inverse of the mobility has been used because the actual expression for the electron mobility cannot be written in a compact form. The quantity μ_{dc} is the electron mobility in the absence of a magnetic field. The electron diffusivity, energy mobility, and energy diffusivity are then calculated using:

$$\mathbf{D}_e = \mu_e T_e, \mu_\varepsilon = \left(\frac{5}{3}\right)\mu_e, \mathbf{D}_\varepsilon = \mu_\varepsilon T_e$$

The source coefficients in the above equations are determined by the plasma chemistry and are written using either rate or Townsend coefficients. Suppose that there are M reactions that contribute to the growth or decay of electron density and P inelastic electron-neutral collisions. In general $P >> M$. In the case of rate coefficients, the electron source term is given by

$$R_e = \sum_{j=1}^{M} x_j k_j N_n n_e$$

where x_j is the mole fraction of the target species for reaction j, k_j is the rate coefficient for reaction j (SI unit: m^3/s), and N_n is the total neutral number density (SI unit: $1/\mathrm{m}^3$). When Townsend coefficients are used, the source term becomes

$$R_e = \sum_{j=1}^{M} x_j \alpha_j N_n |\Gamma_e|$$

where α_j is the Townsend coefficient for reaction j (SI unit: m^2) and Γ_e is the electron flux (SI unit: $1/(\mathrm{m}^2 \cdot \mathrm{s})$). *Townsend coefficients* can increase the stability of the numerical scheme when the electron flux is field driven as is the case with DC discharges.

The electron energy loss is obtained by summing the collisional energy loss over all reactions:

$$R_\varepsilon = \sum_{j=1}^{P} x_j k_j N_n n_e \Delta\varepsilon_j$$

where $\Delta\varepsilon_j$ is the energy loss from reaction j (SI unit: V). In the case of Townsend coefficients, the energy loss is given by

$$R_\varepsilon = \sum_{j=1}^{P} x_j \alpha_j N_n |\Gamma_e| \Delta\varepsilon_j$$

The electron source and inelastic energy loss are automatically computed by the multiphysics interface. The *rate coefficients* can be computed from cross section data by the following integral:

$$k_k = \gamma\int_0^\infty \varepsilon\sigma_k(\varepsilon)f(\varepsilon)d\varepsilon$$

where:

- $\gamma = (2q/m_e)^{1/2}$ ($\mathrm{C}^{1/2}/\mathrm{kg}^{1/2}$)
- m_e is the electron mass (SI unit: kg)
- ε is the energy (SI unit: V)
- σ_k is the collision cross section (SI unit: m^2), and
- f is the electron energy distribution function. Choose between Maxwellian, Druyvesteyn, or a generalized EEDF.

For nonelectron species, the following equation is solved for the mass fraction of each species.

$$\rho\frac{\partial}{\partial t}(w_k) + \rho(\mathbf{u}\cdot\nabla)w_k = \nabla\cdot\mathbf{j}_k + R_k$$

The electrostatic field is computed using the following equation:

$$-\nabla\cdot\varepsilon_0\varepsilon_\mathrm{r}\nabla V = \rho$$

The *space charge density*, ρ is automatically computed based on the plasma chemistry specified in the model using the formula:

$$\rho = q\left(\sum_{k=1}^{N} Z_k n_k - n_e\right).$$

Capacitively Coupled Plasmas

Capacitively coupled plasmas (CCP) represent a monumental challenge when it comes to numerical modeling for the following reasons. The main challenge is the overwhelming number of RF cycles before the plasma evolves to a time-periodic steady-state solution. The Plasma Module includes a special physics interface for such applications, which removes the need to solve the problem in the time domain, while still capturing all the nonlinearities involved in the discharge. This is the Plasma, Time Periodic interface.

Consider the following test problem, which roughly describes the evolution of a metastable species in a capacitively coupled plasma:

$$\left(\frac{1}{f}\right)\frac{du}{dt} = a\sin^2(\omega t) - bu - cu^2$$

where u is representative of the amount of the metastable species. The first term on the right-hand side represents the periodic production of the metastable in the plasma sheath, oscillating with period ω. The second term represents losses due to collisions with the background gas, and the third term losses due to collisions between the metastables.

The evolution of the metastable density is shown in Figure 6-1. It takes close to 100 cycles before the value evolves to its periodic steady-state solution. In many plasma reactors, it can be more like 50 or 100,000 RF cycles before this evolution is complete. Solving this in the time domain is computationally intractable.

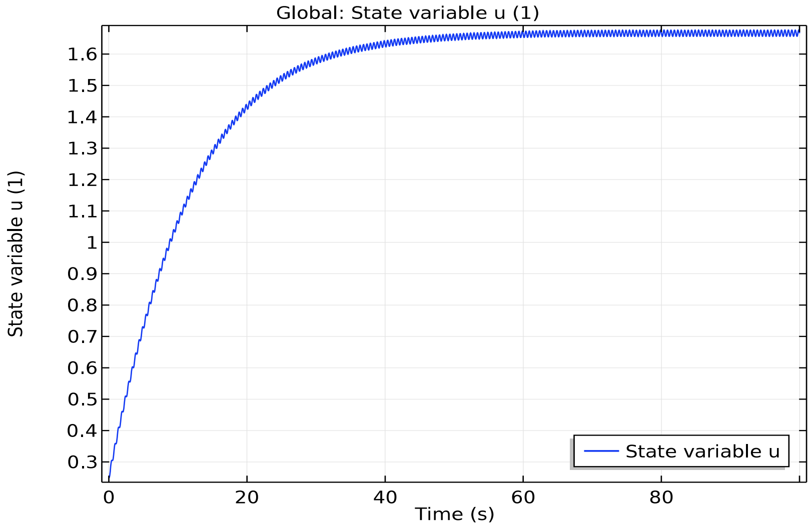


Figure 6-1: Metastable density.

The above equation can be reformulated as a boundary value problem:

$$\left(\frac{1}{f}\right)\frac{du}{dx} = a\sin^2(\omega x) - bu - cu^2$$

with periodic boundary conditions:

$$u(x = 0) = u\left(x = \frac{\pi}{\omega}\right) .$$

Solving this boundary value problem will immediately produce the solution during the period when the periodic steady-state solution is reached. Thus, a time-dependent problem with enormous computational cost is replaced by a simple boundary value problem. This is the basic technique used in the **Plasma, Time Periodic** interface.

In order to facilitate the modeling process, the conversion to a boundary problem is accomplished by attaching an extra dimension to the reactor geometry. This forms a product space where the boundary value problem can be solved while leaving the model setup practically the same as when using the **Plasma** interface. All of the dependent variables may be solved for in the product space, and periodic boundary conditions are applied across the extra dimension. The **Time Periodic** study is used to solve this problem, but it is basically the equivalent of a **Stationary** study.

As well as reducing the computational burden, the technique also has the following advantages:

- Integrals performed over both the extra dimension and the base geometry can be used in the equations themselves. This allows the contacts and terminals to be driven directly with a fixed power rather than a fixed voltage. This is important not only for numerical stability, but also for discharges where an alpha to gamma transition occurs. In these cases, a given voltage excitation could produce two different solutions, depending on the initial conditions, but with a fixed power, only one physical solution exists. Also, the discharge power is usually known, whereas the electrode potential is often not.
- The self-DC-bias can be computed easily with an additional equation, rather than the ad hoc methods used in traditional methods.
- Parametric sweeps over the operating conditions are fast and easy, since the problem is not solved in the time domain. So, for 1D models, a range of powers, pressures, frequencies, and so on can be swept in only a few minutes.
- It is relatively easily to include a matching network in the model, so the plasma can be driven from an L–type matching network. It is also easy to compute the plasma impedance at the fundamental frequency which is useful when designing matching networks.
- The harmonics generated by the plasma are still resolved by the method, there are no approximations in the model. It is possible to observe how these harmonics in the discharge current can lead to an impedance mismatch when driving with an external circuit.
- The method is well suited for a modern computer architecture, since reassembling and paging in and out of memory at each time step is not necessary. Nearly all the time is spent performing the factorization of a sparse matrix with a direct solver, which is highly parallelized and runs at a very high FLOP rate.

There are, of course a couple of downsides worth mentioning:

- Since an extra dimension is attached to the base geometry, the memory requirements are high. For 2D models, 64+ GB of memory is recommended.
- The problem formulation is highly nonlinear, which is quite a challenge for the solver. Convergence might be very difficult for a given setup, depending on the exact geometry and operating conditions.

The Importance of 1D Modeling

As explained above, 2D numerical models of a CCP can be very complex, demanding in terms of computer memory, and time consuming. For these reasons it is strongly suggested to start with a 1D version of the 2D model. 1D models are much faster and numerical stable and as such are ideal to gain insight on the discharge regime of operation, study the plasma chemistry, and investigate parameter spaces quickly.

A 1D model version of an asymmetric CCP reactor is also possible by using a 1D axial symmetric geometry. This way, it is possible to implement the asymmetry between the areas of the ground and powered electrode and allow for the development of a DC self–bias.

In a preparatory phase for a 2D model it is also useful to use a 1D model to check if the mesh resolution is sufficient to correctly describe the discharge physics.

Electric Excitation

In the **Plasma, Time Periodic** interface it is possible to sustain a plasma using power and voltage sources. In both cases a periodic electric potential is applied at a terminal. If a power source is chosen the voltage amplitude of the periodic signal is numerical found to obtain a given power absorbed by the plasma. As an example, when using the **Metal Contact** feature to drive a discharge with a fixed power the following expression and constraint are used:

$$V_s = V_a \cos(2\pi f_p t + \alpha)$$

$$P_{rf} = f_p \int_{\partial t} \int_{\partial\Omega} V_s(\mathbf{n} \cdot \mathbf{J}_i + \mathbf{n} \cdot \mathbf{J}_e + \mathbf{n} \cdot \mathbf{J}_d) dS dt\ .$$

The constraint on the power P_{rf} is used to compute the voltage amplitude V_a such that a fixed amount of power is absorbed by the plasma. In most cases it is numerically more stable to drive a plasma with a power source.

Solving a problem directly for a high power can be numerical difficult. It is strongly recommended to start a simulation with a very low power (a few watts for an industrial reactor) and increase the power to the desired value using the **Auxiliary sweep** in the **Study Extensions** section in the **Study** node. This strategy is used in several models in the application library. In an initial phase of the implementation of a model, a two-study version of the power sweep can be used: first, a study is run at low power, after, its solution is used as initial condition in another study that is used to increase the power. This approach saves the time of running the low power step while polishing the model.

When using the **Auxiliary sweep** the default option for the continuation solver is **Last parameter**. When using this option the solver tries to predict the solution for the next parameter. If power is used as a parameter and with a slow increase in the power (a few watts) it is possible to obtain solutions in just a few iterations. However, in some cases the variation in the parameter needs to be unreasonably small in order for the continuation solver to work well. In those cases it is suggested to use the **No parameter** option, and choose the option **Yes** in **Reuse solution from previous step**. When doing so, the solver does not use a prediction for the next step but it still uses the solution from the previous step as initial condition. With this option it is possible to increase the power with much larger steps, but more iterations are needed.

Meshing Guidelines

When simulating a CCP discharge it is necessary to well resolve the plasma sheath region. A poorly resolved sheath makes the convergence difficult and can lead to physically incorrect results. Quantities that clearly show a poor mesh resolution are the electron temperature and the power absorbed by the electrons. A visual inspection to these quantities is always a good practice and can be done by plotting the results while solving to identify problems at an early stage.

Industrial CCP reactors have very large aspect ratios with the gap between electrodes being about 3 cm and the reactors radius about 30 cm. Using an unstructured mesh (with the same scale in the *r* and *z* directions) in such reactor can lead to a very large number of degrees of freedom because the axial direction needs to be very well resolved. The excessive meshing in the radial direction can be avoided by uncoupling the resolution in the radial and axial directions with the use of a mapped mesh. When using a mapped mesh it is important to convert the mesh (use the **Convert** feature in the mesh node) to avoid spurious patterns in the electron temperature.

Using Consistent Source Frequencies and Period Settings

The **Period** property must correspond to the lower excitation frequency in the model. In the case of one driving frequency, this is quite trivial: it is usually easiest to define a parameter, `f0`, for the frequency, and set the **Period** setting to `1/f0` and the **Frequency** in any **Terminal** or **Metal Contact** feature to `f0`. In the case of multiple frequencies, things become a little more complicated. Because periodic boundary conditions are used on an extra dimension, frequencies can only be exact integer multiples of each other. It is possible to use the two frequencies of 10 and 20 MHz as presented in Figure 6-2, but it is not possible to use two frequencies of 10 and 18 MHz as presented in Figure 6-3.

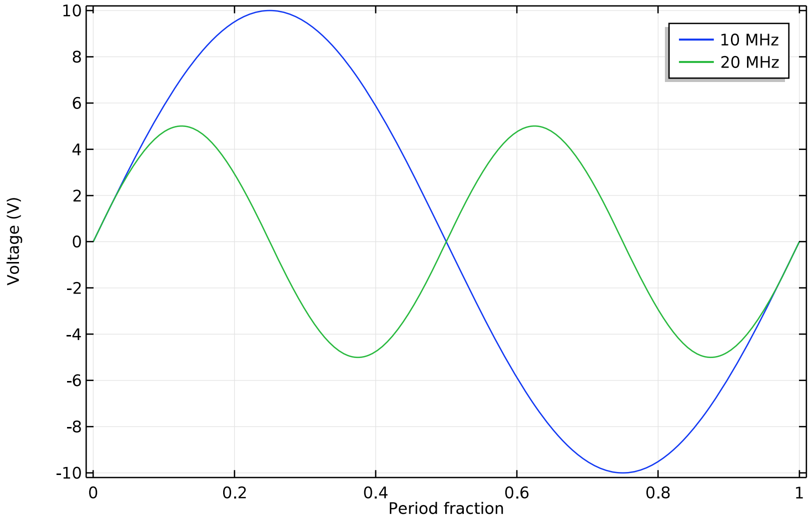


Figure 6-2: Voltage signals for 10 and 20 MHz.

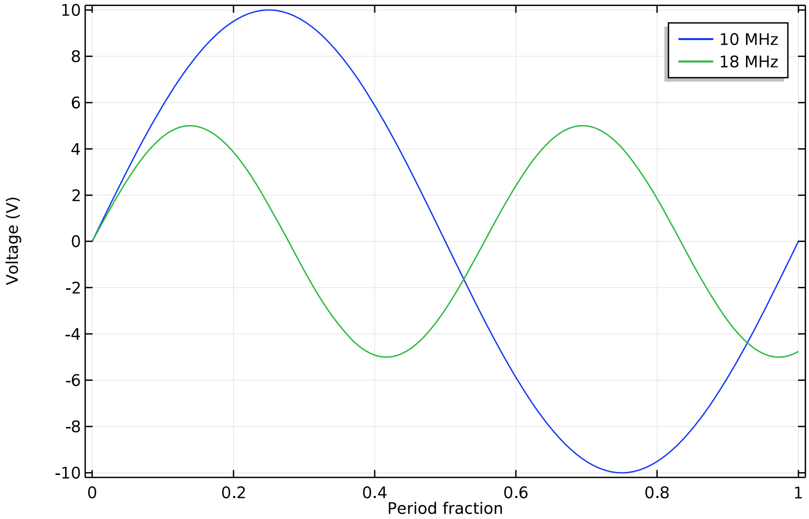


Figure 6-3: Voltage signals for 10 and 18 MHz.

Choosing the Number of Elements in the Extra Dimension

The **Number of elements** in the **Extra Dimension Settings** must be enough to resolve the period evolution of the electric excitation. A minimum of 30 mesh elements per period in the extra dimension is recommended. In some discharges the current and the voltage at the electrodes departs from being a sinusoidal and fine structure might appear. An example of this is a discharge where the electron current contributes significantly to the total current when the sheath collapses. This leads to a burst of electron current that needs to be resolved and consequently more mesh elements in the extra dimension are needed.

When using multiple frequencies to sustain a discharge the number of elements in the extra dimension needs to increase in order to resolve the highest frequency. For a dual frequency CCP reactor with an applied voltage of the from

$$V_s = 10\cos(2\pi f_h t) + \cos(2\pi f_l t) \qquad (6\text{-}21)$$

with a low frequency f_l of 2 MHz and a higher frequency f_h of 60 MHz one needs 900 mesh elements in the extra dimension in order to have 30 mesh elements per period of the high frequency. Figure 6-4 and Figure 6-5 show the signal of Equation 6-21 with different resolutions. When using 100 and 300 elements in the extra dimension the high frequency signal only has 3 and 10 elements per period, respectively, leading to poorly resolved signals.

When using multiple excitation frequencies with frequencies far apart from each other, the number of elements in the extra dimension and consequently the number of degrees of freedom increases dramatically, leading to computationally expensive simulations. In these cases, it is recommended to use a 1D model to know how much resolution of the highest frequency signal is needed to capture all the necessary physics.

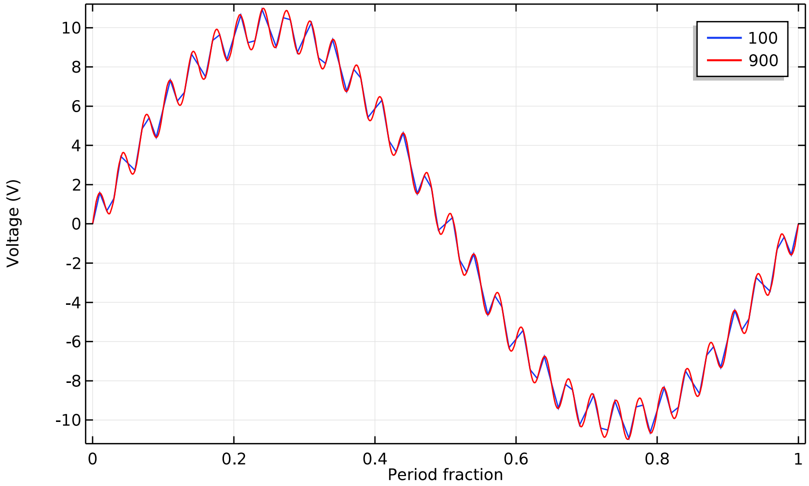


Figure 6-4: Dual frequency voltage for one period of the low frequency with 100 and 900 mesh elements in the extra dimension.

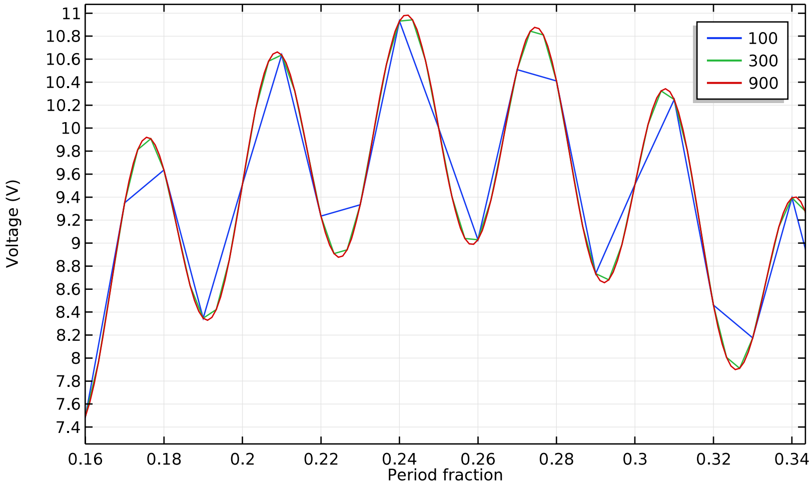


Figure 6-5: Zoom of Figure 6-4 *with the addition of a case with 300 mesh elements in the extra dimension.*

Pulsed Discharges

Particular care must be taken when using the **Pulsed** option in the **Metal Contact** feature. The **Plasma, Time Periodic** interface defines the characteristic period using an additional independent variable, which must be discretized in space using a mesh. To resolve the RF modulation within the on period, at least 30 mesh elements are required per RF period. Consider the case where the discharge is on for 10 cycles at 13.56 MHz (780 ns), then off for another 100 cycles (7.8 μs). During the on-cycle, you need to use 30 mesh elements for each RF period (78 ns). During the off-cycle, there is no RF modulation, so we can use around 100 mesh elements for the entire 7.8 μs. So, we need a total of 310 mesh elements to discretize the extra coordinate representing the characteristic period. This represents a 10-fold increase in computation demand compared to a continuous mode discharge, which would only require 30 mesh elements in the extra coordinate. Therefore, the pulsed option is very computationally expensive, and even 1D models are expected to take a significant amount of time to solve.

The Dual pulse option, where the off-period is replaced by a low power period is even more computationally expensive, since the low power period also needs 30 mesh elements per RF period. So, for the above example, where the 100 cycle off period is replaced by low power operation, we would need 3000 mesh elements to resolve this part. This would make such a model 100 x more expensive than the same continuous mode discharge.

When considering using the pulsed discharge option, use the following to estimate the computation demand:

- In 1D, set up the model and chemistry of interest, and solve the problem using a continuously operated mode. The computation time for this is t_c.
- For the Pulsed option, the expected computation time would be t_c multiplied by the number of RF periods than the discharge is on for.
- For the Dual pulsed option, the expected computation time would be t_c multiplied by the number of RF periods within the characteristic period.

Choosing the Correct Space on Which to Solve for the Heavy Species

When **Product space** is used for the heavy species, the transport equations in the drift–diffusion approximation are solved in the product space of the base geometry and the extra dimension. With this option it is possible to define the **Electric field applied to ions** to be **Instantaneous** or **Time averaged**. These two options correspond to two physical limits where the ions respond instantaneously to the electric field or respond only to the time averaged electric field. The latter limit case, the high-frequency case, is valid when:

$$\omega_{\mathrm{pi}}^2 << \omega^2 \qquad (6\text{-}22)$$

where ω is the electric excitation frequency, and ω_{pi} is the ion plasma frequency. For typical conditions of an industrial CCP reactor using an excitation frequency of 13.56 MHz the ion density in the sheaths (where the electric fields are more intense) is found to be around 10^{15} m^{-3}. For these conditions and for an argon plasma $\omega_{\mathrm{pi}}^2/\omega^2 \sim 0.06$ and consequently the ions only respond to the average potential. The frequency region between the two limit cases is only possible to be described by including the time derivative of the ion velocity in the momentum equation, which is not included in the Plasma Module at the present time.

At high excitation frequencies (in the MHz and above) it is also common to find that the ions and neutral species densities do not change during the excitation period. In these cases it is possible to obtain accurate solutions by solving the heavy species in the **Base geometry**. When using this option the heavy species transport equations are solved in the stationary limit (time derivatives of densities set to zero), the source terms are averaged along the period, and the ion migration is computed with the average field. Since the heavy species are not solved along the period in the product space there is an enormous reduction in degrees of freedom. Consequently, using the **Base geometry** for 2D models reduces considerable the computational time. However, for 1D models the reduction of degrees of freedom is not enough to compensate for the computational cost of numerical averages.

To exemplify some of the consequences of the options discussed above, a 1D model for helium was solved with the heavy species in the **Product space** and in the **Base geometry**. Figure 6-6 shows the results for the total current at the metal contact (only one curve is showed since they are practically the same in both cases), and the ion current. When using the **Product space** the ion current is period modulated through the electric field in the ion migration. When solving the heavy species in the **Base geometry** the ion current is effectively the period average of the one obtained in the **Product space**.

When preparing a 2D model and in doubt of what option to use for the heavy species it is a good practice to test the different options in an equivalent 1D model.

ION TRANSPORT IN PULSED DISCHARGES

Solving for the ions in the base geometry only is much more efficient, and numerically stable, but it does assume that the number density and spatial distribution does not change over the characteristic period. This is a reasonable assumption for the continuous mode operation, but for pulsed discharges with long on or off periods, this assumption begins to break down. Therefore, for pulses where the characteristic period lasts for more than a few microseconds, it may become necessary to solve for the heavy species in the product space, which may be too computationally expensive, or impossible to obtain a converged solution. Therefore, the pulsed options can really only be used to study the electrical characteristics of a discharge, and not the transport of heavy species to reactor walls for discharges with prolonged off periods.

Validity of Fluid Models

The fluid model of a plasma is only valid if the mean free path of the charged particles is much smaller that the reactor characteristic dimensions. This limit occurs around 10 s of millitorr for most CCP industrial reactors. In a low collisional discharge physical mechanisms such as stochastic heating and nonlocal ionization are important to correctly describe the plasma and are not included in the COMSOL Multiphysics Plasma Module at the moment.

Numerical solutions at the collisional limit can still be obtained but caution should be used when analyze them since unphysical behaviors might be present. Also, close to the collisional limit, numerical instabilities tend to cause problems. One of the reasons for the instability is that when applying very large voltages (100s of volts) the electron density reaches very low values in the sheath and as a consequence the electron mean energy (electron energy density divided by the electron density) reaches unphysical values that make the numerical problem very difficult to solve. The **Uniform** model from the **Secondary emission model** property allows to introduce a uniform background electron source that can prevent the electron energy to reach abnormal values.

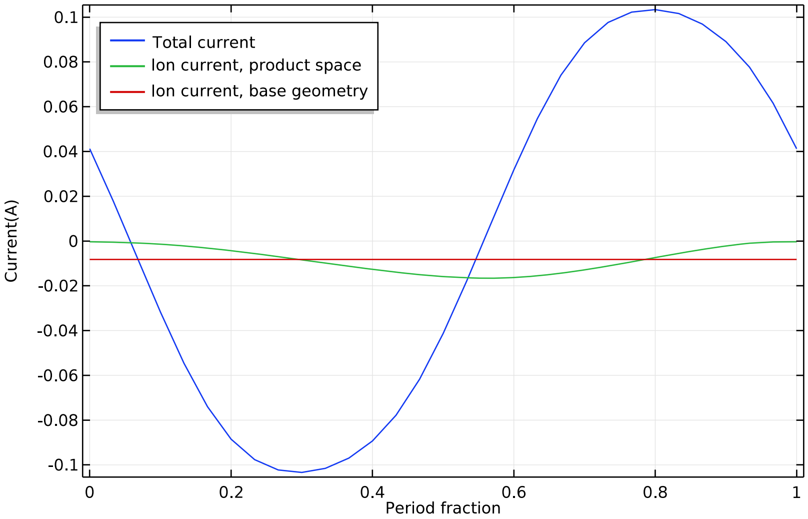


Figure 6-6: Total current and ion current at a metal contact when using the product space and the base geometry.

Complex Chemistries

Plasma chemistries with many species, excited states and reactions are considered complex. Some approaches to help in the implementation of complex plasma chemistries are given below.

- When preparing a new plasma chemistry, 0D and 1D models should be used to take advantages of the fast computational times.
- When there are species from different elements (for example, an Ar/O_2 plasma) the **Inflow** or the **Species Group** feature should be used to facilitate mass conservation for each of the chemical elements. The **Inflow** constrains species at a boundary and is

easier to obtain converge than when using the **Species Group**. The **Inflow** feature can be used as a strategy to ensure mass conservation even if there is no flow in the system.

- If a given species is suspected to be problematic, in that it may be confined to the plasma core, or have very steep gradients, or be negatively charged, use a **Species Group Constraint** feature to constrain its mass fraction to the initial value. The results obtained with the species constrained can then be used as initial condition to a separate study.
- Start with a simple chemistry (species and reactions) and add complexity. A possible approach to implement an Ar/O_2 plasma chemistry would be: (i) solve for Ar only; (ii) add O_2 excitation electron impact reactions which only change the mean electron energy and do not create any other species. Use the **Species Group** feature for O_2 to facilitate the mass conservation of the system. When other O_2 and O states are introduced in the chemistry they should be added to the **Species Group** feature; (iii) add O_2 ionization and ion losses to the walls; (iv) add other excited states and their corresponding reactions; (v) add O_2 dissociation and all chemistry associated with O including surface recombination; (vi) add electron attachment reactions and ion-ion recombination.

Solver Settings

If a model fails to converge, it could be because the default solver settings are too aggressive. There is always a tradeoff between speed and robustness, and the default settings cannot satisfy all cases. The key solver setting that can be changed to help with convergence can be found in the **Fully Coupled** node under **Stationary Solver** in the **Time Periodic** study. In the **Method and Termination** section, there is a parameter called **Restriction for step-size increase** (default value 0.05). This controls how much the damping factor can change between subsequent iterations. If it is difficult to obtain convergence (for a model whose chemistry has been validated in 1D, and is adequately meshed), it is worth lowering the value to 0.02, or even 0.01. This means the solver will have to take more iterations to solve, but it should be more stable.

Inductively Coupled Plasma Reactors with RF Bias

Inductively coupled plasmas reactors with a capacitively coupled RF bias (ICP/CCP reactors) are notoriously difficult to model. The main difficulties are inherited from the modeling of the capacitively coupled component of the problem. The capacitive coupled part is discussed in Capacitively Coupled Plasmas and it is advised to be familiar with this part of the problem first. The inductive part is easy to solve and is discussed in Theory for Inductively Coupled Plasmas.

To model ICP/CCP reactors COMSOL Multiphysics has a dedicated interface called **Inductive Coupled Plasma with RF Bias** that couples the **Plasma, Time Periodic** and the **Magnetic Fields** interfaces. The main difference when comparing with the **Inductive Coupled Plasma** is that the **Plasma** interface is replaced by the **Plasma, Time Periodic** interface. The capacitively coupled excitation is applied in the context of the **Plasma, Time Periodic** interface and consequently a periodic steady-state solution is solved directly. The inductively coupled part of the excitation (normally provided by a coil) is solved by the **Magnetic Fields** interface in the frequency domain.

Below are presented some important aspects and strategies about modeling of ICP/CCP reactors.

Solution Strategy

The recommended solution strategy to model an inductively coupled plasmas with RF bias is as follows. First solve the problem with inductively coupled plasma only using the **Plasma, Time Periodic** interface. To speed up this study set the **Number of elements** in the **Extra Dimension Settings** to 1. Since there is no RF bias applied yet it is not needed to have resolution along the period. After, add a second study that uses the solution of the previous study as initial conditions. Increase the **Number of elements** to 30-50 and use the **Auxiliary sweep** to ramp up the amplitude of the applied voltage.

RF Bias Excitation

For pure CCP problems it is advised to excite the system with a power source (see Electric Excitation) because it makes convergence easier. On the contrary, in ICP/CCP reactors the problem becomes more difficult to solve when using the power source option. For ICP/CCP models it is recommended that the RF bias to be excited using a voltage source. As an example for CCP reactors it is advised to set the **Terminal type** to **Power** in the **Metal Contact** but for ICP/CCP reactors it is advised to set **Terminal type** to **Voltage**.

Period and Spatial Resolution

At the biased electrode the electrons can have strong period modulations and can attain high energies as in a pure CCP reactor. For these reasons the period and space resolution near the biased electrode need to be high as in a pure CCP reactor model. Even when the bias has low voltage amplitudes the current collected deviates strongly from a sinusoidal with the electron current having a strong contribution when the plasma sheath collapses. It is good practice to plot the current along the period of the biased electrode to see if more resolution is needed along the period. It is also good practice to plot the electron temperature to see if there is enough spatial resolution in the vicinity of the biased electrode.

DC Discharges

The theory for modeling DC discharges is described in this section:

- Introduction to DC Discharge Theory
- DC Discharge Boundary Conditions
- Tips for Modeling DC Discharges

Introduction to DC Discharge Theory

Due to the complexity of coupling the electrostatic field to the transport of electrons and heavy species, the Plasma Module provides the Plasma interface for modeling DC discharges. The complicated coupling between the electron transport, heavy species transport, and electrostatic field is handled automatically by the software. Furthermore, the secondary emission flux from ion bombardment on an electrode is automatically computed and used in the boundary condition for electrons. This makes it easy to model things such as positive columns and DC glow discharges.

The physics of DC discharges can be further complicated when a strong, static magnetic field is present. This can lead to highly anisotropic transport properties for the electrons. The Plasma interface automatically computes the tensor form of the electron mobility if requested. DC discharges require reaction rates to be specified in terms of Townsend coefficients rather than rate coefficients. The main reason is that Townsend coefficients provide a better description of the physics in the cathode fall region. The discharge is sustained by secondary emission, rather than a time varying electric field. Therefore, secondary emission effects must be included in any DC discharge model otherwise the plasma simply self-extinguishes.

It is important to be aware of the different time scales at which the components of a plasma can change:

TABLE 6-7: VARIOUS TIME SCALES IN A PLASMA.

PHENOMENA	TIME SCALE
Electromagnetic field propagation	Instantaneous
Electron energy	< 1 ns
Electron transport	nanoseconds
Ion transport	microseconds

TABLE 6-7: VARIOUS TIME SCALES IN A PLASMA.

PHENOMENA	TIME SCALE
Excited species transport	milliseconds
Neutral gas flow	10's milliseconds

DC Discharge Boundary Conditions

Since secondary emission effects are vital in sustaining DC discharges. Therefore the boundary conditions play an important role in DC discharges. The secondary emission flux for electrons is defined as:

$$\mathbf{n}\cdot\Gamma_{e,s} = \sum_{k=1}^{N}\gamma_k(\Gamma_k\cdot\mathbf{n})$$

Additionally, the secondary emission energy flux requires that the mean energy of the secondary electrons are specified:

$$\mathbf{n}\cdot\Gamma_{\varepsilon,s} = \sum_{k=1}^{N}\gamma_k\bar{\varepsilon}_k(\Gamma_k\cdot\mathbf{n})$$

The secondary emission fluxes are automatically computed by the Plasma interface when surface reactions are added to a model. The mean electron energy of the emitted electrons, ε_k is typically a function of the ionization energy of the impinging ion and the work function of the surface:

$$\bar{\varepsilon}_k = \Delta\varepsilon_k - 2W_f$$

where $\Delta\varepsilon_k$ is the ionization energy (SI unit: V) and W_f is the work function of the surface (SI unit: V). On dielectric surfaces, charge accumulates due to the difference in fluxes between the electrons and ions:

$$\frac{d\rho_s}{dt} = \mathbf{n}\cdot\mathbf{J}_i + \mathbf{n}\cdot\mathbf{J}_e$$

where $\mathbf{J}_i$ is the normal total ion current and $\mathbf{J}_e$ is the normal total electron current. The surface charge density, ρ_s is used as a boundary condition in the Electrostatics interface:

$$-\mathbf{n}\cdot\mathbf{D} = \rho_s$$

Tips for Modeling DC Discharges

There are a number of key points to consider when modeling DC discharges:

- For atmospheric pressure DC discharges, the plasma might naturally try to arc. This typically causes the solver to terminate because it cannot reach the desired tolerances. Often this behavior is undesired, so in practical experimental setups a series RC circuit is added between the driven electrode and the power supply. This prevents arcing from occurring and results in a glow discharge. It is recommended to use the **Series RC circuit** option in the **Circuit Settings** section of the **Electric Potential** feature.
- Townsend coefficients should be used in preference to rate coefficients if possible. Townsend coefficients give a more realistic description of cathode fall regions in DC discharges.
- Secondary emission from a surface bombarded by ions must be included somewhere in the model in order for the discharge to be sustained.
- Although a DC discharge ultimately reaches a steady-state condition, the problem is often solved as a time-dependent problem so the plasma can naturally evolve into its equilibrium state.
- In some cases it is possible to use the stationary solver to obtain directly the stationary solution. Normally, this approach is more difficult to setup than a time-dependent one since initial conditions play an important role and some adjustment in solver settings might be needed. This initial effort is often rewarding because models tend to solve faster than a time-dependent approach and parameterization are easier. A good strategy is to use a time-dependent solver to find a solution that can be used as initial conditions for a stationary solver.
- The initial conditions should enforce the electroneutrality constraint.

Theory for Inductively Coupled Plasmas

The Inductively Coupled Plasma Interface and The Microwave Plasma Interface theory is described in this section:

- Introduction to Inductively Coupled Plasmas
- Domain Equations for the Inductively Coupled Plasma Interface
- Wave Heated Discharge Theory
- Plasma Conductivity in the Presence of a Static Magnetic Field
- Plasma Conductivity in the Presence of a Static Magnetic Field

Introduction to Inductively Coupled Plasmas

Inductively coupled plasma (ICP) reactors are important in plasma processing and plasma sources because the plasma density can be considerably higher than in capacitively coupled discharges. ICP reactors are also attractive from the modeling perspective because they are relatively straightforward to model. Inductively coupled plasmas are relatively straightforward to model due to the fact that the induction currents can be solved for in the frequency domain. This means that the RF cycle applied to the driving coil does not need to be explicitly resolved. As such, the quasi steady-state solution is reached in relatively few time steps.

In a 2D axial symmetric ICP reactor, a coil driven at a high frequency (usually 13.56 MHz) creates a magnetic field with only an in-plane component (the *rz*-plane) and a high frequency out-of-plane component of the electric field (in the θ direction). This results in oscillatory electron motion in the θ direction only. Due to this effect, further simplifications can be made and the out-of-plane electric field can be solved for in the frequency domain. Consider the momentum conservation of electrons as given in Equation 4-3:

$$\frac{\partial}{\partial t}(n_e m_e \mathbf{u}_e) + \nabla \cdot n_e m_e \mathbf{u}_e \mathbf{u}_e^T = -(\nabla \cdot \mathbf{p}_e) + q n_e \mathbf{E} - n_e m_e \mathbf{u}_e \nu_m$$

where m_e is the electron mass (SI unit: kg), $\mathbf{u}_e$ is the drift velocity of the electrons (SI unit: m/s), $\mathbf{p}_e$ is the electron pressure tensor (SI unit: Pa), q is the electron change (SI unit: s·A), $\mathbf{E}$ is the electric field (SI unit: V/m) and ν_m is the momentum transfer frequency (SI unit: 1/s). Neglecting the inertial term and taking only the θ-component yields

$$\frac{\partial}{\partial t}(n_e m_e u_{e,\theta}) = -\frac{\partial}{\partial\theta}(n_e k_B T_e) + q n_e E_\theta - n_e m_e u_{e,\theta}\nu_m .$$

Since n_e and T_e are uniform in the θ direction, the first term on the right-hand side can be neglected, resulting in

$$\frac{\partial}{\partial t}(n_e m_e u_{e,\theta}) = q n_e E_\theta - n_e m_e u_{e,\theta}\nu_m$$

which is a linear equation in for $u_{e,\theta}$ that can be Fourier transformed by letting

$$E_\theta = \tilde{E}_\theta e^{j\omega t}$$

and

$$u_{e,\theta} = \tilde{u}_{e,\theta} e^{j\omega t}$$

The resulting equation in the frequency domain is

$$j\omega n_e m_e \tilde{u}_{e,\theta} = q n_e \tilde{E}_\theta - n_e m_e \tilde{u}_{e,\theta}\nu_m .$$

Multiplying this equation by q and defining the out-of-plane electron current density as

$$\tilde{J}_\theta = q n_e \tilde{u}_{e,\theta}$$

the following expression is obtained:

$$\tilde{J}_\theta = \frac{q^2 n_e}{m_e(\nu_m + j\omega)}\tilde{E}_\theta$$

which relates the current density to the electric field in the form of $J = \sigma E_\theta$ with the plasma conductivity, σ, defined as

$$\sigma = \frac{q^2 n_e}{m_e(\nu_m + j\omega)}$$

which is required by Equation 6-23. The third term on the left-hand side of Equation 4-4 now has an additional contribution due to the out-of-plane motion of the electrons:

$$\mathbf{E}\cdot\Gamma_e = E_\theta\cdot\frac{J_\theta}{q} = \frac{1}{q}\left(\frac{1}{2}\right)\mathrm{real}(\tilde{E}_\theta\tilde{J}_\theta)\,.$$

The presence of the driving coil in an ICP reactor also introduces capacitive coupling between the coil and the electrons in the plasma sheath. This is neglected in COMSOL Multiphysics. This is because the electron density in an ICP reactor is typically so high that the Debye length is very small, meaning that the in-plane component of the electric field cannot penetrate into the plasma.

From the theory of the Boltzmann equation in the two-term approximation (see Boltzmann Equation, Two-Term Approximation) the plasma conductivity needs to be modified in order to be coherent with the drift velocity obtained from the Boltzmann equation in the two term approximation.

In all of the above, it has been assumed that the RF frequency is less than the collision frequency between the electrons and the neutrals. In cases where the gas pressure is low, and the RF frequency is high, collisionless heating can occur, in which case the electron inertia cannot be neglected. For a gas pressure of 20 mtorr, the typical collision frequency is on the order of 30 MHz.

When collisionless heating is included in the model an extra degree of freedom for the out-of-plane electron drift velocity, u_θ, is added:

$$n_e u_\theta j\omega t + n_e \upsilon_e u_\theta - \eta_{\mathrm{eff}}\nabla\cdot(n_e\nabla u_\theta) = -\frac{q}{m_e}n_e E_\theta$$

where η_{eff} is the effective viscosity coefficient defined as:

$$\eta_{\mathrm{eff}} = \left(\frac{c^2 V_{th}^4}{\pi^2\omega_p^2\omega}\right)^{\frac{1}{3}}$$

and V_{th} is the thermal velocity defined as:

$$V_{\mathrm{th}} = \sqrt{\frac{3T_e q}{m_e}}$$

and ω_p is the plasma frequency defined as

$$\omega_p = \sqrt{\frac{q^2 n_e}{\varepsilon_0 m_e}}.$$

The plasma conductivity, σ, is set to zero and the out-of-plane current density, J_ϕ has the following term added to it:

$$J_\theta = n_e u_\theta q .$$

Domain Equations for the Inductively Coupled Plasma Interface

The Inductively Coupled Plasma Interface solves a system of coupled partial differential equations for:

- The electron density
- The mean electron energy
- The mass fraction of each of the heavy species
- The electrostatic potential
- The electric field due to induction currents

For a nonpolarized plasma, the induction currents are computed in the frequency domain using the following equation:

$$(j\omega\sigma - \omega^2\varepsilon_0\varepsilon_r)\mathbf{A} + \nabla \times (\mu_0^{-1}\nabla \times \mathbf{A}) = \mathbf{J}^{\mathrm{e}} \qquad (6\text{-}23)$$

- Plasma Reactors Theory
- The Magnetic Fields Interface in the *AC/DC User's Guide*

The links to items described in other user's guides or reference manuals do not work in the PDF, only from the on line help.

Theory for Microwave Plasmas

Microwave plasmas, or wave-heated discharges, find applications in many industrial areas such as semiconductor processing, surface treatment, and the abatement of hazardous gases.

Introduction

Microwave plasmas are sustained when electrons can gain enough energy from an electromagnetic wave as it penetrates into the plasma. The physics of a microwave plasma is quite different depending on whether the TE mode (out-of-plane electric field) or the TM mode (in-plane electric field) is propagating. In both cases, it is not possible for the electromagnetic wave to penetrate into regions of the plasma where the electron density exceeds the critical electron density (around $7.6{\cdot}10^{16}$ $1/\mathrm{m}^3$ for 2.45 GHz). The critical electron density is given by the formula:

$$n_e = \frac{\varepsilon_0 m_e \omega^2}{q^2}$$

where ε_0 is the permittivity of free space, m_e is the electron mass, ω is the angular frequency, and q is the electron charge. This corresponds to the point at which the angular frequency of the electromagnetic wave is equal to the plasma frequency. The pressure range for microwave plasmas is very broad. For electron cyclotron resonance (ECR) plasmas, the pressure can be on the order of around 1 Pa, while for non-ECR plasmas, the pressure typically ranges from 100 Pa up to atmospheric pressure. The power can range from a few watts to several kilowatts. Microwave plasmas are popular due to the cheap availability of microwave power.

Wave Heated Discharge Theory

In the Microwave Plasma interface, the electromagnetic waves are computed in the frequency domain and all other variables in the time domain. In order to justify this approach, we start from Maxwell's equations, which state that:

$$\nabla \times \tilde{\mathbf{E}} = -\frac{\partial}{\partial t}\tilde{\mathbf{B}} \tag{6-24}$$

$$\nabla \times \tilde{\mathbf{H}} = \tilde{\mathbf{J}}_p + \frac{\partial}{\partial t}\tilde{\mathbf{D}} \tag{6-25}$$

where $\tilde{\mathbf{E}}$ is the electric field (V/m), $\tilde{\mathbf{B}}$ is the magnetic flux density (T), $\tilde{\mathbf{H}}$ is the magnetic field (A/m), $\tilde{\mathbf{J}}_p$ is the plasma current density (A/m^2), and $\tilde{\mathbf{D}}$ is the electrical displacement (C/m^2). The tilde is used to denote that the field is varying in time with frequency $\omega/2\pi$. The plasma current density can be approximated by this expression:

$$\tilde{\mathbf{J}}_p = -qn_e\tilde{\mathbf{v}}_e \tag{6-26}$$

where q is the unit charge (C), n_e is the electron density (1/m^3), and $\tilde{\mathbf{v}}_e$ is the mean electron velocity under the following two assumptions:

1 Ion motion is neglected with respect to the electron motion on the microwave time scale.

2 The electron density is assumed constant in space on the microwave time scale.

The mean electron velocity on the microwave time scale, $\tilde{\mathbf{v}}_e$, is obtained by assuming a Maxwellian distribution function and taking a first moment of the Boltzmann equation Ref. 2:

$$\frac{\partial\tilde{\mathbf{v}}_e}{\partial t} = -\frac{q}{m_e}\tilde{\mathbf{E}} + \nu_m\tilde{\mathbf{v}}_e \tag{6-27}$$

where m_e is the electron mass (kg) and ν_m is the momentum transfer frequency between the electrons and background gas (1/s). As pointed out in Ref. 1, the equations are linear, so we can take a Fourier transform of the equation system. Taking a Fourier transform of equation Equation 6-27 gives

$$j\omega\bar{\mathbf{v}}_e + \nu_m\bar{\mathbf{v}}_e = \frac{q}{m_e}\bar{\mathbf{E}}$$

where the tilde has been replaced by a bar to reflect the fact that we are now referring to the amplitude of the fields. Multiplying both sides by *-en*$_e$ and rearranging gives:

$$-qn_e\bar{\mathbf{v}}_e = \frac{n_eq^2}{m_e(\nu_m + j\omega)}\bar{\mathbf{E}}$$

or, in simpler form

$$\bar{\mathbf{J}}_p = \sigma\bar{\mathbf{E}}$$

where

$$\sigma = \frac{n_e q^2}{m_e(\nu_m + j\omega)} \qquad (6\text{-}28)$$

Equation 6-24 and Equation 6-25 can be rearranged by taking the time derivative of Equation 6-25 and substituting in Equation 6-24:

$$\nabla \times \mu^{-1}\nabla \times \bar{\mathbf{E}} = (\omega^2\varepsilon_0\varepsilon_r - j\omega\sigma)\bar{\mathbf{E}}$$

where μ is the permeability, σ is given in equation Equation 6-28 above, and the plasma relative permittivity set to one. The equation could also be recast where the relative permittivity is complex-valued and the plasma conductivity is zero Ref. 3. The convention employed throughout the Plasma Module is that the plasma conductivity is given by equation Equation 6-28 and the plasma relative permittivity is set to 1.

From the theory of the Boltzmann equation in the two-term approximation (see Boltzmann Equation, Two-Term Approximation) the plasma conductivity needs to be modified in order to be coherent with the drift velocity obtained from the Boltzmann equation in the two term approximation.

Solving the above equation with appropriate boundary conditions allows for the power transferred from the electromagnetic fields to the electrons to be calculated:

$$Q_{rh} = \frac{1}{2}\mathrm{real}(\bar{\mathbf{J}} \cdot \bar{\mathbf{E}}^{\circ})$$

where **J** is the total current density (the plasma current plus the displacement current density) and ° denotes the complex conjugate.

AMBIPOLAR FIELDS

In addition to the equation above, a set of equations are solved in the time domain for the electron density n_e, electron energy density n_ε, plasma potential V, and all ionic and neutral species. These are the same equations solved in the Plasma interface and are given in the Plasma Reactors Theory section.

TE AND TM MODE PROPAGATION

In 2D or 2D axisymmetric models, the electromagnetic waves propagate in either the transverse electric (TE) mode or the transverse magnetic (TM) mode. In the TE mode, the electric field is only in the transverse direction and the magnetic field in the direction of propagation. Therefore, COMSOL Multiphysics solves only for the out-of-plane component of the high-frequency electric field. In the TM mode, the magnetic field is in the transverse direction and the electric field only in the direction of propagation, so COMSOL Multiphysics solves only for the in-plane components of the high-frequency electric field.

TE MODE

In the TE mode, electrons do not experience any change in the high-frequency electric field during the microwave time scale. This means that the phase coherence between the electrons and electromagnetic waves is only destroyed through collisions with the background gas. The loss of phase coherence between the electrons and high-frequency fields is what results in energy gain for the electrons. Therefore, the momentum collision frequency is simply given by:

$$\nu_m = \nu_e$$

where ν_e is the collision frequency between the electrons and neutrals.

TM MODE

The TM mode causes in-plane motion of the electrons on the microwave time scale, so in regions where the high-frequency electric field is significant (the contour where the electron density is equal to the critical density), the time-averaged electric field experienced by the electrons may be nonzero. This destroys the phase coherence between the electrons and the fields, causing the electrons to gain energy. This is an example of a nonlocal kinetic effect, which is difficult to approximate with a fluid model. However, since this effect is similar to collisions with a background gas, the nonlocal effects can be approximated by adding an effective collision frequency to the momentum collision frequency:

$$\nu_m = \nu_e + \nu$$

where ν is the effective collision frequency to account for nonlocal effects. This is discussed in more detail in Ref. 1, where an effective collision frequency of no more than $\omega/20$ is suggested.

ECR REACTORS

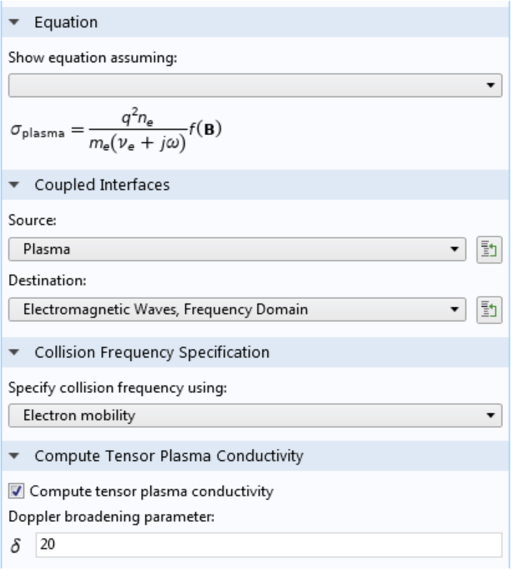
When modeling ECR (*electron cyclotron resonance*) reactors, another layer of complication is added to the problem. The electron transport properties become tensors and functions of a static magnetic flux density, which can be created using permanent magnets. The plasma conductivity also becomes a full tensor, and a highly nonlinear function of the static magnetic flux density. In addition, it is necessary to consider all three components of the electromagnetic field. Comprehensive details on how to set up and solve a model of an ECR reactor can be found in the Dipolar Microwave Plasma Source model documentation.

RESONANCE ZONE

The above equations are quite straightforward to solve, provided that the plasma frequency is below the angular frequency everywhere in the modeling domain. At a frequency of 2.45 GHz, this corresponds to an electron density of $7.6 \cdot 10^{16}$ $1/\text{m}^3$, which is lower than most industrial applications. When the plasma density is equal to this value, the electromagnetic wave transitions from propagating waves to evanescent waves. Applications of microwave plasmas where the electron density is greater than the critical density include:

- Atmospheric pressure discharges, where the electron density can be several orders of magnitude higher than the critical density.
- Traveling-wave-sustained and surface-wave discharges. For the surface wave to propagate, the electron density must be higher than the critical density.

The resonance zone can be smoothed by activating the **Compute tensor plasma conductivity** check box in the **Plasma Conductivity Coupling** multiphysics coupling:



Show equation assuming:

$\sigma_{\text{plasma}} = \frac{q^2 n_e}{m_e(\nu_e + j\omega)} f(\mathbf{B})$

Source: Plasma

Destination: Electromagnetic Waves, Frequency Domain

▾ Collision Frequency Specification

Specify collision frequency using: Electron mobility

▾ Compute Tensor Plasma Conductivity

☑ Compute tensor plasma conductivity

Doppler broadening parameter:

δ 20

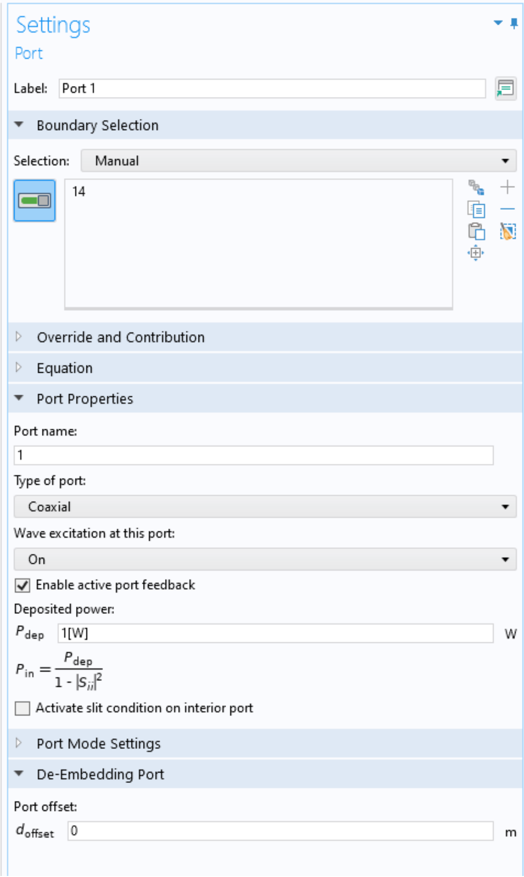
The **Doppler broadening parameter**, δ, corresponds to the value used for the effective collision frequency via the formula:

$$\nu_m = \nu_e + \frac{\omega}{\delta}$$

Therefore, a value of 20 is a compromise between accuracy and numerical stability as detailed above.

DEPOSITED AND REFLECTED POWER

When using the Port boundary condition, the sum of the deposited and reflected power is supplied by default. It is also possible to specify only the deposited power, as shown in the **Settings** window below:



Settings

Port

Label: Port 1

Boundary Selection

Selection: Manual

14

Override and Contribution

Equation

Port Properties

Port name:

1

Type of port:

Coaxial

Wave excitation at this port:

On

☑ Enable active port feedback

Deposited power:

P_{dep} 1[W] W

$$P_{\mathrm{in}} = \frac{P_{\mathrm{dep}}}{1 - |S_{ii}|^2}$$

□ Activate slit condition on interior port

Port Mode Settings

De-Embedding Port

Port offset:

d_{offset} 0 m

When you use the **Enable active port feedback** option, some of the power is deposited and some is reflected back out of the port, depending on the plasma's current state. The plasma can go from absorbing a very small amount to a very large amount of power in a very short time period, which can make the problem numerically unstable or lead to the solver taking extremely small time steps.

MODELING SUGGESTIONS

The following is a collection of tips and tricks to try to help with convergence and decrease computation time:

- For some cases fixing the power into the plasma by using the option **Enable active port feedback** leads to very small time steps. For these cases, the recommended strategy is to not use the **Enable active port feedback** option if possible and use a **Frequency-Transient** study to find a solution (some effort might be needed to find a pair of input power and initial electron density). After, use the solution as initial conditions for a **Frequency-Stationary** solver to parameterized the input power using the **Auxiliary sweep** option. This strategy is demonstrated in the Application Library example *Dipolar Microwave Plasma Source*.

- Start the Doppler broadening parameter to a number below 1, then ramp this number up to 20 over the course of the simulation. This smears out the resonance zone initially, then gradually make the region smaller and smaller. You can implement it by defining a Ramp function with a cutoff value of 20.
- If the number density in the discharge is very high, it is probably operating in full surface-wave mode. In this case, it may be necessary to use a Doppler broadening parameter of 10. This makes the model more stable when solving, but the accuracy of such a model should be carefully considered.
- The initial electron density should be below the critical plasma density in the TM mode case. This recommendation is not strictly necessary for the TE mode case.
- Biasing an electrode may have a significant effect on the discharge because it may shift the contour of critical plasma density away from its unbiased location. The best thing to do is to solve the model with a value of around 5 for the Doppler broadening parameter, then slowly increase its value.

SOLVER SETTINGS

Solver settings play an important role, and COMSOL Multiphysics automatically generates the best solver settings depending on how the model is set up. By default, when the **Enable active port feedback** option is used, the solver settings mentioned below are implemented. The segregated solver is used with two groups:

- All the plasma variables (electron density, electron energy, ion density, plasma potential, and so on).
- All the variables associated with the electromagnetic waves (high frequency electric field, S-parameters).

When the **Enable active port feedback** option in the **Port** boundary condition is used, the solver suggestion is modified so that there are three groups:

- All the plasma variables (electron density, electron energy, ion density, plasma potential, and so on).

- All the variables associated with the electromagnetic waves (high-frequency electric field, S-parameters).
- A dependent variable called `Pdeposited`, which is a differential algebraic equation used to fix the deposited, rather than total power.

In COMSOL Multiphysics the absorption of electromagnetic waves must be realized through a complex-valued electrical conductivity. It is not possible to specify a complex-valued permittivity. The Plasma Conductivity Coupling multiphysics coupling automatically computes the plasma conductivity based on the electron density, collision frequency, and any external DC magnetic flux density that is present.

Plasma Conductivity in the Presence of a Static Magnetic Field

The relationship between the plasma current density and the electric field becomes more complicated in the presence of a DC magnetic field. The following equation defines this relationship:

$$\sigma^{-1}\cdot\mathbf{J}=\mathbf{E}$$

where σ is the plasma conductivity tensor which is a function of the electron density, collision frequency and magnetic flux density. Using the definitions:

$$\alpha=\frac{q}{m_e(\nu_e+j\omega)},\ \beta=n_eq\alpha$$

where q is the electron charge, m_e is the electron mass, n_e is the collision frequency, and ω is the angular frequency of the electromagnetic field, the inverse of the plasma conductivity can be written as:

$$\sigma^{-\mathbf{1}}=\begin{bmatrix}\frac{1}{\beta} & -\frac{\alpha}{\beta}B_z & \frac{\alpha}{\beta}B_y\\ \frac{\alpha}{\beta}B_z & \frac{1}{\beta} & -\frac{\alpha}{\beta}B_x\\ -\frac{\alpha}{\beta}B_y & \frac{\alpha}{\beta}B_x & \frac{1}{\beta}\end{bmatrix}$$

where $\mathbf{B}$ is the magnetic flux density which has three components B_x, B_y, and B_z. The inverse of the plasma conductivity is used because it can be written in a compact, convenient form. The plasma conductivity is a highly nonlinear function of the magnetic flux density. The conductivity exhibits resonant behavior at a critical magnetic flux density given by the simple formula:

$$B_{\mathrm{res}} = \frac{m_e \omega}{q}$$

At this critical magnetic flux density the electrons continually gain energy from the electric field at all phases of the RF cycle. This principle, called electron cyclotron resonance is used in ECR reactors.

References

1. G.J.M. Hagelaar, K. Makasheva, L. Garrigues, and J.-P. Boeuf, "Modelling of a dipolar microwave plasma sustained by electron cyclotron resonance," *J. Phys. D: Appl. Phys.*, vol. 42, p. 194019 (12pp), 2009.

2. R.L. Kinder and M.J. Kushner, "Consequences of mode structure on plasma properties in electron cyclotron resonance sources," *J. Vac. Sci. Technol. A*, vol. 17, p. 2421, 1999.

3. M.A. Lieberman and A.J. Lichtenberg, *Principles of Plasma Discharges and Materials Processing*, Wiley, 2005.

Theory for Global Models

Introduction

If the **Diffusion Model** property is set to **Global** the model equations solved are greatly simplified because the spatial information of the different quantities in the plasma reactor are treated as volume-averaged. Without the spatial derivatives the numerical solution of the equation set becomes considerably simpler and the computational time is greatly reduced. This type of model is useful when investigating a broad region of parameters with complex plasma chemistries.

To avoid unnecessary complexity in the notation the same symbols which were used to represent the space-dependent quantities are used to represent the volume-averaged quantities. In the following the different densities, mass fractions and other quantities are to be thought as volume-averaged.

HEAVY SPECIES

For a mixture consisting of $k = 1, \ldots, Q$ species and $j = 1, \ldots, N$ reactions the mass-fraction balance equations for the first $Q - 1$ species is given by

$$V\rho\frac{d}{dt}(w_k) = m_f w_{f,k} - m_o w_k + VR_k + \sum_l h_l A_l R_{\mathrm{surf},k,l} M_k$$
$$-w_k\left(m_f + m_o + \sum_l h_l A_l M_{f,l}\right)$$

where:

- m_f is the total feed mass-flow rate (SI unit: kg/s))
- m_o is the outlet mass-flow rate (SI unit: kg/s)
- $w_{f,k}$ is the mass fraction of species kth in the feed (1)
- h_l is a correction factor for surface l (1)
- A_l is the area of surface l (SI unit: m^2)
- V is the reactor volume (SI unit: m^3),
- $R_{\mathrm{surf},k,l}$ is the surface rate expression of surface l (SI unit: mol/(m^2·s)).
- $M_{f,l}$ is the inward mass flux of surface l (SI unit: kg/(m^2·s)).

The sum in the last two term is over surfaces were species are lost or created. The forth term on the right hand side accounts for surface reactions of species *k*th. The last term on the right hand side is introduce because the species mass-balance equations are written in the nonconservative form and it is used the mass-continuity equation to replace for the mass density time derivative.

To take into account possible variations of the system total mass or pressure the mass-continuity equation can also be solved

$$V\frac{d\rho}{dt} = m_f - m_o + \sum_l h_l A_l M_{f,l} \,. \tag{6-29}$$

The total feed mass-flow rate feeds is computed from

$$m_f = Q_{\mathrm{SCCM}} N_{\mathrm{Std}} \frac{M_{m_f}}{N_A} \frac{1\ \mathrm{cm}^3}{1\ \mathrm{min}} \tag{6-30}$$

where Q_{SCCM} is the total mass flow given in number of SCCM units, M_{m_f} is the mean molar mass of the feed, and N_{Std} is the standard number density computed at 1 atm and 273.15 K.

Closed Reactor

In the **Reactor** property it is possible to choose three different types of reactor models. If the **Reactor type** is set to **Closed reactor** the mass-flow feeds are set to zero

$$m_f = m_o = 0 \tag{6-31}$$

and Equation 6-29 is solved to take into account possible mass changes in the system caused by surface reactions.

Constant Mass

If the **Reactor type** is set to **Constant mass** the outlet mass-flow is set such that the feed and surface reactions cannot change the mass of the system

$$m_o = m_f + \sum_l h_l A_l M_{f,l} \tag{6-32}$$

and the pressure in the reactor is found in order to keep the mass-density constant

$$\frac{d\rho}{dt} = 0 \,. \tag{6-33}$$

Constant Pressure

If the **Reactor type** is set to **Constant pressure** Equation 6-29 is not solved and the outlet mass-flow feed is set such that the mass feed and surface reactions cannot change the mass of the system

$$m_o = m_f + \sum_l h_l A_l M_{f,l}. \tag{6-34}$$

This condition ensures that the feed and surface reactions cannot change the mass density. However, the mass density can still change as a result of a change in the mean molar mass of the gas mixture in order to maintain a constant pressure.

Global Model Surface Reactions

A great part of a successful implementation of a plasma global modal depends on how the surface losses are estimated. For positive ions the forward rate constant can be estimate to be equal to the Bohm velocity

$$k_{\mathrm{f},i} = \sqrt{\frac{eN_aT_e}{M_k}} \tag{6-35}$$

where T_e is the electron temperature (V).

For neutral species the forward rate constant can be estimate by

$$\frac{1}{k_{\mathrm{f},i}} = \frac{1}{k_{\gamma,i}} + \frac{1}{k_{D,i}} \tag{6-36}$$

$$k_{D,i} = \frac{D_{k,m}}{\Lambda^2{}_{\mathrm{eff}}}\frac{V}{A_l}. \tag{6-37}$$

When the **Motz–Wise** correction option is set to **On**

$$k_{\gamma,i} = \left(\frac{\gamma_i}{1-\gamma_i/2}\right)\frac{1}{(\Gamma_{\mathrm{tot}})^m}\left(\frac{1}{4}\right)\sqrt{\frac{8RT}{\pi M_n}} \tag{6-38}$$

and when the **Motz–Wise** correction option is set to **Off**, the forward rate constant is given by

$$k_{\gamma,i} = \frac{\gamma_i}{(\Gamma_{\mathrm{tot}})^m}\left(\frac{1}{4}\right)\sqrt{\frac{8RT}{\pi M_n}}. \tag{6-39}$$

Equation 6-37 is an estimation of the diffusive losses to the wall where Λ_{eff} is the effective diffusion length, and $D_{k.m}$ is the mixture-average diffusion coefficient of species k.

Surface reactions can be adjusted using the Correction factor h_l. It is common practice to correct the surface ion losses by a factor that takes into account the ion spatial profile. Models for the ion correction factor can be found in the literature Ref. 2 and Ref. 3.

HEAVY SPECIES ENERGY CONSERVATION

The heavy species energy equation can be used to compute the background gas temperature

$$V\rho C_p \frac{dT}{dt} = m_f \sum_k w_{f,k}(h_{f,k} + h_k) + Q + Q_S + V\frac{dp}{dt} \tag{6-40}$$

where C_p is the specific heat at constant pressure of the mixture, T is the gas temperature, $h_{f,k}$ is the enthalpy of species k in the feed, h_k is the enthalpy of species k. The heat source from collisions is given by

$$Q = -V\sum_i H_j r_j + V \sum_{j,\text{ elastic}} 2\frac{m_e}{m_k}\frac{3}{2}\left[T_e(eV) - \frac{k_B}{e}T_{\text{gas}}(K)\right]Fr_j \tag{6-41}$$

where the first term on the right-hand side is the energy release from reactions, H_j is the reaction enthalpy, and r_j is the reaction rate. The last term is the energy gain from elastic collisions between electrons and heavy species. Heat losses by transport are including in a simplified form:

$$Q_S = k\frac{T_S - T}{\Lambda_S^2}V \tag{6-42}$$

where k is the thermal conductivity of the mixture, T_S is the surface temperature, and Λ_S is the diffusion length.

ELECTRON DENSITY AND ELECTRON ENERGY DENSITY

The electron number density is obtained from electroneutrality

$$n_e = \sum_{k=1}^{N} Z_k n_k \tag{6-43}$$

and electron energy density n_ε is computed from

$$e\frac{d}{dt}(n_\varepsilon) = R_\varepsilon + \frac{P_{\mathrm{abs}}}{V} + \sum_{l}\sum_{\mathrm{ions}} e h_l \frac{A_l}{V} R_{\mathrm{surf},\,k,\,l} N_a(\varepsilon_e + \varepsilon_i) \qquad (6\text{-}44)$$

where P_{abs} is the power absorbed by the electrons (SI unit: W), and e is the elementary charge. The last term on the right-hand side accounts for the kinetic energy transported to the surface by electrons and ions. The summation is over all positive ions and all boundaries with surface reactions, ε_e is the mean kinetic energy lost per electron lost, ε_i is the mean kinetic energy lost per ion lost, and N_a is Avogadro's number.

- *Global Model Coupled with the Two-Term Boltzmann Equation*: Application Library path **Plasma_Module/Global_Modeling/ boltzmann_global_model_argon**
- *Chlorine Discharge Global Model*: Application Library path **Plasma_Module/Global_Modeling/chlorine_global_model**

References for Global Modeling

1. R.J. Kee, M.E. Coltrin, and P. Glarborg, *Chemically Reacting Flow Theory and Practice*, Wiley, 2003

2. M.A. Lieberman and A.J. Lichtenberg, *Principles of Plasma Discharges and Materials Processing*, Wiley, 2005.

3. E.G. Thorsteinsson, and J.T. Gudmundsson, "A global (volume averaged) model of a chlorine discharge," *Plasma Sources Sci. Technol.*, vol. 19, p. 015001 (15pp), 2010.

Computing Versus Assuming the EEDF

With the fluid model that COMSOL employs, the electron energy distribution function (EEDF) can either be an assumed function (such as Maxwellian or Druyvesteyn), or it can be computed using a two-term approximation of the Boltzmann equation. When choosing to solve the Boltzmann equation in the two-term approximation (BETT) in a space dependent or global model the BETT is solved fully coupled with the remaining equations of the plasma model. This means that for each time step or iteration, the BETT is solved with the local mole fraction of the different species (and the local ionization degree and electron density if e-e collisions are included) obtained from the fluid model equations. The EEDF obtained from the solution of the BETT is used to compute macroscopic rate constants and transport coefficients intervening in the model equations thus closing the loop.

The coupling of the model equations with the BETT can be made in two ways:

- If the **Local field approximation** is used (and the model is a global model), the reduced field for which the BETT is solved must be given by the user. If the model is space dependent the reduced field is obtained from the solution of Poisson's equation.
- If the **Local energy approximation** is used, the reduced field is computed such that the mean electron energy (obtained by averaging the EEDF) is equal to the mean electron energy obtained from Equation 6-44.

When the BETT is solved, an extra dimension is attached to the reactor geometry (also referred here as base geometry). The extra dimension represents the electron energy coordinate and the BETT is solved in this space. It is important to note that there are as many BETT being solved as elements in the base geometry (if there are 1,000 elements in the base geometry, there are 1,000 BETT being solved). This makes this option very computationally expensive.

Since the recommended number of elements in the extra dimension should be larger than 50 the number of degrees of freedom of the product space (defined by the base geometry and the extra dimension) and the memory requirements increases considerable when the BETT is solved.

In summary, when computing the EEDF:

- Computational time increases considerably. A typical ICP model using an analytic EEDF that normally takes 1 minute will take 24 hours when solving for the EEDF.

- The memory requirements increase considerably. For 2D models, 64GB+ of memory is recommended.
- The problem formulation is highly nonlinear, which is quite a challenge for the solver. Convergence might be very difficult for a given setup, or not even possible, depending on the exact geometry and operating conditions.

The following approaches are considered good practice when solving the BETT coupled with a space dependent or global model:

- The model should first be set up using an analytic EEDF.
- It is advised to do preliminary studies using the **Boltzmann Equation, Two-Term Approximation** interface to understand how the computed and analytic EEDFs differ and how they affect the quantities of interest.
- If the model is space dependent start with a 1D model or a global model if possible.
- An **EEDF Initialization** study is needed to provide initial conditions for the fully coupled problem. With this study only the BETT is solved. Therefore, this study can also be used to do preliminary analyzes. However, keep in mind that there are as many BETT being solved as elements in the base geometry, which implies that even the initialization study can take 10's of minutes in 2D models.

When solving the BETT coupled with a global model or a space dependent model an **EEDF Initialization** study needs to be solved first to provide initial conditions for the fully coupled problem.

In conclusion, special care must be taken when computing the EEDF, as it makes the problem much more nonlinear and convergence is much more difficult, and often impossible. Computing the EEDF should only be used as a final “verification” stage, when the parameter space and operating conditions have been extensively studied using an assumed function. Due to the overwhelming computational demand, only limited technical support can be provided in these advanced models.

- DC Glow Discharge Coupled with the Two-Term Boltzmann Equation: Application Library path **Plasma_Module/ Space-Dependent_EEDF_Modeling/positive_column_1d_boltzmann**
- GEC ICP Reactor, Fluid Model Coupled with the Two-Term Boltzmann Equation: Application Library path **Plasma_Module/ Space-Dependent_EEDF_Modeling/argon_gec_icp_boltzmann**

Using Stationary Studies in Plasma Models

Preparing a plasma model to be solved using a stationary study can be more difficult than with a time-dependent study. However, the initial effort can be very rewarding because models tend to solve faster, parameterization are easy to setup, and because of the possibility to use the Optimization module.

The easiest discharges to solve with a stationary approach are inductively coupled plasma and DC. It is also possible to solve wave heated discharge models when the input power is used instead of deposited power.

Below are a few points to have in mind when using a stationary study:

- Initial conditions of the various species densities play a key role in obtaining a solution. If the initial densities are too far from the solution the solver might not be able to converge. If the initial densities are too low the solver might tend to go to the zero density solution and fail while trying to do so.
- Complex chemistries with more than one species (that is, argon and oxygen) tend to be difficult to model. In part, this is related with the previous point. It can happen that the initial mass fraction is not preserved. In these cases, one of the following strategies is needed: add point constraint for the mass fraction of a species at some point of the reactor wall; use the **Species Group** feature to preserve the mass fraction of a given species group; or add an **Inlet** to a given species.
- Finer meshes are often needed when comparing with a time-dependent approach.
- The solver settings can be adjusted to make convergence possible in some difficult cases, and can also be adjusted to increase the solution time in easier models like ICP reactors. If a model fails to converge, it could be because the default solver settings are too aggressive. There is always a tradeoff between speed and robustness, and the default settings cannot satisfy all cases. The key solver setting which can be changed to help with convergence can be found in the **Fully Coupled** node under **Stationary Solver**. In the **Method and Termination** section, there are parameter called **Restriction for step-size increase** (default value 1) and **Restriction for step-size update** (default value 1.5). These controls how much the damping factor can change between subsequent iterations. If it is difficult to obtain convergence it is worth lowering these values. This means the solver will have to take more iterations to solve, but it

should be more stable. Decreasing the **Restriction for step-size update** to 1.2 or 1.1 can greatly improve the model stability.

- Sometimes it is extremely difficulty to obtain a solution. In this cases a good strategy is to use the solutions of a time-dependent study to provide initial conditions for the stationary study. As an example, solving for corona discharges directly with a stationary study is not recommended. The strategy should be to use the time-dependent solver to get a solution for an easy case (that is, using a low applied voltage) and feed this solution into the stationary solver where parameterizations can be done.
- When using the Microwave Plasma interface make sure that the input power is used instead of deposited power.

Electronegative Discharges

Electronegative plasmas are plasmas that contain negative ions. Negative ions are mainly created by electron dissociative attachment (for example, $e+Cl_2$=>$Cl+Cl^-$). This reaction tends to be very effective at low electron energies and can reduce the electrons in a discharge to a point that an ion-ion discharge is obtained. The transport and volume creation/destruction mechanisms tend to be more complex than electropositive plasmas in many aspects.

In electronegative discharges negative ions are well confined by the ambipolar electric field and losses by transport are very small. This means that to achieve a steady-state volume losses need to be included for negative ions. The mechanisms by which negative ions are lost depend on the gas mixture and pressure and they are: mutual recombination with positive ions (for example, Cl^-+Cl^+=>2C or Cl-+Ar^+=>Cl+Ar), detachment in collisions with excited or neutral atoms or molecules (for example, Cl^-+Cl=>Cl_2+e or Cl^-+Cl_2=>$Cl+Cl_2+e$), and electron-impact detachment (for example, $e+Cl^-$=>Cl+2e).

In electronegative discharges it is often possible to identify two spatial regions using the electronegativity (ratio of the negative ion density to the electron density): (i) one in the core of the discharge (the electronegative core) with high electronegativity where the dominant charge species are positive and negative ions; (ii) and the other close to the boundaries (electropositive edges) where the dominant charged species are electron and positive ions. In the transition between these two regions the negative ion density drops abruptly causing a chock-like phenomena. This transition needs to be well resolved spatially. If not, oscillations can be seen in the negative ion density and the model might not converge. Some strategies to deal with this involve increasing the negative ions diffusion and thy are:

- Increase the negative ion temperature of about 0.3 eV. The transition from the electronegative core of the plasma to the electropositive sheath region can be very abrupt and a higher ion temperature makes the transport numerical easier. The ion temperature is defined in the section **Mobility and Diffusivity Expressions** in the species **Settings**. By default the ion temperature is the gas temperature.
- Enable **Isotropic diffusion for ions** in the **Inconsistent Stabilization** section (the stabilization sections are visible when **Stabilization** is selected in **Show More Options**). This option adds artificial diffusion to all ions and helps smoothing the sharp transition of the negative ion density between the electropositive edge and the

electronegative core, and also increase the density of the negative ions in the electropositive edge effectively increasing its losses by transport. This option should be used very carefully since completely wrong results can be obtained if too much diffusion is used (the tuning parameter for ions should not be larger than 0.1). A useful strategy is to start with a large **Tuning parameter for ions** (for example, 0.5) and ramp it down using an **Auxiliary sweep**.

7

Equilibrium Discharges Interfaces

This chapter describes the physics interfaces found under the **Plasma>Equilibrium Discharges** branch ( ). The Equilibrium Discharges interfaces can be used to model plasma in or in close local thermodynamic equilibrium (LTE), that is, where the electrons and heavy particles temperature are approximately equal. Close to LTE conditions are usually observed at high pressure and under small values of electrical fields.

In this chapter:

Overview of Equilibrium Discharge Interfaces

The **Model Wizard** and the **Add Physics** window contain a number of multiphysics interfaces for modeling of equilibrium discharges, all available under the **Equilibrium Discharges** branch ( ) within the **Plasma** branch ( ). This section gives an overview introduction to these multiphysics interfaces:

- The Equilibrium Discharges Interface ( ): Available in 3D, couples Magnetic and Electric Fields, Heat Transfer in Fluids, and Laminar Flow interfaces with several multiphysics coupling features.
- The Equilibrium Discharges, In-Plane Currents interface ( ): Available in 2D and 2D axisymmetric, it has the same settings as The Equilibrium Discharges Interface.
- The Equilibrium Discharges, Out-of-Plane Currents interface ( ): Available in 2D and 2D axisymmetric, it shares most of the settings with The Equilibrium Discharges Interface. The difference is that the magnetic field is computed with the **Magnetic Fields** interface, which does not solve current conservation equation. This magnetic interface is used here since it is computationally cheaper than the **Magnetic and Electric Fields**. Another difference is that the interface does not include the ion and electron heating at the plasma boundaries.

The Equilibrium Discharges Interface

!	This physics interface requires an AC/DC Module license. For theory and physics interface feature descriptions, see the *AC/DC Module User's Guide*.

The **Equilibrium Discharges** ( ) interface is used to study equilibrium discharges in a magnetohydrodynamics (MHD) framework. This multiphysics interface adds three single physics interfaces: Magnetic and Electric Fields, Heat Transfer in Fluids, and Laminar Flow, together with several multiphysics coupling features. The multiphysics couplings add the MHD coupling between the Magnetic and Electric Fields and the Laminar Flow interfaces. The multiphysics couplings also add heating and cooling of the equilibrium plasma by enthalpy transport, Joule heating and radiation loss as well as special boundary conditions to model the ion and electron heating at the plasma boundaries.

When a predefined **Equilibrium Discharges** interface is added from the **Plasma>Equilibrium Discharges** branch ( ) of the **Model Wizard** or **Add Physics** windows, **Magnetic and Electric Fields**, **Heat Transfer in Fluids**, and **Laminar Flow** interfaces are added to the Model Builder. In addition, the **Multiphysics Couplings** node is added, which automatically includes the multiphysics coupling features **Equilibrium Discharge Heat Source**, **Magnetohydrodynamics**, **Nonisothermal Flow**, and **Equilibrium Discharge Boundary Heat Source**.

On the Constituent Physics Interfaces

The Magnetic and Electric Fields interface is used to compute magnetic field and induced current distributions in and around coils, conductors, and magnets. The Magnetic and Electric Fields interface solves Maxwell's equations formulated using the magnetic vector potential and, optionally for coils, the scalar electric potential as the dependent variables.

The Heat Transfer in Fluids interface provides features for modeling heat transfer by conduction, convection, and radiation. A Fluid model is active by default on all domains. All functionality for including other domain types, such as a solid domain, is also available. The temperature equation defined in solid domains corresponds to the differential form of the Fourier's law that may contain additional contributions like heat sources.

The Laminar Flow interface solves for conservation of energy, mass, and momentum in fluids.

For details about predefined multiphysics couplings, see Multiphysics Modeling Workflow in the *COMSOL Multiphysics Reference Manual*.

SETTINGS FOR PHYSICS INTERFACES AND COUPLING FEATURES

When physics interfaces are added using the predefined couplings, for example **Equilibrium Discharges**, specific settings are included with the physics interfaces and the coupling features.

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

For example, if single **Magnetic and Electric Fields**, **Heat Transfer in Fluids**, and **Laminar Flow** interfaces are added, COMSOL Multiphysics adds an empty **Multiphysics Couplings** node. You can then choose from the available coupling features, **Equilibrium Discharge Heat Source**, **Magnetohydrodynamics**, **Nonisothermal Flow**, and **Equilibrium Discharge Boundary Heat Source** but the modified settings are not included.

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

TABLE 7-1: MODIFIED SETTINGS FOR AN EQUILIBRIUM INDUCTIVELY COUPLED PLASMA INTERFACE.

PHYSICS INTERFACE OR COUPLING FEATURE	MODIFIED SETTINGS
Magnetic and Electric Fields	The discretization of the dependent variables is set to linear.
Heat Transfer in Fluids	In the **Fluid** default feature, the **Absolute pressure,** p_A (**Model Input** section) and the **Velocity field,** $\mathbf{u}$ (**Heat Convection** section) are automatically set to the variables from the **Nonisothermal Flow** multiphysics coupling feature.

TABLE 7-1: MODIFIED SETTINGS FOR AN EQUILIBRIUM INDUCTIVELY COUPLED PLASMA INTERFACE.

PHYSICS INTERFACE OR COUPLING FEATURE	MODIFIED SETTINGS
Laminar Flow	In the **Fluid Properties** default feature, the **Density,** ρ is automatically set to the variable from the **Nonisothermal Flow** multiphysics coupling feature. The **Compressibility** option is set to **Weakly Compressible**. The **Use pseudo time stepping for stationary equation form** check box is automatically selected under the **Advanced Settings** section.
Equilibrium Discharge Heat Source	The Domain Selection is the same as that of the participating physics interfaces. The corresponding Magnetic Fields and Heat Transfer in Fluids interfaces are preselected in the **Coupled Interfaces** section.
Magnetohydrodynamics	The Domain Selection is the same as that of the participating physics interfaces. The corresponding Magnetic Fields and Laminar Flow interfaces are preselected in the **Coupled Interfaces** section.
Nonisothermal Flow	The Domain Selection is the same as that of the participating physics interfaces. The corresponding Laminar Flow and Heat Transfer in Fluids interfaces are preselected in the **Coupled Interfaces** section.
Equilibrium Discharge Boundary Heat Source	The Boundary Selection contains all boundaries of the participating physics interfaces. The corresponding Electric Currents and Heat Transfer in Fluids interfaces are preselected in the **Coupled Interfaces** section.

PHYSICS INTERFACE AND COUPLING FEATURES

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

Coupling Features

- The Equilibrium Discharge Heat Source and Equilibrium Discharge Boundary Heat Source multiphysics coupling nodes are described in this chapter.

- The Magnetohydrodynamics multiphysics coupling node is described in this chapter.
- The Nonisothermal Flow coupling node is described for The Nonisothermal Flow and Conjugate Heat Transfer Interfaces in the *COMSOL Multiphysics Reference Manual*.

Physics Interface Features

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

In general, to add a node, go to the **Physics** toolbar, no matter what operating system you are using.

- The available physics features for The Laminar Flow Interface are listed in the section Domain, Boundary, Pair, and Point Nodes for Single-Phase Flow.
- The available physics features for the heat transfer interfaces are listed in the sections Domain Features, Boundary Features, and Edge and Point Features in the *COMSOL Multiphysics Reference Manual*.
- The available physics features for The Magnetic and Electric Fields Interface are listed in the section Domain, Boundary, Point, and Pair Nodes for the Magnetic Fields Interface in the *AC/DC Module User's Guide*.

Equilibrium Discharge Heat Source

The **Equilibrium Discharge Heat Source** multiphysics coupling ( ) represents the source term Q (SI units: W/m^3) in the heat equation implemented by

$$\rho C_p\left(\frac{\partial T}{\partial t}+\mathbf{u}\cdot\nabla T\right)-\nabla\cdot(k\nabla T) = Q \qquad (7\text{-}1)$$

The source term Q includes three selectable source/sink components:

- Resistive heating (ohmic heating).

$$Q = \mathbf{J}\cdot\mathbf{E} \qquad (7\text{-}2)$$

- Volumetric net radiation loss Q_{rad} defined by the total volumetric emission coefficient, which comes from a material property.

- Enthalpy transport (energy carried by the electric current)

$$\frac{\partial}{\partial T}\left(\frac{5k_BT}{2q}\right)(\nabla T\cdot\mathbf{J}) \tag{7-3}$$

  Note that the enthalpy transport term prevails, for instance, in the boundary layers close to electrodes in a fully ionized electric discharge(Ref. 1).

SETTINGS

The **Label** is the default multiphysics coupling feature name.

The **Name** is used primarily as a scope prefix for variables defined by the coupling node. Refer to such variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different coupling nodes or physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first multiphysics coupling feature in the model) is `phs1`.

HEAT SOURCE COMPONENTS

By default, the **Include enthalpy transport**, **Include Joule heating**, and **Include volumetric net radiation loss** check boxes are selected. From the list, select a **Total volumetric emission coefficient** Q_{rad} (SI unit: W/m^3) — **From material** (the default) or **User defined**. For **User defined**, enter a value or expression in the field. The heat source component are selectable in order to be able to see the effect of each of the component on the plasma dynamics.

EQUILIBRIUM DISCHARGE HEAT SOURCE

This section defines the physics involved in the Equilibrium Discharge Heat Source multiphysics coupling. By default, the applicable physics interface is selected in the **Electromagnetic** list to apply the **Heat transfer** to its physics interface to establish the coupling.

You can also select **None** from either list to uncouple the **Equilibrium Discharge Heat Source** node from a physics interface. If the physics interface is removed from the **Model**

Builder, for example **Heat Transfer in Fluids** is deleted, then the **Heat transfer** list defaults to **None** as there is nothing to couple to.

> If a physics interface is deleted and then added to the model again, and in order to reestablish the coupling, you need to choose the physics interface again from the lists. This is applicable to all multiphysics coupling nodes that would normally default to the once present physics interface. See Multiphysics Modeling Workflow in the *COMSOL Multiphysics Reference Manual*.

Equilibrium Discharge Boundary Heat Source

The **Equilibrium Discharge Boundary Heat Source** multiphysics coupling ( ) maps the electromagnetic surface losses as a heat source on the boundary (SI unit: W/m^2) in the heat transfer part of the model.

SETTINGS

The **Label** is the default multiphysics coupling feature name.

The **Name** is used primarily as a scope prefix for variables defined by the coupling node. Refer to such variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different coupling nodes or physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first multiphysics coupling feature in the model) is `bphs1`.

ELECTRODE PROPERTIES

Select an **Electrode polarity** — **Anode** (the default) or **Cathode**. Enter a **Surface work function** Φ (SI unit: V). The default is 4.15 V.

ELECTRON CURRENT

This section is available when **Cathode** is selected as the **Electrode polarity**.

Enter an **Effective Richardson's constant** A^*_R (SI unit: A/(m^2·K^2)). The default is 120 A/(m^2·K^2). Enter an **Effective work function** Φ_{eff} (SI unit: V). The default is 2.6 V.

ION CURRENT

This section is available when **Cathode** is selected as the **Electrode polarity**. Enter a value or expression for the **Plasma's ionization potential** V_{ion} (SI unit: V). The default is 15.7 V.

SOURCE POSITION

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

Select a **Source position** — **Layer** (the default), **Upside**, or **Downside**.

EQUILIBRIUM DISCHARGE BOUNDARY HEAT SOURCE

This section has the same setting information as defined under Equilibrium Discharge Heat Source.

Magnetohydrodynamics

The **Magnetohydrodynamics** ( ) multiphysics coupling feature is used to model the interaction of magnetic fields and conducting fluids. It couples one of the magnetic field interfaces and the **Laminar Flow** interface. The supported magnetic field interfaces are the **Magnetic Fields**, **Magnetic and Electric Fields**, **Magnetic Field Formulation**, and **Rotating Machinery, Magnetic** interfaces in the AC/DC Module. By default, the coupling feature passes the Lorentz force $\mathbf{F} = \mathbf{J} \times \mathbf{B}$ from the magnetic field interface to the **Laminar Flow** interface and the electromotive force (the induced electric field) $\mathbf{E} = \mathbf{v} \times \mathbf{B}$ from the **Laminar Flow** interface to the magnetic field interface.

The **Magnetohydrodynamics** feature is available in 2D, 2D axisymmetric, and 3D geometries and is allowed for both Stationary and Time Dependent studies (including Frequency-Stationary and Frequency-Transient studies). When the deformation of the conducting fluids cannot be neglected (that is, when topology changes are important), the **Moving Mesh** feature (added from the **Definitions**) is usually used together with the **Magnetohydrodynamics** feature.

!

- The **Magnetohydrodynamics** feature cannot be used to model magnetizable fluids (ferrohydrodynamics). In magnetizable fluids, the magnetization forces can be much larger than the Lorentz force. In such cases, add a **Force Calculation** feature from the magnetic field interface and a **Volume Force** feature from the **Laminar Flow** interface. Note that the **Force Calculation** feature includes the force contribution not only from magnetic polarization but also from the Lorentz force, that is, the total electromagnetic force. Therefore, when the coupling between the magnetic field interface and **Laminar Flow** interface is implemented with the **Force Calculation** feature and the **Volume Force** feature, the **Magnetohydrodynamics** feature should not be used in the same domain.
- The difference between the **Magnetic and Electric Fields** interface and the **Magnetic Fields** interface is that the former solves the current conservation equation explicitly, while the latter assumes the conservation of current implicitly. In 3D magnetohydrodynamics modeling, the most general approach is to use the **Magnetic and Electric Fields** interface to solve the magnetic field. The **Magnetic Fields**, **Magnetic Field Formulation**, and **Rotating Machinery, Magnetic** interfaces can be useful in specific cases.

SETTINGS

The **Label** is the multiphysics coupling feature name. The default **Label** (for the first multiphysics coupling feature in the model) is `Magnetohydrodynamics 1`.

The **Name** is used primarily as a scope prefix for variables defined by the coupling node. Refer to such variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different coupling nodes or physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first multiphysics coupling feature in the model) is `mhd1`.

COUPLED INTERFACES

This section defines the physics involved in the multiphysics coupling. The **Electromagnetic** and **Fluid flow** lists include all applicable physics interfaces.

When the Magnetohydrodynamics node is added from the **Physics** ribbon (Windows users), **Physics** contextual toolbar (macOS and Linux users), or context menu (all users), then the first physics interface of each type in the component is selected as the default.

You can also select **None** from either list to uncouple the Magnetohydrodynamics node from a physics interface. If the physics interface is removed from the **Model Builder**, for example **Solid Mechanics** is deleted, then the list defaults to **None** as there is nothing to couple to.

If a physics interface is deleted and then added to the model again, then in order to reestablish the coupling, you need to choose the physics interface again from the lists. This is applicable to all multiphysics coupling nodes that would normally default to the once present interface. See Multiphysics Modeling Workflow in the *COMSOL Multiphysics Reference Manual*.

COUPLED SETTINGS

The coupling behavior can be easily controlled by selecting or clearing the **Include Lorentz force** and **Include electromotive force** check boxes. For example, if **Include Lorentz force** is selected and **Include electromotive force** is cleared, only the Lorentz force is considered and the electromotive force is neglected. This is usually valid when the magnetic Reynolds number is very low.

Static Current Density Component

This feature is obsolete and will be removed in future versions. Consider using the Magnetic and Electric Fields interface instead.

The **Static Current Density Component** multiphysics coupling () takes the induced current density defined in the Electric Currents interface

$$\mathbf{J} = \sigma\mathbf{E}$$

and maps it to the external current density defined in the Magnetic Fields interface.

SETTINGS

The **Label** is the default multiphysics coupling feature name.

The **Name** is used primarily as a scope prefix for variables defined by the coupling node. Refer to such variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different coupling nodes or physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first multiphysics coupling feature in the model) is `scdc1`.

STATIC CURRENT DENSITY COMPONENT

This section defines the physics involved in the Static Current Density Component multiphysics coupling. By default, the applicable physics interface is selected in the **From electric currents** list and is applied to the **To magnetic fields** interface to establish the coupling.

You can also select **None** from either list to uncouple the Static Current Density Component node from a physics interface. If the physics interface is removed from the **Model Builder**, for example **Electric Currents** is deleted, then the **From electric currents** list defaults to **None** as there is nothing to couple to.

> If a physics interface is deleted and then added to the model again, and in order to reestablish the coupling, you need to choose the physics interface again from the lists. This is applicable to all multiphysics coupling nodes that would normally default to the once present physics interface. See Multiphysics Modeling Workflow in the *COMSOL Multiphysics Reference Manual*.

Induction Current Density Component

> ! This feature is obsolete and will be removed in future versions. Consider using the Magnetic and Electric Fields interface instead.

The **Induction Current Density Component** multiphysics coupling ( ) takes the induced current density defined in the Magnetic Fields interface

$$\mathbf{J} = -\sigma\frac{\partial\mathbf{A}}{\partial t} \tag{7-4}$$

and maps it to the external current density defined in the Electric Currents interface.

SETTINGS

The **Label** is the default multiphysics coupling feature name.

The **Name** is used primarily as a scope prefix for variables defined by the coupling node. Refer to such variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different coupling nodes or physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first multiphysics coupling feature in the model) is `icdc1`.

INDUCTION CURRENT DENSITY COMPONENT

This section defines the physics involved in the Induction Current Density Component multiphysics coupling. By default, the applicable physics interface is selected in the **From magnetic fields** list and is applied to the **To electric currents** interface to establish the coupling.

You can also select **None** from either list to uncouple the Induction Current Density Component node from a physics interface. If the physics interface is removed from the **Model Builder**, for example **Magnetic Fields** is deleted, then the **From magnetic fields** list defaults to **None** as there is nothing to couple to.

If a physics interface is deleted and then added to the model again, and in order to reestablish the coupling, you need to choose the physics interface again from the lists. This is applicable to all multiphysics coupling nodes that would normally default to the once present physics interface. See Multiphysics Modeling Workflow in the *COMSOL Multiphysics Reference Manual.*

Lorentz Force

This feature is obsolete and will be removed in future versions. Consider using the Magnetohydrodynamics feature instead.

The **Lorentz Force** multiphysics coupling ( ) maps the Lorentz Force contribution defined in the Magnetic Fields interface to the volume force defined in the Laminar Flow interface.

SETTINGS

The **Label** is the default multiphysics coupling feature name.

The **Name** is used primarily as a scope prefix for variables defined by the coupling node. Refer to such variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different coupling nodes or physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first multiphysics coupling feature in the model) is `lf1`.

LORENTZ FORCE

This section defines the physics involved in the Lorentz Force multiphysics coupling. By default, the applicable physics interface is selected in the **Electromagnetic** list to apply the **Fluid flow** to its physics interface to establish the coupling.

You can also select **None** from either list to uncouple the Lorentz Force node from a physics interface. If the physics interface is removed from the **Model Builder**, for example **Magnetic Fields** is deleted, then the **Electromagnetic** list defaults to **None** as there is nothing to couple to.

Theory for the Equilibrium Discharges Interface

Equilibrium discharges (thermal plasmas) have nowadays a large range of industrial applications including cutting, welding, spraying, waste destruction, and surface treatment (Ref. 1).The current methods used for producing thermal plasmas generally employ high-intensity arcs, inductively coupled high-frequency discharges, or microwave discharges. The purpose of the Equilibrium Discharge interface is to help users model thermal plasmas generated by the first two methods (that is, arcs or inductively coupled discharges). Accordingly, the targeted applications are primarily DC and inductively coupled plasma torches as well as arc welding devices and circuit breakers.

Thermal plasmas are assumed to be under partial to complete local thermodynamic equilibrium (LTE) conditions (Ref. 1). At a macroscopic level, these kind of plasma can be considered as conductive fluid mixtures, which leads to the magnetohydrodynamics (MHD) equations. The latter combine the Navier–Stokes, heat, and Maxwell's equations to describe the motion of the conducting fluid in an electromagnetic field.

The Equilibrium Discharge interfaces are based on a set of assumptions that lead to simplifications of the MHD equations:

- The plasma is fully ionized (two-component mixture, that is, electrons and ions).
- The plasma is under local thermodynamic equilibrium (LTE) conditions (the electron temperature T_e is approximately equal to the heavy particle temperature, T_h, under low electric field at atmospheric pressure).
- The plasma is considered to be a locally neutral Newtonian fluid mixture.
- The plasma flow is laminar and quasi-incompressible (low Mach number).
- Viscous dissipation and pressure work in the energy equation are neglected.
- The displacement current is negligible (quasistatic approximation, that is, low frequencies).
- Magnetic diffusion dominates induction by fluid motion ($R_{em} << 1$).
- The plasma is optically thin.

The magnetic Reynolds number (R_{em}) gives the ratio of induction by fluid motion over the magnetic diffusion

$$R_{em} = \frac{|\nabla \times (\mathbf{u} \times \mathbf{B})|}{|\eta \nabla^2 \mathbf{B}|} \cong \frac{u_0 l_0}{\eta} \tag{7-5}$$

Where the magnetic diffusivity coefficient

$$\eta = (\mu_0 \sigma)^{-1}$$

is the inverse of the product of the plasma conductivity (σ) and vacuum permeability (μ_0). The values u_0 and l_0 are the velocity and length scale of the system under study, respectively.

DOMAIN EQUATIONS

Considering the assumptions stipulated above, the Equilibrium discharges can be described by a set of equations defined in the physics interfaces constituting the different multiphysics interfaces. Note that all the physics features available in the individual physics interfaces are also available to the multiphysics interface in use.

Energy Conservation (Heat Transfer in Fluids Interface and Equilibrium Discharge Heat Source Multiphysics Coupling Feature)

The energy conservation equation is used for all the Equilibrium Discharge interfaces.

$$\rho C_p\left(\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T\right) - \nabla \cdot (k \nabla T) = Q \tag{7-6}$$

The source Q (W/m^3) defined in the Equilibrium Discharge Heat Source multiphysics coupling feature includes three source/sink components:

- Resistive heating (ohmic heating)

$$Q = \mathbf{J} \cdot \mathbf{E} \tag{7-7}$$

The definition of the electric field differs following the physics interface in use.

- Equilibrium Discharges, Out-of-Plane Currents interface

$$\mathbf{E} = -\frac{\partial \mathbf{A}}{\partial t} \tag{7-8}$$

- Equilibrium Discharges, In-Plane Currents interface and Equilibrium Discharges interfaces

$$\mathbf{E} = -\left(\nabla V + \frac{\partial \mathbf{A}}{\partial t}\right) \tag{7-9}$$

- Volumetric net radiation loss Q_{rad} defined by the total volumetric emission coefficient, which is a material property from Ref. 1.
- Enthalpy transport (energy carried by the electric current)

$$\frac{\partial}{\partial T}\left(\frac{5k_BT}{2q}\right)(\nabla T \cdot \mathbf{J}) \tag{7-10}$$

Note that the enthalpy transport term prevails, for example, in the boundary layers close to electrodes in a fully ionized electric discharge, Ref. 1.

Momentum Conservation (Laminar Flow Interface)

The momentum conservation equation is used for the Equilibrium Inductively Coupled Plasma and Combined Inductive/DC Discharge interfaces.

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u}\right) = \nabla\cdot\left[-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla\cdot\mathbf{u})\mathbf{I}\right] + \mathbf{F} \tag{7-11}$$

Where μ is the dynamic viscosity of the fluid, **I** is the identity matrix and

$$\mathbf{F} = \mathbf{J}\times\mathbf{B}$$

the Lorentz force acting on the fluid defined in the magnetic field interface.

Ampere's Law and Current Conservation (Magnetic and Electric Fields Interface)

The Ampere's Law and Current conservation equations are used for the Equilibrium Discharges, In-Plane Currents interface and Equilibrium Discharges interfaces.

$$\nabla\cdot\mathbf{J} = 0 \tag{7-12}$$

$$\nabla\times\left(\frac{1}{\mu_r\mu_0}\nabla\times\mathbf{A}\right) = \mathbf{J} \tag{7-13}$$

Where the total current density is expressed as

$$\mathbf{J} = \sigma\left(-\nabla V - \frac{\partial \mathbf{A}}{\partial t} + \mathbf{u} \times \mathbf{B}\right) \tag{7-14}$$

Ampere's Law (Magnetic Fields Interface)

The Ampere's law equation is used for the Equilibrium Discharges, Out-of-Plane Currents interfaces. It does not solve the current conservation equation and the total current density is expressed as

$$\mathbf{J} = \sigma\left(-\frac{\partial \mathbf{A}}{\partial t} + \mathbf{u} \times \mathbf{B}\right) \tag{7-15}$$

TRANSPORT PROPERTIES

The above equations require specification of material properties, which are functions of temperature. The Equilibrium Discharge folder in the Material Browser contains properties for density, specific heat, viscosity, thermal conductivity and electrical conductivity as a function of temperature up to 24,000 K. Available gases include air, argon, helium, hydrogen, nitrogen, and oxygen. The data is taken from the tables in the Appendix of Ref. 1.

BOUNDARY EQUATIONS

The Equilibrium Discharge interfaces include by default cooling/heating of electrodes in contact with the equilibrium discharge Ref. 2.

Ion Bombardment and Thermionic Emission at the Cathode

Positive ions from the plasma are accelerated toward the cathode and generates heat at the electrode surface. As the electrode heats up, more electrons are emitted by thermionic emission, which cause cooling of the cathode. The associated cathode heat flux is defined as:

$$-\mathbf{n} \cdot (-k\nabla T) = -|\mathbf{J}_{\text{elec}}|\phi_s + |\mathbf{J}_{\text{ion}}|V_{\text{ion}} \tag{7-16}$$

Where k is the thermal conductivity (W/(m·K)), ϕ_s is the surface work function of the electrode (V), and V_{ion} is the ionization potential of the plasma (V). The ion current density norm is defined by

$$|\mathbf{J}_{\text{ion}}| = |\mathbf{J} \cdot \mathbf{n}| - |\mathbf{J}_{\text{elec}}| \tag{7-17}$$

where

$$|\mathbf{J}\cdot\mathbf{n}|$$

is the normal current density at the interface and where

$$|\mathbf{J}_{elec}| = \begin{cases} J_R(T), & |\mathbf{J}\cdot\mathbf{n}| > J_R(T) \\ |\mathbf{J}\cdot\mathbf{n}|, & |\mathbf{J}\cdot\mathbf{n}| \le J_R(T) \end{cases} \tag{7-18}$$

is the electron current density norm. The latter is defined by Richardson-Dushman current density if the total normal current density is larger than

$$J_R(T) = A_R T^2 \exp\left(-\frac{q\phi_{\mathrm{eff}}}{k_B T}\right) \tag{7-19}$$

Where A_R is the Richardson's constant (A/(m^2·K^2)), q is the electronic charge (C), k_B is the Boltzmann's constant (J/K), and ϕ_{eff} is the effective work function of the surface (V). Note that the ion current density norm $\mathbf{J}_{\mathrm{ion}} = 0$ if the Richardson-Dushman current density is larger than the total normal current at the interface.

Resistive Heating at the Anode

Electron entering the anode generates heat. Following the approach presented in Ref. 2, you can assume that there is no ion current and hence no ion heating at the anode. Accordingly the anode heat flux is defined as:

$$-\mathbf{n}\cdot(-k\nabla T) = |\mathbf{J}\cdot\mathbf{n}|\phi_s \tag{7-20}$$

Where

$$|\mathbf{J}\cdot\mathbf{n}|$$

is the normal current density at the interface, k is the thermal conductivity, and ϕ_{s} is the surface work function of the anode. Note that all the physics features available in the individual interfaces are also available to the multiphysics interface in use. This include, for instance, radiation heat losses, and wall boundary conditions.

References

1. M.I. Boulos, P. Fauchais, and E. Pfender, *Thermal Plasmas: Fundamentals and Applications*, Plenum Press, Springer, 1994.

2. J.J. Lowke, “A Unified Theory of Arcs and their Electrodes,” *J. Phys. IV France*, vol. 7, C4-283-C4-294, 1997

8

The Charge Transport Interface

This chapter describes the Charge Transport interface found under the **Plasma>Species Transport** branch ( ).

In this chapter:

The Charge Transport Interface

The **Charge Transport (ct)** interface ( ), found under the **Plasma >Species Transport** branch ( ), solves for the space charge density of one type of charge carrier in regard of the polarity (that is, positive or negative charge carriers). The charge carriers can be transported by migration in the electric field and convection. In the Corona interface the Charge Transport interface is coupled to the Electrostatics interface to model corona discharges.

When this physics interface is added, these default nodes are also added to the **Model Builder** — **Transport Properties**, **No Flux** (the default boundary condition), and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions. You can also right-click **Charge Transport** to select physics features from the context menu.

For details about the Corona Discharge interface, see The Corona Discharge Interface.

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `ct`.

TRANSPORT MECHANISMS

Select the **Convection** check box to add convection to the default migration in the electric field transport mechanism. The selection adds the section Convection on the **Transport Properties** node.

CONSISTENT STABILIZATION

To display this section, click the **Show** button ( ) and select **Stabilization**. There are two consistent stabilization methods available — **Streamline Diffusion** and **Crosswind diffusion**. By default both methods are selected. Streamline and crosswind diffusion stabilization adds artificially diffusion in the direction of the streamlines and in the direction perpendicular to the streamlines, respectively, to stabilize the numeric solution of a migration/convection-dominated PDE.

When the **Crosswind diffusion** check box is selected, a weak term that reduces spurious oscillations is added to the transport equation. The resulting equation system is always nonlinear.

INCONSISTENT STABILIZATION

To display this section, click the **Show** button (👁) and select **Stabilization**. This method is equivalent to adding an isotropic diffusion term to the equations in order to dampen the effect of oscillations by making the system somewhat less dominated by convection. By default, the **Isotropic diffusion** check box is not selected, because this type of stabilization adds artificial isotropic diffusion and affects the accuracy of the original problem. If required, select the **Isotropic diffusion** check box and enter a **Tuning parameter** as a scalar positive value. The default value is 0.25. A higher value adds more isotropic diffusion

DISCRETIZATION

To display all settings available in this section, click the **Show** button (👁) and select **Advanced Physics Options.** The space charge density variable is set to use Linear elements by default. The Value type when using splitting of complex variable is set to Complex by default.

DEPENDENT VARIABLES

The dependent variables (field variables) is the **Space Charge Density**. The name can be changed but the names of fields and dependent variables must be unique within a model.

- Domain, Boundary, and Pair Nodes for the Charge Transport Interface
- Theory for the Charge Transport Interface

Positive and Negative Corona Discharges: Application Library path **Plasma_Module/Corona_Discharges/positive_and_negative_corona_discharge s**

Domain, Boundary, and Pair Nodes for the Charge Transport Interface

The Charge Transport Interface has following domain and boundary features available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or right-click to access the context menu (all users).

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

In the *COMSOL Multiphysics Reference Manual* see Table 2-4 for links to common sections and Table 2-5 to common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

- Transport Properties
- Initial Values
- No Flux
- Source
- Space Charge Density

Transport Properties

Use the **Transport Properties** node to set the quantities defining the drift and convection transport mechanisms: model inputs, electric potential, reduced ion mobility, charge number, and velocity field.

MODEL INPUTS

This section has fields and values that are inputs to expressions that define material properties. If such user-defined property groups are added, the model inputs display here.

- Enter a **Temperature** T (SI unit: K) to compute the gas density.
- Enter an **Absolute pressure** p_A (SI unit: Pa) to compute the gas density.

The gas density is only used to compute the ion mobility from the given **Reduced ion mobility.**

MIGRATION IN THE ELECTRIC FIELD

This section defines quantities used to compute the drift component of the current.

- Enter a **Electric potential** V (SI unit: V) used to compute the electric field which results in migration of the ions.
- Enter a **Reduced ion mobility** $\mu_i N$ (SI unit: 1/(V.m.s)) used to compute the ion mobility.
- Enter a **Charge number** Z_q(dimensionless).

CONVECTION

This section is available only when the **Convection** check box in the **Transport Mechanisms** property is selected. Enter the **Velocity field u** or in the case of multiphysics coupling select the interface from which the velocity should come.

Initial Values

The **Initial Values** node adds initial values for the space charge density.

INITIAL VALUES

Enter values or expressions for the initial value of the **Space charge density** ρ (SI unit: C/m^3). Enter an approximate guess for the space charge density in the volume and be coherent with the sign of the charge carriers being solved for. As an example, if the **Charge number** in Migration in the electric field is -1 enter a negative number for the initial value of the space charge density.

No Flux

Use the **No Flux** boundary condition to sets the normal component of the current density of the charge carriers to zero

$$-\mathbf{n}\cdot\mathbf{J} = 0$$

Source

Use the **Source** to add a volume current source.

SOURCE

Select an option from the **Specify source using** list — **Current source** (the default) or **Reaction rate**.

When **Current source** is selected enter a value or expression for the **Current source** S (SI unit: A/m^3). When **Reaction rate** is selected enter a value or expression for the **Reaction rate** R (SI unit: $mol/(m^3.s)$). The actual current source is then computed by multiplying the reaction rate by the elementary charge and the Avogadro's Number.

Space Charge Density

The **Space charge density** boundary condition fixes the space charge density to a specific value.

$$\rho = \rho_0 \text{ or} \rho = n_0 e$$

SPACE CHARGE DENSITY

Select an option from the **Specify space charge density using** list — **Space charge density** (the default) or **Number density**.

When **Space charge density** is selected enter a value or expression for the **Space charge density** ρ_0 (SI unit: C/m^3). When **Number density** is selected enter a value or expression for the **Number density** n_0 (SI unit: $1/m^3$). The actual space charge density is then computed by multiplying the number density by the elementary charge.

Theory for the Charge Transport Interface

The Charge Transport Interface theory is described in this section:

- Charge Transport Theory
- Current Source Coefficient
- No Flux Boundary Condition
- Space Charge Density Boundary Condition
- Reference for the Charge Transport Interface

Charge Transport Theory

The Charge Transport interface solves for the space charge density of one type of charge carrier (that is, positive or negative ions) in the presence of electric fields and fluid convection. This interface is design to describe the ion density in the transport region of a corona discharge. The Corona Discharge interface consists of the coupling of the Charge Transport and the Electrostatics interfaces where the electric field and potential at the corona electrode can be set using the Electrode multiphysics coupling feature. The imposition of the electric field and potential at a boundary requires a special formulation of the charge transport equations described below. The charge conservation, current density, and Poisson's equations are the starting point

$$\nabla \bullet \mathbf{J} = S \tag{8-1}$$

$$\mathbf{J} = Z_q \mu_i \rho \mathbf{E} + \rho \mathbf{u} \tag{8-2}$$

$$\varepsilon_0 \nabla^2 V = -\rho\,. \tag{8-3}$$

This set of equations is manipulated to obtain the equation used to describe the space charge density

$$\mu_i\left(\frac{\rho^2}{\varepsilon_0} - \nabla V \bullet \nabla \rho\right) + \rho \nabla V \bullet \nabla \mu_i + \nabla \rho \bullet \mathbf{u} = S\,. \tag{8-4}$$

where:

- $\mathbf{J}$ is the current density (SI unit: $\mathrm{C/m^2}$).
- ρ is the space charge density (SI unit: $\mathrm{C/m^3}$).
- S is the current source (SI unit: $\mathrm{C/m^3}$).
- Z_q is the charge number (SI unit: 1).
- $\mathbf{E}$ is the electric field (SI unit: V/m).
- V is the electric field (SI unit: V/m).
- μ_i is the ion mobility (SI unit: $\mathrm{m^2/(V.s)}$).
- $\mathbf{u}$ is the neutral fluid velocity vector (SI unit: m/s). This usually comes from a Navier-Stokes interface.

Current Source Coefficient

A current source can be added to described volume reactions that result in the creation of destruction of charge carriers. The default option for the source term is to define a current source directly. There is however the option to define a current source using a reaction rate. This approach is more meaningfully when dealing with corona discharges and the reaction rate can be estimated using the Plasma interface.

No Flux Boundary Condition

The No Flux boundary condition simply sets the normal component of the current density of the charge carriers to zero

$$-\mathbf{n}\cdot\mathbf{J} = 0$$

Space Charge Density Boundary Condition

The Space Charge Density boundary condition fixes the space charge density to a specific value.

$$\rho = \rho_0 \ \mathrm{or}\, \rho = n_0 e$$

Reference for the Charge Transport Interface

1. D. Cagnoni , F. Agostini, T. Christen, N. Parolini, I. Stevanovic, and C. de Falco, "Multiphysics Simulation of Corona Discharges Induced Ionic Wind", *Journal of Applied Physics*, vol. 114, pp. 233301-10, (2013).

2. K. Adamiak and P. Atten "Simulation of Corona Discharges in Point-Plane Configuration", *Journal of Electrostatics*, vol. 61, pp. 85-98, (2004).

9

The Corona Discharge Interface

This chapter describes the Corona Discharge interface found in the **Plasma>Electric Discharges** branch ( ). The Corona Discharge interface can be used to model the transport region of a corona discharge and obtain the space charge density, electric potential, ion current, and the electrohydrodynamic force.

In this chapter:

The Corona Discharge Interface

The **Corona Discharge** interface ( ) is used to model the transport region of a corona discharge where only one type of charge carrier is present (that is, positive or negative ions depending on the polarity of the corona).

When a predefined **Corona Discharge** interface is added from the **Plasma>Electric Discharges** branch ( ) of the **Model Wizard** or **Add Physics** windows, **Electrostatics** and **Charge Transport** interfaces are added to the Model Builder. In addition, the **Multiphysics Couplings** node is added, which automatically includes the multiphysics coupling features **Space Charge Density Coupling** and **Potential Coupling**. The **Electrode** multiphysics feature is not added by default but this feature is fundamental in modeling the corona discharge.

When the **Charge Transport** interface ( ) is added, these default nodes are also added to the **Model Builder** — **Transport Properties**, **No Flux**, and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, a source terms and boundary conditions.

When the **Electrostatics** interface ( ) is added, these default nodes are also added to the **Model Builder** — **Charge Conservation**, **Zero Charge**, and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, a ground electrode.

For details about predefined multiphysics couplings, see Multiphysics Modeling Workflow in the *COMSOL Multiphysics Reference Manual*.

Positive and Negative Corona Discharges: Application Library path **Plasma_Module/Corona_Discharges/positive_and_negative_corona_discharges**

Electrostatic Precipitator: Application Library path **Plasma_Module/Corona_Discharges/electrostatic_precipitator**

SETTINGS

The settings for the **Charge Transport** interface can be found in The Charge Transport Interface section. The settings for the **Electrostatics** interface can be found in The Electrostatics Interface in the *AC/DC Module User's Guide*.

These nodes are described in this section:

- Space Charge Density Coupling
- Potential Coupling
- Electrode

Space Charge Density Coupling

The **Space Charge Density Coupling** multiphysics node ( )adds the space charge density computed in the Charge Transport interface to Poisson's Equation in the Electrostatics interface.

Potential Coupling

The **Potential Coupling** multiphysics node ( ) applies the potential dependent variable in the Electrostatics interface into the migration term of the Charge Transport interface.

Electrode

The **Electrode** multiphysics node ( ) sets the electric field and an electric potential at a given boundary defined in the selection of this feature. The normal component of the electric field is set to a given value E_0 at the boundary selection

$$\mathbf{n} \bullet \mathbf{E} = E_0 \tag{9-1}$$

and is used as a boundary condition to Poisson's equation solved in the Electrostatics interface. The charge density from the Charge Transport interface at the boundary selection is found using a Lagrange multiplier to verify

$$V - V_0 = 0 \tag{9-2}$$

where V_0 is a given electric potential.

ELECTRIC POTENTIAL

This is the value of the electric potential to be applied in the selected boundary. In the **Electric potential** V_0 (SI unit: V) field enter a value or expressions for the electric potential.

ELECTRIC FIELD

Select an option from the **Electrode electric field** list — **Peek's Law** (the default) or **User defined**. When **Peek's Law** is selected this section has the following fields that are used to compute the electric field using Peek's Law:

- Enter a **Disruptive electric field** E_c (SI unit: V/m).
- Enter an **Empirical constant** c (SI unit: $\mathrm{m}^{1/2}$).
- Enter an **Electrode radius** r_c (SI unit: m).

If **Electrode electric field** is set to **User defined** enter a value or expression for the **Electric field** E_0 (SI unit: V/m).

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

If a physics interface is deleted and then added to the model again, and in order to reestablish the coupling, you need to choose the physics interface again from the lists. This is applicable to all multiphysics coupling nodes that would normally default to the once present physics interface. See Multiphysics Modeling Workflow in the *COMSOL Multiphysics Reference Manual*.

Corona Discharges Theory

Coronas are weakly luminous self-sustained discharges, which appears near sharp points, edges, or thin wires where the electric field is much more intense than in the rest of the gap Ref. 1. The appearance of corona discharges is explained by physical mechanisms of electron creation in the region of intense electric field where ionization occurs. The creation mechanism of electrons depends on the polarity of the corona electrode. In the case the electrode is a cathode the corona is called *negative corona* and the discharge ignition is analogous to a Townsend breakdown due to secondary emission from the cathode

$$\int_0^{x_m} [\alpha(x) - \eta(x)]dx = \ln\left(1 + \frac{1}{\gamma}\right) \tag{9-3}$$

where α and η are the ionization and attachment Townsend coefficients, γ is the secondary electron emission coefficient, and the integration is between the cathode and the positions x_m where $\alpha(x_m)=\eta(x_m)$. In the case that the electrode is an anode the corona is called *positive corona* and the ignition mechanism is related to cathode-directed streamers described by the generalized Meek breakdown criterion

$$\int_0^{x_m} [\alpha(x) - \eta(x)]dx = 18 - 20\,. \tag{9-4}$$

However the value of the integral in Equation 9-3 is about two times smaller than the one in Equation 9-4, the value of the breakdown electric field is only slight different because of the steep dependence of the Townsend coefficients with the electric field.

Equation 9-3 and Equation 9-4 can be used to find a lower limit for the electric field at the corona electrode above which there is gas breakdown. However, instead of solving the integrals above a common approach to estimate the breakdown electric field in corona discharges is to use empirical expressions of the type proposed by Peek. For coaxial type coronas, Peek proposed the following expression for the breakdown in air

$$E_0 = 3.1\times10^6\delta\left(1 + \frac{0.0308}{\sqrt{\delta r_i}}\right) \tag{9-5}$$

where E_0 (SI unit: V/m) is the breakdown electric field, δ is the gas number density normalized to the gas density at 760 Torr and 293.15 K, and r_i is the radius of the corona electrode. This expression can be adapted to other gases and geometries by adjusting the numerical coefficients.

The creation of charge carriers occurs only in a region very close to the corona electrode where the electric fields are very intense. In the remainder of the gap, charged carriers are transported in a weak electric field thus closing the circuit. In the transport region the charge carriers are positive ions in the positive corona and negative ions in the negative corona. If the gas is only weekly electronegative there can be electrons as well.

The physics of coronas ignition and maintenance is rather complicated to describe numerical since it involves cathode fall, streamers, and photo-ionization. The model described in this section aims to bypass all this complex physics by providing suitable boundary conditions. In practice, a potential and an electric field are imposed at the corona electrode and the model uses a charge conservation equation to describe the transport region of a corona where only one type of charge carrier exits (that is, positive or negative ions).

In this type of models, to justify a boundary condition where an electric field and a potential are imposed at the corona electrode, it is used Kaptsov's hypothesis that says that the electric field at the corona electrode after the discharge breakdown is independent of the applied voltage and stays fixed at that value Ref. 2.

DOMAIN EQUATIONS

The model is based on the conservation of current transported by the charged carriers. It should be emphasized that the model is not self-consistent in the sense that both potential and the electric field need to be given at the corona electrode. In other words, the electric field necessary to sustain the discharge is not obtained from first principles: electron and ion transport, electrons gaining energy from the electric field, and electrons losing energy in collisions with the background gas.

The space charge density and the electrostatic potential are solved using the charge conservation, current density, and Poisson's equations Ref. 2

$$\nabla \bullet \mathbf{J} = S \tag{9-6}$$

$$\mathbf{J} = Z_q \mu_i \rho \mathbf{E} + \rho \mathbf{u} \tag{9-7}$$

$$\varepsilon_0 \nabla^2 V = -\rho \tag{9-8}$$

where:

- $\mathbf{J}$ is the current density (SI unit: C/m^2).
- ρ is the space charge density (SI unit: C/m^3).
- S is the current source (SI unit: C/m^3).
- Z_q is the charge number (SI unit: 1).
- $\mathbf{E}$ is the electric field (SI unit: V/m).
- V is the electric field (SI unit: V/m).
- μ_i is the ion mobility (SI unit: m^2/(V.s)).
- $\mathbf{u}$ is the neutral fluid velocity vector (SI unit: m/s).

Equation 9-6, Equation 9-7, and Equation 9-8 are manipulated to obtained a single equation for the space charge density

$$\mu_i\left(\frac{\rho^2}{\varepsilon_0} - \nabla V \bullet \nabla\rho\right) + \rho\nabla V \bullet \nabla\mu_i + \nabla\rho \bullet \mathbf{u} = S \qquad (9\text{-}9)$$

that is solved coupled with Poisson's equation.

BOUNDARY CONDITIONS

In the corona model presented here, as discussed above, the boundary conditions at the corona electrode are unusual and consists of giving an electric field and a potential at the corona electrode. The normal component of the electric field at the corona electrode is used as a boundary condition for Poisson's equation

$$\mathbf{n} \bullet \mathbf{E} = E_0 \,. \qquad (9\text{-}10)$$

The boundary condition for Equation 9-9 involves in finding the space charge density at the corona electrode, using a Lagrange multiplier, so that the imposed potential V_0 is verified

$$V - V_0 = 0 \,. \qquad (9\text{-}11)$$

Since the field and potential are not found self-consistently their values need to have physical meaning in order to represent a corona discharge.

When changing the polarity of the corona make sure that the electric field and potential at the electrode (defined in the Electrode multiphysics coupling), the sign of the charge number of the charge carriers (defined in the Transport Properties feature in the Charge Transport interface) and the sign of the initial value of the space charge density (defined in the Initial Values feature of the Charge Transport interface) are coherent.

References

1. Y.P. Raizer, *Gas Discharges Physics*, Springer, 1997.

2. K. Adamiak and P. Atten "Simulation of Corona Discharges in Point-Plane Configuration", *J. Electrostat.*, vol. 61, pp. 85–98, 2004.

10

The Electrical Breakdown Detection Interface

This chapter describes the underlying theory for the Electrical Breakdown Detection interface, found under the **Plasma>Electric Discharges** branch ( ).

In this chapter:

The Electrical Breakdown Detection Interface

The **Electrical Breakdown Detection (ebd)** interface ( ), found under the **Plasma>Electric Discharges** branch ( ), estimates the likelihood that electrical breakdown will occur between electrically excited and grounded surfaces. The electric field is typically computed via an Electrostatics or Electric Currents interface in a separate study. This interface integrates the Townsend growth coefficient along electric field lines by releasing tracer particles from cathode surfaces.

When this physics interface is added, these default nodes are also added to the **Model Builder**: **Electrical Breakdown Detection** and **Wall** (the default boundary condition). Note that no initial values are required in this physics interface. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions for the cathode. You can also right-click the **Electrical Breakdown Detection** node to select physics features from the context menu. As a minimum requirement, a **Cathode** node must be added to the model, and at least one boundary must be selected for it.

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `ebd`.

ADDITIONAL VARIABLES

To store additional information about the trajectories of the test particles as they follow the electric field lines, activate **Store particle status data**, which creates additional postprocessing variables containing information about whether the test particles have reached a target surface, or are still in the modeling domain. The option **Store particle release statistics** can be used to keep track of which test particles were released by a particular **Cathode** feature.

DEPENDENT VARIABLES

The dependent variables (field variables) are the **Particle Position** which consists of 3 components (in 3D). The name can be changed but the names of fields and dependent variables must be unique within a model.

Electrical Breakdown Between Spheres: Application Library path **Plasma_Module/Electrical_Breakdown_Detection/breakdown_between_spheres**

Domain, Boundary, and Pair Nodes for the Electrical Breakdown Detection Interface

The Electrical Breakdown Detection Interface has these domain, boundary, and pair nodes, listed in alphabetical order, available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (macOS or Linux users), or right-click to access the context menu (all users).

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

In the *COMSOL Multiphysics Reference Manual* see Table 2-4 for links to common sections and Table 2-5 to common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

- Cathode
- Nonlocal Accumulator (subnode to **Cathode**)
- Wall
- Accumulator (subnode to **Wall**)
- Electrical Breakdown Detection
- Particle Counter

Cathode

Use the **Cathode** node to determine the boundaries from which the test particles should be released. This boundary should typically have a negative potential with respect to the grounded boundaries in the system.

CATHODE PROPERTIES

Enter values or expressions for each of the following:

- **Secondary emission coefficient** γ_i (dimensionless). This is used to determine the threshold for whether the integrated Townsend coefficient along streamlines exceeds a specific value. The higher the value, the more likely electrical breakdown will occur. The default is 0.07, and values are typically between 0.001 and 0.4.
- **Minimum field for sustained propagation** E_{min} (SI unit: V/m). Sometimes the field on the electrode needs to exceed a certain quantity in order for a sustained discharge to occur. If that is not the case in a specific model, this setting can be ignored.
- **Refinement factor**. This factor determines how many particles are to be released per mesh element on the surface mesh. The default means that one particle will be released per surface mesh element. This value should not need to be changed except in rare cases, where the mesh is very coarse and there are obstructions between the cathode and grounded surfaces.

INITIAL VALUE OF AUXILIARY DEPENDENT VARIABLES

If there are auxiliary dependent variables included, then this section will be populated with text fields to specify initial conditions for those variables.

Nonlocal Accumulator

Use the **Nonlocal Accumulator** subnode to communicate information from a particle's current position to the surface from which it was released.

The subnode is available from the context menu (right-click the Cathode parent node) or from the **Physics** toolbar, **Attributes** menu.

Each **Nonlocal accumulator** subnode defines a variable, called the accumulated variable, that is computed using variables defined on particles released by the parent **Inlet** node or on domains and boundaries encountered by such particles.

ACCUMULATOR SETTINGS

Select an **Accumulator type**: **Density** (default) or **Count**.

- For **Density** the accumulated variable is divided by the volume of the mesh element where it is defined.
- For **Count** the accumulated variable is unaffected by the element size.

Select an option from the **Accumulate over** list: **Elements** (default) or **Elements and time**.

- For **Elements** the accumulated variable is proportional to the instantaneous value of the **Source** term R for all applicable particles.
- For **Elements and time** the time derivative of the accumulated variable is proportional to the instantaneous value of the **Source** term R for all applicable particles, and thus the accumulated variable considers the time history of particles in the modeling domain instead of just their current values.

Enter the **Accumulated variable name**. The default is `rpi`.

Enter a **Source** R. The unit of the source depends on the settings in the **Units** section. The source term is used to calculate the accumulated variable in a manner specified by the **Accumulate over** and **Accumulator type** settings.

Select a **Source geometric entity level**: **Domains**, **Boundaries**, or **Domains and boundaries**.

- If **Domains** is selected, particles only contribute to the accumulated variable on their releasing surface if they are still active; that is, they are still propagating through a domain.
- If **Boundaries** is selected, the particles only contribute to the accumulated variable if they have become stuck or frozen to a boundary somewhere in the model.
- If **Domains and boundaries** is selected, all of the active, stuck, and frozen particles released by a feature can contribute to the accumulated variable.

UNITS

Select a **Dependent variable quantity** from the list; the default is **Dimensionless [1]**. To enter a unit, select **None** from the list and in the **Unit** field enter a value, for example, `K`, `m/s`, or `mol/m^3`.

SMOOTHING

The accumulated variable is piecewise discontinuous across different boundary elements. Select the **Compute smoothed accumulated variable** check box to define a spatially smoothed accumulated variable; at each point on the boundary, the smoothed accumulated variable is the average of the accumulated variable over a circle. Enter a **Smoothing radius** r (SI unit: m). The default is 0.1 m.

Wall

The **Wall** is the default boundary condition on all exterior boundaries. This boundary condition stops particles that hit a surface. By default, the **Freeze** condition is used.

Choose any of the following from the **Wall condition** list. This affects the type of information that is accessible while evaluating and visualizing the results.

- **Freeze** (the default): all information about the particles can be evaluated during postprocessing.
- **Stick**: only the particle position can be evaluated during postprocessing. Other particle-based quantities, such as auxiliary variables, will be evaluated as zero.
- **Disappear**: the particle is removed from the geometry when it hits a surface. Any quantity that is evaluated on the particle after this time will evaluate to not-a-number (NaN).

Select one of the following from the **Primary particle condition** list: **None** (the default), **Probability**, or **Expression**. For **Probability** or **Expression**, select an option from the **Otherwise** list: **Freeze** (the default), **Stick**, or **Disappear**.

- For **Probability**, enter a value or expression for the **Probability** γ (dimensionless). The default is 0. For each particle that hits the wall, there is a probability of γ that it will be affected by the **Wall** condition, and a probability of $1 - \gamma$ that it will be affected by the behavior chosen from the **Otherwise** list.
- For **Expression**, enter a value or expression for the **Evaluation expression** e (dimensionless). The default is 1. For each particle that hits the wall, the **Wall condition** will be used if the expression has a nonzero value (here nonzero is considered “true”) or will be affected by the **Otherwise** setting if the value is zero.

Accumulator

The **Accumulator** subnode is available from the context menu (right-click the Wall parent node) or from the **Physics** toolbar, **Attributes** menu. Each **Accumulator** subnode defines a variable, called the accumulated variable, on each boundary element in the selection of the parent node. Whenever a particle hits a boundary element, the value of the accumulated variable in that element is incremented based on the value of the user-defined **Source** term R for the incident particle.

ACCUMULATOR SETTINGS

Select an option from the **Accumulator type** list: **Density** (default) or **Count**.

- For **Density** the accumulated variable is divided by the surface area (in 3D) or length (in 2D) of the boundary element where it is defined.
- For **Count** the accumulated variable is the sum of the source terms of all particles that hit the boundary element, and is unaffected by the boundary element size.

Select an option from the **Accumulate over** list: **Particle-wall interactions** (default) or **Particles in boundary elements**.

- For **Particle-wall interactions** the accumulated variable is affected by all particles that hit the boundary element, including those that pass through or are reflected by the boundary.
- For **Particles in boundary elements** the accumulated variable is only affected by particles that freeze or stick to the boundary element.

Enter the **Accumulated variable name**. The default is `rpb`. The accumulated variable is defined as `<scope>.<name>`, where `<scope>` includes the name of the physics interface node, parent boundary condition, and the **Accumulator** node; and `<name>` is the accumulated variable name.

For example, if the **Accumulator** subnode is added to a Wall node in an instance of the Mathematical Particle Tracing interface using the default variable name `rpb`, the accumulated variable name might be `pt.wall1.bacc1.rpb`.

Enter a **Source** R. The unit of the source term depends on the settings in the **Units** section. Whenever a particle collides with a boundary element in the selection of the parent node, the accumulated variable in that element is incremented by the source term. If the **Accumulator type** is set to **Density**, the source term is divided by the area of the boundary element (in 3D) or the length of the boundary element (in 2D).

UNITS

Select a **Dependent variable quantity** from the list; the default is **Dimensionless [1]**. To enter a unit, select **None** from the list and in the **Unit** field enter a value, for example, `K`, `m/s`, or `mol/m^3`.

SMOOTHING

The accumulated variables are computed using discontinuous shape functions. Select the **Compute smoothed accumulated variable** check box to compute a smoothed accumulated variable by computing the average value of the variable within a sphere of a user-defined radius. Then enter a **Smoothing radius** r (SI unit: m). The default is 0.1 m.

Electrical Breakdown Detection

The **Electrical Breakdown Detection** node allows for specification of the background gas properties, through which the test particles will propagate. This included information about the pressure, the Townsend coefficient data and the electric field.

DOMAIN PROPERTIES

Choose how to specify the number density using the **Number density specification** list. Select either:

- **Specify pressure and temperature** (the default). Enter values for the **Pressure** and **Temperature**. The ideal gas law is then used to calculate the background number density.
- **Specify number density and temperature**. In this case, enter a value for the **Background number density** and the **Temperature** is then used to compute the background pressure, which is often needed during postprocessing.

TOWNSEND COEFFICIENT

This section is used to define how the Townsend coefficient should be specified. Choose between:

- **Dry air**. When this option is selected, the Townsend coefficient is specified via an interpolation function from experimental data, Ref. 2 and references therein.
- **Lookup table**. In this case, specify two columns of data, the first column should be the **Reduced electric field** (SI unit: V m^2, default unit: Td) and the second column the **Townsend coefficient data** (SI unit: m^2). When this option is chosen, a **Units** section appears below where the preferred unit can be selected. This data can often be found online, or can be generated by solving the Boltzmann equation under the two–term approximation.
- **Coefficients**. This allows for specification of a **Townsend coefficient prefactor** and a **Townsend coefficient exponential factor**. In the literature, the Townsend coefficient prefactor is usually given in units of 1/(cm torr), whereas the requested unit is s^2/kg. Both units correspond to the same physical quantity, so it is acceptable to enter, `7.79[1/(cm*torr)]` for the prefactor, which is also the default. Similarly, for the exponential factor, the literature reports in units of V/(cm torr), whereas the requested unit is m^2/(s A). Again, both units correspond to the same physical

quantity, so the unit reported in the literature can be appended to the numeric values entered.

- **User defined.** In this case, enter a value or expression for the **Townsend growth coefficient**. The default is `1E-22[m^2]`, but it should typically be a function of the reduced electric field.

ELECTRIC FIELD

Specify the electric field either in terms of an **Electric potential** or **Electric field**. These should typically come from an Electrostatics or Electric Currents interface, and computed beforehand in a separate study. Also specify the **Maximum electric field** (SI unit: V/m). This value is used to scale the electric field when solving, which can make the list of times to solver for easier to specify.

Particle Counter

Use the **Particle Counter** feature to compute information about particles that are located in a set of selected domains or on a set of selected boundaries. The counter may detect all particles or only the particles released by a specified release feature. Computed variables are the number of particles transmitted, the number of particles transmitted at the final time, the transmission probability, and a logical expression which can be used to filter the rendered particles during results processing.

PARTICLE COUNTER

Select an option from the **Release feature** list. If **All** (the default) is selected, the **Particle Counter** collects information about all particles in the selected domains or boundaries, regardless of how they were released. Alternatively, select a particle release feature from the list, and then only the particles produced by that feature are counted.

Theory for the Electrical Breakdown Interface

The Electrical Breakdown Detection interface defines 3 different regimes that can occur in any given device. The breakdown condition for a self-sustaining discharge is given by the following:

$$\gamma_i\left(\exp\left(\int_0^D N\alpha ds\right)-1\right) = 1 \tag{10-1}$$

where γ_i is the secondary emission coefficient (dimensionless), N is the number density (SI unit: $1/\mathrm{m}^3$), α is the reduced Townsend growth or decay coefficient (SI unit: m^2), s is the arc length along the particle trajectory, and D is the distance from the source boundary to any destination boundary. Using this, the following 3 regimes are defined.

NO DISCHARGE

Rearranging Equation 10-1, it is obvious that no discharge will occur if the following condition is true:

$$\int_0^D N\alpha ds < \ln(1+1/\gamma_i)\,.$$

SUSTAINED DISCHARGE

When the left side of Equation 10-1 is greater than 1, a self-sustaining discharge can occur. Another way of writing this condition is that a self-sustained discharge can form when the Townsend condition is met:

$$\int_0^D N\alpha ds > \ln(1+1/\gamma_i)\,.$$

This is not necessarily catastrophic to an electrical design, since the current is usually limited in such a discharge. The third case, however, can be catastrophic.

STREAMER

When the exponential of the left side is above around 10^8, a streamer will form across the gap. Mathematically, the streamer condition is given by:

$$\int_0^D N\alpha ds > 17.7 + \ln(d/(1[\mathrm{cm}]))$$

where d is the gap distance in cm.

The Electrical Breakdown Detection physics interface defines a variable, `ebd.bi`, which takes the value of 0 for the no-discharge case, 1 for the sustained discharge, and 2 for the streamer. This variable is plotted by default when running a study.

All the information about in which regime the system will operate is embedded in the reduced Townsend coefficient, α. The reduced Townsend coefficient is a strong function of the reduced electric field:

$$\alpha = \alpha\left(\frac{E}{N}\right)$$

where $\mathbf{E}$ is the electric field parallel to the streamlines (SI unit: V/m). The software computes the integral by solving an ordinary differential equation along the test particle trajectories:

$$\frac{d\alpha_D}{ds} = \alpha\left(\frac{N}{N_{\mathrm{stp}}}\right)$$

where N_{stp} is the number density at standard temperature and pressure. Another quantity of interest is the pressure multiplied by the path length. This is also computed by solving the following ordinary differential equation:

$$\left(\frac{1}{p}\right)\frac{d}{ds}(p_D) = 1$$

where it is assumed the pressure is constant along the trajectory.

References

1. Larry K. Warne, Roy E. Jorgenson, and Scott D. Nicolaysen, *Ionization Coefficient Approach to Modeling Breakdown in Nonuniform Geometries*, Sandia Report (2003).

2. J. Dutton, *A survey of Electron Swarm Data*, J. Phys. Chem. Ref. Data, Vol 4, No 3, 1975.

3. J.M. Meek, J.D. Craggs, *Electrical Breakdown of Gases* (1953).

11

AC/DC Interfaces

This chapter describes the physics interfaces found under the **AC/DC** branch (⚡).

In this chapter:

- The Electrostatics Interface
- The Electric Currents Interface
- The Electrical Circuit Interface
- Theory for the Electrostatics Interface
- Theory for the Electric Currents Interface
- Theory for the Electrical Circuit Interface
- Connecting to Electrical Circuits
- SPICE Import and Export

The Electrostatics Interface

The **Electrostatics (es)** interface (⚡), found under the **AC/DC>Electric Fields and Currents** branch when adding a physics interface, is used to compute the electric field, electric displacement field, and potential distributions in dielectrics under conditions where the electric charge distribution is explicitly prescribed. The formulation is stationary except for when it is used together with other physics interfaces. Eigenfrequency, frequency-domain, small-signal analysis, and time-domain modeling are supported in all space dimensions.

The physics interface solves Gauss' law for the electric field using the scalar electric potential as the dependent variable.

Charge Conservation is the main node, which adds the equation for the electric potential and has a Settings window for defining the constitutive relation for the electric displacement field and its associated properties such as the relative permittivity.

When this physics interface is added, these default nodes are also added to the **Model Builder** — **Charge Conservation**, **Zero Charge** (the default boundary condition), and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions and space charges. You can also right-click **Electrostatics** to select physics features from the context menu.

Physics-Controlled Mesh

The physics-controlled mesh is controlled from the **Mesh** node's **Settings** window (if the **Sequence type** is **Physics-controlled mesh**). There, in the table in the **Physics-Controlled Mesh** section, find the physics interface in the **Contributor** column and select or clear the check box in the **Use** column on the same table row for enabling (the default) or disabling contributions from the physics interface to the physics-controlled mesh.

Information from the physics, such as the presence of an infinite elements domain or periodic condition, will be used to automatically set up an appropriate meshing sequence.

!

In the *COMSOL Multiphysics Reference Manual* see the Physics-Controlled Mesh section for more information about how to define the physics-controlled mesh.

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `es`.

CROSS-SECTION AREA (1D COMPONENTS)

For 1D components, enter a default value for the **Cross-section area** A (SI unit: m^2). The default value of 1 is typically not representative for a thin domain. Instead it describes a unit thickness that makes the 1D equation identical to the equation used for 3D components. See also Change Cross Section.

THICKNESS (2D COMPONENTS)

For 2D components, enter a default value for the **Out-of-plane thickness** d (SI unit: m). The default value of 1 is typically not representative for a thin dielectric medium, for example. Instead it describes a unit thickness that makes the 2D equation identical to the equation used for 3D components. See also Change Thickness (Out-of-Plane).

MANUAL TERMINAL SWEEP SETTINGS

Enter a **Reference impedance** Z_{ref} (SI unit: Ω). The default is 50 Ω.

Select the **Use manual terminal sweep** check box to switch on the sweep and invoke a parametric sweep over the terminals. Enter a **Sweep parameter name** to assign a specific name to the variable that controls the terminal number solved for during the sweep. The **Sweep parameter name** must also be declared as a model parameter. The default is `PortName`.

The generated lumped parameters are in the form of capacitance matrix elements. The terminal settings must consistently be of either fixed voltage or fixed charge type.

The lumped parameters are subject to **Touchstone file export**. Enter a file path or **Browse** for a file. Select a **Parameter format (value pairs)** for the Touchstone export — **Magnitude and angle (MA)** (the default), **Magnitude in dB and angle (DB)**, or **Real and imaginary parts (RI)**. Select an option from the **If file exists** list — **Overwrite** or **Create new**. Create new is useful when the model is solved multiple times with different settings. Select a **Parameter to export**: **Z** (the default), **Y**, or **S**.

From the **Home** toolbar add a **Parameters** node and enter the chosen name and assign to it a temporary **Expression** of unity into the **Parameters** table. You can also right-click **Global Definitions** to add the node.

Only a temporary expression needs to be entered at this stage. During the analysis process, the solver assigns a proper value to this parameter. The generated lumped parameters are in the form of capacitance matrix elements. The terminal settings must consistently be of either fixed voltage or fixed charge type.

For most applications it is simpler to sweep over terminals using the Stationary Source Sweep study step. See Stationary Source Sweep in the *COMSOL Multiphysics Reference Manual* and (for more details) Stationary Source Sweep in the *AC/DC Module User's Guide*.

DEPENDENT VARIABLES

The dependent variable is the **Electric potential** V. You can change its name, which changes both the field name and the variable name. If the new name coincides with the name of another electric potential field in the model, the physics interfaces shares degrees of freedom. The new name must not coincide with the name of a field of another type or with a component name belonging to some other field.

DISCRETIZATION

Select the shape order for the **Electric potential** dependent variable — **Linear**, **Quadratic** (the default), **Cubic**, **Quartic**, or **Quintic**. For more information about the **Discretization** section, see Settings for the Discretization Sections in the *COMSOL Multiphysics Reference Manual*.

In the *COMSOL Multiphysics Reference Manual*, see Table 2-4 for links to common sections and Table 2-5 to common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

- Domain, Boundary, Edge, Point, and Pair Nodes for the Electrostatics Interface
- Theory for the Electrostatics Interface

Electric Sensor: Application Library path **COMSOL_Multiphysics/ Electromagnetics/electric_sensor**

Domain, Boundary, Edge, Point, and Pair Nodes for the Electrostatics Interface

The Electrostatics interface has these domain, boundary, edge, point, and pair nodes available.

ABOUT THE BOUNDARY CONDITIONS

The relevant physics interface condition at interfaces between different media is

$$\mathbf{n}_2 \cdot (\mathbf{D}_1 - \mathbf{D}_2) = \rho_\mathrm{s}$$

In the absence of surface charges, this condition is fulfilled by the natural boundary condition

$$\mathbf{n} \cdot [(\varepsilon_0 \nabla V - \mathbf{P})_1 - (\varepsilon_0 \nabla V - \mathbf{P})_2] = -\mathbf{n} \cdot (\mathbf{D}_1 - \mathbf{D}_2) = 0$$

AVAILABLE NODES

These nodes, listed in alphabetical order, are available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or right-click to access the context menu (all users). Also see Table 11-1 for a list of interior and exterior boundary conditions, including edge, point, and pair availability.

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

- Change Cross Section
- Change Thickness (Out-of-Plane)
- Charge Conservation
- Charge Conservation, Piezoelectric[1,2]
- Conduction Loss (Time-Harmonic)
- Dielectric Contact
- Dielectric Shielding
- Distributed Capacitance
- Electric Displacement Field
- Electric Potential
- Electrostatic Point Dipole
- External Surface Charge Accumulation
- Floating Potential
- Ground
- Initial Values
- Line Charge
- Line Charge (on Axis)
- Line Charge (Out-of-Plane)
- Metal Contact
- Periodic Condition
- Point Charge
- Point Charge (on Axis)
- Space Charge Density
- Surface Charge Accumulation
- Surface Charge Density
- Symmetry Plane (for Electric Field)
- Terminal
- Thin Low Permittivity Gap
- Zero Charge (the default boundary condition)

[1] This feature is available with the Piezoelectricity multiphysics interface.

[2] Requires either the Acoustics Module, MEMS Module, or Structural Mechanics Module.

Table 11-1 lists the interior and exterior boundary conditions available with this physics interface. It also includes edge, point, and pair availability.

TABLE 11-1: INTERIOR AND EXTERIOR BOUNDARY CONDITIONS (INCLUDING EDGE, POINT, AND PAIR AVAILABILITY) FOR THE ELECTROSTATICS INTERFACE.

NODE	INTERIOR	EXTERIOR	ALSO AVAILABLE FOR
Change Cross Section	x	x	pairs
Change Thickness (Out-of-Plane)	x	x	pairs
Dielectric Shielding	x	x	pairs
Distributed Capacitance		x	pairs
Electric Displacement Field	x	x	pairs
Electric Potential	x	x	edges, points, and pairs
External Surface Charge Accumulation		x	pairs
Floating Potential	x	x	pairs

TABLE 11-1: INTERIOR AND EXTERIOR BOUNDARY CONDITIONS (INCLUDING EDGE, POINT, AND PAIR AVAILABILITY) FOR THE ELECTROSTATICS INTERFACE.

NODE	INTERIOR	EXTERIOR	ALSO AVAILABLE FOR
Ground	x	x	edges, points, and pairs
Periodic Condition		x	not applicable
Surface Charge Accumulation	x		not applicable
Surface Charge Density	x	x	pairs
Symmetry Plane		x	not applicable
Terminal	x	x	domains
Thin Low Permittivity Gap	x		not applicable
Zero Charge (the default)	x	x	pairs

For axisymmetric components, COMSOL Multiphysics takes the axial symmetry boundaries (at $r = 0$) into account and automatically adds an **Axial Symmetry** node to the model that is valid on the axial symmetry boundaries only. There are also **Line Charge (on Axis)** and **Point Charge (on Axis)** available.

In the *COMSOL Multiphysics Reference Manual*, see Table 2-4 for links to common sections and Table 2-5 to common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

Charge Conservation

The **Charge Conservation** node adds the equations for charge conservation according to Gauss' law for the electric displacement field. It provides an interface for defining the constitutive relation and its associated properties such as the relative permittivity.

MATERIAL TYPE

The **Material type** setting decides how materials behave and how material properties are interpreted when the mesh is deformed. Select **Solid** for materials whose properties change as functions of material strain, material orientation, and other variables evaluated in a material reference configuration (material frame). Select **Nonsolid** for materials whose properties are defined only as functions of the current local state at each point in the spatial frame, and for which no unique material reference configuration can be defined. Select **From material** to pick up the corresponding setting from the domain material on each domain.

CONSTITUTIVE RELATION D-E

Select a **Dielectric model** to describe the macroscopic properties of the medium (relating the electric displacement $\mathbf{D}$ with the electric field $\mathbf{E}$) and the applicable material properties, such as the relative permittivity. Select:

- **Relative permittivity** (the default) to use the constitutive relation $\mathbf{D} = \varepsilon_0\varepsilon_r\mathbf{E}$. Then the default is to take the **Relative permittivity** ε_r (dimensionless) values **From material**. For **User defined**, select **Isotropic**, **Diagonal**, **Symmetric**, or **Full** and enter values or expressions in the field or matrix. The default is 1.
- **Polarization** to use the constitutive relation $\mathbf{D} = \varepsilon_0\mathbf{E} + \mathbf{P}$. Then enter the components based on space dimension for the **Polarization** vector $\mathbf{P}$ (SI unit: $\mathrm{C/m^2}$). The defaults are 0 $\mathrm{C/m^2}$.
- **Remanent electric displacement** to use constitutive relation $\mathbf{D} = \varepsilon_0\varepsilon_r\mathbf{E} + \mathbf{D}_r$, where $\mathbf{D}_r$ is the remanent displacement (the displacement when no electric field is present). Then the default is to take the **Relative permittivity** ε_r (dimensionless) values **From material**. For **User defined**, select **Isotropic**, **Diagonal**, **Symmetric**, or **Full** and enter values or expressions in the field or matrix. Then enter the components based on space dimension for the **Remanent electric displacement** $\mathbf{D}_r$ (SI unit: $\mathrm{C/m^2}$). The defaults are 0 $\mathrm{C/m^2}$.

Conduction Loss (Time-Harmonic)

This feature requires either the Acoustics Module, or MEMS Module, or Structural Mechanics Module. See the individual documentation for information.

The **Conduction Loss (Time-Harmonic)** subnode to **Charge Conservation** allows you to model possible conductive losses in a dielectric material. The effect is only active in an Eigenfrequency or Frequency Domain study.

CONDUCTION CURRENT

By default, the **Electrical conductivity** $\sigma$for the media is defined **From material**. You can also select **User defined** or **Linearized resistivity**.

- For **User defined** select **Isotropic**, **Diagonal**, **Symmetric**, or **Full** depending on the characteristics of the electrical conductivity, and then enter values or expressions for the **Electrical conductivity** σin the field or matrix.
- For **Linearized resistivity** the default **Reference temperature** T_{ref}, and **Resistivity temperature coefficient** α, and **Reference resistivity** ρ_0 are taken **From material**, which means that the values are taken from the domain (or boundary) material. T is the current temperature, which can be a value that is specified as a model input or the temperature from a heat transfer interface. The definition of the temperature field appears in the **Model Inputs** section.

LOCATION IN USER INTERFACE

Context Menus

Electrostatics>Charge Conservation>Conduction Loss (Time-Harmonic)

Ribbon

Physics tab with **Charge Conservation** node selected in the model tree:

Attributes>Conduction Loss (Time-Harmonic)

Initial Values

The **Initial Values** node adds an initial value for the electric potential V that can serve as an initial condition for a transient simulation or as an initial guess for a nonlinear solver.

INITIAL VALUES

Enter a value or expression for the initial value of the **Electric potential** V (SI unit: V). The default value is 0 V.

Space Charge Density

The **Space Charge Density** node adds a space charge density ρ, which appears on the right-hand side of the equation that the physics interface defines.

SPACE CHARGE DENSITY

Enter a value or expression for the **Space charge density** ρ_v (SI unit: C/m^3). The default is 0 C/m^3.

Zero Charge

The **Zero Charge** node adds the condition that there is zero charge on the boundary so that $\mathbf{n} \cdot \mathbf{D} = 0$. This boundary condition is also applicable at symmetry boundaries where the potential is known to be symmetric with respect to the boundary. This is the default boundary condition at exterior boundaries. At interior boundaries, it means that no displacement field can penetrate the boundary and that the electric potential is discontinuous across the boundary.

Ground

The **Ground** node implements ground (zero potential) as the boundary condition $V = 0$.

Ground means that there is a zero potential on the boundary. This boundary condition is also applicable at symmetry boundaries where the potential is known to be antisymmetric with respect to the boundary.

For some physics interfaces, also select additional **Ground** nodes from the **Edges** (3D components) or **Points** (2D and 3D components) submenus. For 2D axisymmetric components, it can be applied on the Symmetry axis.

BOUNDARY, EDGE, OR POINT SELECTION

⚠ Beware that constraining the potential on edges or points in 3D or on points in 2D usually yields a current outflow that is mesh dependent.

CONSTRAINT SETTINGS

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

Electric Potential

The **Electric Potential** node provides an electric potential V_0 as the boundary condition $V = V_0$.

Because the electric potential is being solved for in the physics interface, the value of the potential is typically defined at some part of the geometry. For some physics interfaces, also select additional **Electric Potential** nodes from the **Edges** (3D components) or **Points** (2D and 3D components) submenus. For 2D axisymmetric components, it can be applied on the symmetry axis.

BOUNDARY, EDGE, OR POINT SELECTION

⚠ Beware that constraining the potential on edges or points in 3D or on points in 2D usually yields a current outflow that is mesh dependent.

ELECTRIC POTENTIAL

Enter the value or expression for the **Electric potential** V_0 (SI unit: V). The default is 0 V.

CIRCUIT SETTINGS

🔍 Capacitively Coupled Plasmas

Select the **Circuit type** — **None** (the default), **Series RC circuit**, **Ballast resistor**, or **Blocking capacitor**. If **None** is selected, no external circuit is added.

Series RC Circuit

If **Series RC circuit** is selected, enter a value or expression for the **Ballast resistor** R_b (SI unit: Ω) and **Blocking capacitance** C_b (SI unit: F). The defaults are 1000 Ω and 1 pF, respectively. This option is typically used for metallic electrodes in DC discharges. The presence of the external circuit prevents the discharge from arcing. If arcing begins to occur, the solver tends to fail and the ballast resistor value should be increased. The presence of the blocking capacitor also helps when solving by ramping up the applied potential from zero over a finite period of time.

Ballast Resistor

If **Ballast resistor** is selected, enter a value or expression for the **Ballast resistor** R_b (SI unit: Ω). The default is 1000 Ω. The presence of the ballast resistor reduces the likelihood that the discharge arcs. However, it is recommended that the **Series RC circuit** be chosen in preference to this one.

Blocking Capacitor

If **Blocking capacitor** is selected, enter a value or expression for the **Blocking capacitance** C_b (SI unit: F). The default is 1 pF. For 1D components, enter a value for the **Electrode surface area** A_s (SI unit: m^2). The default is 0.01 m^2. This value is used to compute the total amount of electrical current flowing into the electrode due to the time-varying displacement field and flow of charged particles. This option is typically used when modeling capacitively coupled plasmas, since these typically operate with a blocking capacitor between the power supply and the electrodes.

CONSTRAINT SETTINGS

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

Surface Charge Density

The **Surface Charge Density** node provides the following surface-charge boundary condition for exterior boundaries (left) and interior boundaries (right):

$$-\mathbf{n}\cdot\mathbf{D} = \rho_s, \qquad \mathbf{n}\cdot(\mathbf{D}_1-\mathbf{D}_2) = \rho_s$$

Specify the surface charge density ρ_s at an outer boundary or at an interior boundary between two nonconducting media.

Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or click **Harmonic Perturbation** in the **Physics** toolbar. For more information, see Harmonic Perturbation — Exclusive and Contributing Nodes in the *COMSOL Multiphysics Reference Manual.*

SURFACE CHARGE DENSITY

Enter the value or expression for the **Surface charge density** ρ_s (SI unit: C/m^2).

External Surface Charge Accumulation

The **External Surface Charge Accumulation** node implements the boundary condition

$$-\mathbf{n}\cdot\mathbf{D} = \rho_s$$

where ρ_s is the solution of the following distributed ODE on the boundary:

$$\frac{d\rho_s}{dt} = \mathbf{n}\cdot\mathbf{J}_i+\mathbf{n}\cdot\mathbf{J}_e$$

where $\mathbf{n}\cdot\mathbf{J}_i$ is the normal component of the total ion current density on the wall and $\mathbf{n}\cdot\mathbf{J}_e$ is the normal component of the total electron current density on the wall, which are feature inputs.

MATERIAL TYPE

The **Material type** setting decides how materials behave and how material properties are interpreted when the mesh is deformed. Select **Solid** for materials whose properties change as functions of material strain, material orientation, and other variables evaluated in a material reference configuration (material frame). Select **Nonsolid** for materials whose properties are defined only as functions of the current local state at each point in the spatial frame, and for which no unique material reference configuration can be defined. Select **From material** to pick up the corresponding setting from the domain material on each domain.

EXTERNAL SURFACE CHARGE ACCUMULATION

Enter values or expressions for the **Normal ion current density** $\mathbf{n}\cdot\mathbf{J}_i$ (SI unit: $\mathrm{A/m^2}$) sand the **Normal electron current density** $\mathbf{n}\cdot\mathbf{J_e}$ (SI unit: $\mathrm{A/m^2}$).

Symmetry Plane (for Electric Field)

The **Symmetry Plane** node adds a boundary condition that represents symmetry or antisymmetry in the electric field, depending on which option is chosen.

SYMMETRY TYPE

Choose between **Symmetry** and **Antisymmetry** for the electric field. If **Symmetry** is chosen, the boundary condition is

$$\mathbf{n}\cdot\mathbf{E} = 0$$

which states that the normal component of the electric field is zero. If **Antisymmetry** is chosen, the boundary condition becomes

$$\mathbf{n}\times\mathbf{E} = 0$$

which states that the tangential component of the electric field is zero.

Electric Displacement Field

The **Electric Displacement Field** node adds the following electric-displacement boundary condition:

$$\mathbf{n} \cdot \mathbf{D} = \mathbf{n} \cdot \mathbf{D}_0$$

It specifies the normal component of the electric displacement field at a boundary.

ELECTRIC DISPLACEMENT FIELD

Enter the coordinates of the **Boundary electric displacement field** $\mathbf{D}_0$ (SI unit: $\mathrm{C/m^2}$).

Periodic Condition

The **Periodic Condition** node defines periodicity or antiperiodicity between two boundaries. If required, activate periodic conditions on more than two boundaries, in which case the Periodic Condition tries to identify two separate surfaces that can each consist of several connected boundaries. For more complex geometries, it might be necessary to use the **Destination Selection** section, which you activate by right-clicking the **Periodic Condition** node and select **Manual Destination Selection**. With that section, the boundaries which constitute the source and destination surfaces can be manually specified.

When this feature is used in conjunction with a Sector Symmetry feature on connected boundaries, wherever the sector symmetry boundaries connect with the periodic boundaries, the same periodic condition feature cannot be used on both sides. At least two periodic condition features are required for the model to compute correctly.

BOUNDARY SELECTION

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

When using nonconforming meshes on the source and destination of a periodic boundary pair, for numerical stability, a finer mesh should be applied on the destination side. Use conforming meshes if possible.

DESTINATION SELECTION

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

PERIODIC CONDITION

Select a **Type of periodicity** — **Continuity** (the default), **Antiperiodicity**, or **Floquet periodicity**. Select:

- **Continuity** to make the potential periodic (equal on the source and destination).
- **Antiperiodicity** to make it antiperiodic.
- **Floquet periodicity** (only available with products supporting piezoelectric modeling). Specify the components of the **k-vector for Floquet periodicity** $\mathbf{k}_\mathrm{F}$ (SI unit: rad/m).

CONSTRAINT SETTINGS

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

ORIENTATION OF SOURCE

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

ORIENTATION OF DESTINATION

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

Thin Low Permittivity Gap

Use the **Thin Low Permittivity Gap** node

$$\mathbf{n}\cdot\mathbf{D}_1 = \frac{\varepsilon_0\varepsilon_\mathrm{r}}{d}(V_1 - V_2)$$

$$\mathbf{n}\cdot\mathbf{D}_2 = \frac{\varepsilon_0\varepsilon_\mathrm{r}}{d}(V_2 - V_1)$$

to model a thin gap of a material with a small permittivity compared to the adjacent domains. The layer has the thickness d and the relative permittivity ε_r. The indices 1 and 2 refer to the two sides of the boundary.

THIN LOW PERMITTIVITY GAP

The default is to take the **Relative permittivity** ε_r (dimensionless) values **From material**. For **User defined**, enter a different value or expression. Enter a **Thickness** d (SI unit: m). The default is 5 mm.

Surface Charge Accumulation

On interior boundaries, the **Surface Charge Accumulation** node implements the boundary condition

$$\mathbf{n}\cdot(\mathbf{D}_1 - \mathbf{D}_2) = \rho_\mathrm{s}$$

where ρ_s is the solution of the following distributed ODE on the boundary:

$$\frac{d\rho_\mathrm{s}}{dt} = \mathbf{n}\cdot\mathbf{J}_i + \mathbf{n}\cdot\mathbf{J}_e$$

where $\mathbf{n}\cdot\mathbf{J}_i$ is the normal component of the total ion current density on the wall and $\mathbf{n}\cdot\mathbf{J}_e$ is the normal component of the total electron current density on the wall.

SURFACE CHARGE ACCUMULATION

Enter the values or expressions for the **Normal ion current density** $\mathbf{n}\cdot J_i$ (SI unit: A/m^2) and the **Normal electron current density** $\mathbf{n}\cdot J_e$ (SI unit: A/m^2).

Dielectric Barrier Discharge: Application Library path **Plasma_Module/ Direct_Current_Discharges/argon_dbd_1d**

Dielectric Shielding

The **Dielectric Shielding** node adds dielectric shielding as a boundary condition. It describes a thin layer with thickness d_s and a bulk relative permittivity ε_r that shields the electric field:

$$\mathbf{n}\cdot\mathbf{D} = -\nabla_\mathrm{t}\cdot\varepsilon_0\varepsilon_{\mathrm{rs}}d_\mathrm{s}\nabla_\mathrm{t}V$$

Use this boundary condition when approximating a thin domain with a boundary to reduce the number of mesh elements.

MATERIAL TYPE

The **Material type** setting decides how materials behave and how material properties are interpreted when the mesh is deformed. Select **Solid** for materials whose properties change as functions of material strain, material orientation and other variables evaluated in a material reference configuration (material frame). Select **Nonsolid** for materials whose properties are defined only as functions of the current local state at each point in the spatial frame, and for which no unique material reference configuration can be defined. Select **From material** to pick up the corresponding setting from the domain material on each domain.

ELECTRIC SHIELDING

The default is to take the **Relative permittivity** ε_r (dimensionless) values **From material**. It takes it from the adjacent domains if not explicitly defined. For **User defined** select **Isotropic**, **Diagonal**, **Symmetric**, or **Full** and enter values or expressions in the field or matrix. Enter a **Surface thickness** d_s (SI unit: m) of the shielding. The default is 1 m.

Terminal

The **Terminal** node provides a boundary or domain condition for connection to external circuits, to transmission lines, or with a specified voltage or charge. By specifying zero charge, a floating potential condition is obtained.

An operation called **Split by Connectivity** (✎) is available by right-clicking the parent node. The operation is useful when there are several terminal domains or boundaries that are not geometrically connected. Clicking the operation node will automatically generate several **Terminal** nodes where each selection is geometrically connected.

The **Terminal** node also enables the computation of the lumped parameters of the system, such as capacitance.

TERMINAL

Specify the terminal properties. Enter a unique **Terminal name** that will be used to identify the global variables (such as the terminal voltage or current) that are defined by the terminal. The **Terminal name** can contain alphanumeric characters, but it should be numeric when performing terminal sweeps.

Select a **Terminal type** — **Charge** (the default), **Voltage**, **Circuit**, **Terminated (power)**, or **Terminated (voltage)**. Select:

- **Charge** to prescribe the total charge deposited on the terminal boundaries. Enter the total charge Q_0. The default is 0 C, which corresponds to a uncharged floating electrode.
- **Voltage** to specify a fixed electric potential at the terminal. Enter an electric potential V_0. The default is 1 V.
- **Circuit** to specify a terminal connected to an Electrical Circuit interface that models an external circuit. The Terminal node provides a current-voltage characteristic to the circuit.

The Electrical Circuit Interface requires a current from the Terminal, so the **Circuit** terminal type can only be used in Time Dependent or Frequency Domain studies, in which the current can be computed as the time derivative of the charge.

The **Circuit** type cannot be used together with a terminal sweep.

- **Terminated (power)** to connect the terminal to an impedance that might represent a load or a transmission line. When **Terminated (power)** is selected, the scattering parameters (S-parameters) are computed. The **Terminated (power)** setting can only be used in frequency-domain studies.
  - Enter a **Terminal power** P_0 to specify the input power at this terminal from the transmission line. This excitation can be toggled using a port sweep.
  - Select an option from the **Characteristic impedance from** list to define the value of the impedance — **Physics interface** or **User defined**. For **Physics interface**, the **Reference impedance** Z_{ref} defined in the **Settings** window for the physics under

Sweep Settings is used. For **User defined**, enter a specific impedance Z_{ref} for this terminal. The default is 50 Ω.

- **Terminated (voltage)** to connect the terminal to an impedance that might represent a load or a transmission line. The **Terminated (voltage)** option is essentially the same as **Terminated (power)**, with the power defined as: $P_0 = 0.5 \cdot |V_0|^2 / Z_{\mathrm{ref}}$.

When **Terminal type** is selected as **Charge** or **Voltage**, a contribution can be added as **Harmonic Perturbation** by right-clicking the parent node or clicking **Harmonic Perturbation** in the **Physics** toolbar. For more information see Harmonic Perturbation — Exclusive and Contributing Nodes in the *COMSOL Multiphysics Reference Manual*.

Initial values

The initial values subsection is used to provide initial values for the internal state variables in the Terminal. These settings can be useful to provide an initial guess to the nonlinear solver or the initial value in a time-dependent study.

The **Initial value for voltage** is available for all Terminal types except Voltage. Enter an **Initial value for voltage** V_{init}.

The **Initial value for charge** is available for the Circuit excitation. It specifies the initial charge on the terminal when the circuit is connected. Enter an **Initial value for charge** Q_{init}.

CONSTRAINT SETTINGS

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

Studies and Solvers in the *COMSOL Multiphysics Reference Manual*

Floating Potential

The **Floating Potential** node is used when modeling a metallic electrode at floating potential. It applies a constant voltage V_0 on the boundary (for domain features, this is the boundary enclosing the selected domain), such that the total normal electric displacement field $\mathbf{D}$ equals a specific charge Q_0:

$$V = V_0$$

$$\int_{\partial\Omega} (\mathbf{D} \cdot \mathbf{n}) dS = Q_0,$$

where $\partial\Omega$ represents the boundary and $\mathbf{n}$ refers to the surface normal. The constant boundary voltage implies the tangential electric field equals zero; the electric field will be perpendicular to the boundary:

$$-\nabla_{\mathrm{t}} V = \mathbf{E}_{\mathrm{t}} = 0$$

$$\mathbf{n} \times \mathbf{E} = 0.$$

In case of $Q_0 = 0$ (the default case), the boundary will behave as an unconnected, neutrally charged, good conductor under electrostatic conditions (a floating equipotential). Although locally the displacement field may vary, the total electric displacement field entering or leaving the boundary equals zero.

If the floating potential touches a point, boundary, or domain feature that is not floating (a **Terminal** or **Ground** feature), the floating potential will acquire that feature's potential. If the floating potential is set to a certain charge, or connected to a circuit, it behaves like a terminal.

FLOATING POTENTIAL

The **Floating potential identifier** text area shows the unique identifier for the floating potential feature. It is used to identify the global variables created by the node, such as the voltage. The **Floating potential group** check box in the **Settings** window for **Floating Potential** controls how potentials are assigned to boundary segments in the feature's boundary selection. If this check box is not selected, a single potential is applied to all boundaries in the selection. If the check box is selected, each group of contiguous boundaries in the selection is given a unique potential. This simplifies the setup of models with many floating electrodes. The values of the potential at each group of boundaries are then made available in postprocessing, collected in a vector variable.

The following options are not available if the **Floating potential group** check box is selected. Select a specification for the **Electric charge** — choose **User defined** to specify a total **Charge** Q_0 deposited on the surface. Select **Circuit** to connect the floating potential to an Electrical Circuit.

> ! The Electrical Circuit Interface requires a current from the Floating Potential, so the **Circuit** terminal type can only be used in Time Dependent or Frequency Domain studies, in which the current can be computed as the time derivative of the charge.

Initial Values

The initial values subsection is used to provide initial values for the internal state variables in the Floating potential. These settings can be useful to provide an initial guess to the nonlinear solver or the initial value in a time-dependent study.

Enter an **Initial value for voltage** V_{init}.

The **Initial value for charge** is available for the Circuit excitation. It specifies the initial charge on the floating potential when the circuit is connected. Enter an **Initial value for charge** Q_{init}.

CONSTRAINT SETTINGS

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

> Settings and Properties Windows for Feature Nodes in the *COMSOL Multiphysics Reference Manual*

Distributed Capacitance

The **Distributed Capacitance** node adds a distributed capacitance boundary condition according to the following equations for exterior boundaries:

$$-\mathbf{n}\cdot\mathbf{D} = \varepsilon_0\varepsilon_r\frac{V_{\mathrm{ref}}-V}{d_s}$$

Use this boundary condition to model a thin sheet or film of a dielectric material. The sheet has the relative permittivity ε_{r} and the surface thickness d_{s}, and it is connected to the reference potential V_{ref}.

DISTRIBUTED CAPACITANCE

Enter the values or expressions for **Relative permittivity** ε_r (dimensionless), **Surface thickness** d_s (SI unit: m), and **Reference potential** V_{ref}. The default value for the surface thickness is 10^{-3} m (1 mm) and 0 V for the reference potential.

Line Charge

For 3D components, use the **Line Charge** node to specify line charges along the edges of a geometry. Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or clicking **Harmonic Perturbation** in the **Physics** toolbar. For more information see Harmonic Perturbation — Exclusive and Contributing Nodes in the *COMSOL Multiphysics Reference Manual*.

EDGE SELECTION

Beware that constraining the potential on edges usually yields a current outflow that is mesh dependent.

LINE CHARGE

Enter a value or expression to apply a **Line charge** Q_L (SI unit: C/m). This source represents electric charge per unit length and the default is 0 C/m.

Line Charge (on Axis) and Line Charge (Out-of-Plane)

Line Charge (on Axis)

For 2D axisymmetric components, use the **Line Charge (on Axis)** node to specify line charges along the symmetry axis.

Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or click **Harmonic Perturbation** in the **Physics** toolbar. For more information see Harmonic Perturbation — Exclusive and Contributing Nodes in the *COMSOL Multiphysics Reference Manual*.

LINE CHARGE (ON AXIS)

Enter a value or expression to apply a **Line charge** Q_L (SI unit: C/m). This source represents electric charge per unit length and the default is 0 C/m.

Line Charge and Line Charge (Out-of-Plane)

Line Charge (Out-of-Plane)

For 2D and 2D axisymmetric components, points are selected and this is the same as a line out-of-plane.

Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or clicking **Harmonic Perturbation** in the **Physics** toolbar. For more information see Harmonic Perturbation — Exclusive and Contributing Nodes in the *COMSOL Multiphysics Reference Manual*.

Use the **Line Charge (Out-of-Plane)** node to specify line charges along the points of a geometry for 2D and 2D axisymmetric components.

POINT SELECTION

Beware that constraining the potential on points usually yields a current outflow that is mesh dependent.

LINE CHARGE (OUT-OF-PLANE)

Enter a value or expression to apply a **Line charge** Q_L (SI unit: C/m). This source represents electric charge per unit length and the default is 0 C/m.

Line Charge and Line Charge (on Axis)

Point Charge

The **Point Charge** node adds a point source to 3D components. The point charge represents an electric displacement field flowing out of the point.

Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or clicking **Harmonic Perturbation** in the **Physics** toolbar. For more information see Harmonic Perturbation — Exclusive and Contributing Nodes in the *COMSOL Multiphysics Reference Manual*.

POINT SELECTION

⚠ Beware that constraining the potential on points usually yields a current outflow that is mesh dependent.

POINT CHARGE

Enter a value or expression to apply a **Point charge** Q_P (SI unit: C) to points. This source represents an electric displacement field flowing out of the point. The default is 0 C.

Point Charge (on Axis) and Line Charge (Out-of-Plane)

Point Charge (on Axis)

The **Point Charge (on Axis)** node adds a point source to 2D axisymmetric components. The point charge represents an electric displacement field flowing out of the point.

Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or clicking **Harmonic Perturbation** in the **Physics** toolbar. For more information see Harmonic Perturbation — Exclusive and Contributing Nodes in the *COMSOL Multiphysics Reference Manual*.

POINT SELECTION

⚠ Beware that constraining the potential on points usually yields a current outflow that is mesh dependent.

POINT CHARGE (ON AXIS)

Enter a value or expression to apply a **Point charge** Q_P (SI unit: C) to points on an axis. This source represents an electric displacement field flowing out of the point. The default is 0 C.

Point Charge and Line Charge (Out-of-Plane)

Change Cross Section

This node is available with 1D components. This setting overrides the global **Cross-Section Area** setting made in any physics interface that uses this feature. For 2D components, see Change Thickness (Out-of-Plane).

Use the **Change Cross Section** node to set the cross-section area for specific geometric entities.

CHANGE CROSS SECTION

Enter a value or expression for the **Cross-section area** A. The default value of one unit area is typically not representative for the actual domain. Instead, it describes a unit area that makes the 1D equation identical to the equation used for 3D components.

Change Thickness (Out-of-Plane)

This node is available for 2D components. This setting overrides the global **Thickness** setting made in any physics interface that uses this node. For 1D components, see Change Cross Section.

Use the **Change Thickness (Out-of-Plane)** node to set the out-of-plane thickness for specific geometric entities.

CHANGE THICKNESS (OUT-OF-PLANE)

Enter a value or expression for the **Out-of-plane thickness** d (SI unit: m). The default value is, in most cases, 1 unit length, which is typically not representative for a thin domain. Instead it describes a unit thickness that makes the 2D equation identical to the equation used for 3D components.

Charge Conservation, Piezoelectric

This feature is available with the Piezoelectricity interface, which requires either the Acoustics Module, or MEMS Module, or Structural Mechanics Module. See the individual documentation for information.

The **Charge Conservation, Piezoelectric** node is normally used together with a **Piezoelectric Effect** multiphysics coupling node and a corresponding **Piezoelectric Material** node in the **Solid Mechanics** interface. The node is added by default to the **Electrostatics** interface when adding a Piezoelectricity interface. It is also available from the context menu (right-click the **Electrostatics** interface parent node) or from the **Physics** toolbar.

When the **Charge Conservation, Piezoelectric** node is added to the electrostatics interface in the absence of an active **Piezoelectric Effect** multiphysics coupling node, the material behaves similarly to a Charge Conservation node, with electric properties corresponding to the relative permittivity entered (see below). The piezoelectric effect is *not* included in the corresponding equation system.

ELECTRIC DISPLACEMENT

If the node is used together with an active **Piezoelectric Effect** multiphysics coupling node, then these settings are locked. Note that if they are unlocked, then the material behaves like a dielectric and *not* a piezoelectric. In this case, the default is to take the **Relative permittivity** $\varepsilon_{\mathrm{rS}}$ (dimensionless) values **From material**. For **User defined**, select **Isotropic**, **Diagonal**, **Symmetric**, or **Full** and enter values or expressions in the field or matrix.

Metal Contact

The **Metal Contact** node is used for modeling what happens when a plasma is in contact with a metal surface. The **Metal Contact** can only be applied on boundaries adjacent to the **Plasma Model** feature. This feature is available when using The Plasma Interface, The Inductively Coupled Plasma Interface, and The Microwave Plasma Interface.

TERMINAL

Specify the terminal properties. To indicate which boundaries belong to the same terminal, enter the same name in the **Terminal name** field.

Select a **Terminal type** — **Voltage** (the default), **Current** or **Circuit**. Select:

- **Voltage** to enter an electric potential V_0.
- **Current** to enter the Terminal current I_0. This ensures that the total current arriving on the boundary is equal to the specified value. The current arriving on the boundary is the sum of the displacement current due to the time varying electric potential and conduction current due to the fact that ions an electrons may be incident on the surface.
- **Circuit** to specify a terminal connected to an external circuit. The Terminal node provides a current-voltage characteristic to the circuit element.

If the finite element discretization is employed and when using the **Current** or **Circuit** options (see below), then the **Weak constraints** option in **Constraint** could be activated in the **Constraint Settings** section to make the problem easier to solve.

QUICK CIRCUIT SETTINGS

This section is available when **Voltage** is selected as the **Terminal type**. Select a **Quick circuit type** — **None** (the default), **Series RC circuit**, **Ballast resistor**, or **Blocking capacitor**. These settings are the same as described under Circuit Settings for Electric Potential.

CONSTRAINT SETTINGS

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

Dielectric Contact

Use the **Dielectric Contact** node is used for modeling what happens when a plasma is in contact with a dielectric (insulating) surface. The **Dielectric Contact** can only be applied on boundaries adjacent to the **Plasma Model** feature.

SURFACE CHARGE TYPE

Select a Surface charge type — **Surface charge accumulation** (the default) or **Surface charge density**. For **Surface charge density** enter the value or expression for σ_s.

Distributed Capacitance

Click to select the **Enable distributed capacitance** check box. Then enter the values or expressions for **Relative permittivity** ε_r (dimensionless), **Surface thickness** d_s (SI unit: m), and **Reference potential** V_ref. The default value for the surface thickness is 10^{-3} m (1 mm) and 0 V for the reference potential.

Electrostatic Point Dipole

Add an **Electrostatic Point Dipole** node to 3D and 2D components. Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or click **Harmonic Perturbation** in the **Physics** toolbar. For more information see Harmonic Perturbation — Exclusive and Contributing Nodes in the *COMSOL Multiphysics Reference Manual*.

The **Electrostatic Point Dipole** represents the limiting case of zero separation distance between two equally strong point sources of opposing signs while maintaining the product between separation distance and source strength at a fixed value (p). The dipole moment is a vector entity with positive direction from the negative charge to the positive one.

Electrostatic Point Dipole

Select a **Dipole specification** — **Magnitude and direction** (the default) or **Electric dipole moment**.

- For **Magnitude and direction** enter coordinates for the **Electric dipole moment direction** $\mathbf{n}_p$ (dimensionless) and the **Electric dipole moment, magnitude** p.
- For **Electric dipole moment** enter coordinates for the **Electric dipole moment** $\mathbf{p}$.

The Electric Currents Interface

The **Electric Currents (ec)** interface ( ), under the **AC/DC>Electric Fields and Currents** branch when adding a physics interface, is used to compute electric field, current, and potential distributions in conducting media under conditions where inductive effects are negligible; that is, when the skin depth is much larger than the studied device.

Depending on the licensed products, stationary, frequency-domain, small-signal analysis, and time-domain modeling are supported in all space dimensions. In the time and frequency domains, capacitive effects are also accounted for.

The physics interface solves a current conservation equation based on Ohm's law using the scalar electric potential as the dependent variable. The interface is typically used to model good conductors for which Ohm's law applies. It is not suitable for modeling poor conductors like air and insulators.

Current Conservation is the main node, which adds the equation for the electric potential and provides a Settings window for defining the electrical conductivity as well as the constitutive relation for the electric displacement field and its associated material properties, such as the relative permittivity.

When this physics interface is added, these default nodes are also added to the **Model Builder** — **Current Conservation**, **Electric Insulation** (the default boundary condition), and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions and current sources. You can also right-click **Electric Currents** to select physics features from the context menu.

Physics-Controlled Mesh

The physics-controlled mesh is controlled from the **Mesh** node's **Settings** window (if the **Sequence type** is **Physics-controlled mesh**). There, in the table in the **Physics-Controlled Mesh** section, find the physics interface in the **Contributor** column and select or clear the check box in the **Use** column on the same table row for enabling (the default) or disabling contributions from the physics interface to the physics-controlled mesh.

Information from the physics, such as the presence of an infinite elements domain or periodic condition, will be used to automatically set up an appropriate meshing sequence.

! In the *COMSOL Multiphysics Reference Manual* see the Physics-Controlled Mesh section for more information about how to define the physics-controlled mesh.

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `ec`.

CROSS-SECTION AREA (1D)

Enter a default value for the **Cross-section area** A (SI unit: m^2). The default value of $1\ \mathrm{m}^2$ is typically not representative for a small domain. Instead it describes a unit area that makes the 1D equation identical to the equation used for 3D components. See also Change Cross Section (described for the Electrostatics interface).

THICKNESS (2D)

Enter a default value for the **Out-of-plane thickness** d (SI unit: m) (see Equation 11-1). The default value of 1 m is typically not representative for a thin dielectric medium, for example. Instead it describes a unit thickness that makes the 2D equation identical to the equation used for 3D components. See also Change Thickness (Out-of-Plane) (described for the Electrostatics interface).

MANUAL TERMINAL SWEEP SETTINGS

Enter a **Reference impedance** Z_{ref} (SI unit: Ω). The default is 50 Ω.

Select the **Use manual terminal sweep** check box to switch on the sweep and invoke a parametric sweep over the terminals. Enter a **Sweep parameter name** to assign a specific name to the variable that controls the terminal number solved for during the sweep. The default is `PortName`. The generated lumped parameters are in the form of capacitance matrix elements. The terminal settings must consistently be of either fixed voltage or fixed charge type.

The lumped parameters are subject to **Touchstone file export**. Enter a file path or **Browse** for a file. Select a **Parameter format (value pairs)** for the Touchstone export — **Magnitude and angle (MA)** (the default), **Magnitude in dB and angle (DB)**, or **Real and imaginary parts (RI)**. Select an option from the **If file exists** list — **Overwrite** or **Create new**. Create new is useful when the model is solved multiple times with different settings. Select a **Parameter to export** — **Z** (the default), **Y**, or **S**.

For most applications it is simpler to sweep over terminals using the Stationary Source Sweep study step. See Stationary Source Sweep in the *COMSOL Multiphysics Reference Manual* and (for more details) Stationary Source Sweep in the *AC/DC Module User's Guide*.

DEPENDENT VARIABLES

The dependent variable is the **Electric potential** V. You can change its name, which changes both the field name and the variable name. If the new name coincides with the name of another electric potential field in the model, the physics interfaces share degrees of freedom. The new name must not coincide with the name of a field of another type or with a component name belonging to some other field.

DISCRETIZATION

Select the shape order for the **Electric potential** dependent variable — **Linear**, **Quadratic** (the default), **Cubic**, **Quartic**, or **Quintic**. For more information about the **Discretization** section, see Settings for the Discretization Sections in the *COMSOL Multiphysics Reference Manual*.

- Domain, Boundary, Edge, Point, and Pair Nodes for the Electric Currents Interface
- Theory for the Electric Currents Interface

Pacemaker Electrode: Application Library path **COMSOL_Multiphysics/Electromagnetics/pacemaker_electrode**

Domain, Boundary, Edge, Point, and Pair Nodes for the Electric Currents Interface

The Electric Currents interface has these domain, boundary, edge, point, and pair nodes available from the **Physics** ribbon toolbar (Windows users) or **Physics** context menu (Mac or Linux users). You can also right-click to access the context menu (all users).

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

ABOUT THE BOUNDARY CONDITIONS

The exterior and interior boundary conditions listed in Table 11-1 are available. The relevant physics interface condition at interfaces between different media and interior boundaries is continuity; that is,

$$\mathbf{n}_2 \cdot (\mathbf{J}_1 - \mathbf{J}_2) = 0$$

which is the natural boundary condition.

AVAILABLE NODES

These nodes are available for this physics interface, listed in alphabetical order. Also see Table 11-1 for a list of interior and exterior boundary conditions, including edge, point, and pair availability.

- Boundary Current Source
- Contact Impedance
- Current Conservation
- Current Source
- Distributed Impedance
- Electric Insulation
- External Current Density
- Floating Potential
- Initial Values
- Line Current Source
- Line Current Source (on Axis)
- Normal Current DensityPoint Current Source
- Sector Symmetry
- Symmetry Plane (for Electric Field)
- Terminal

These nodes are described for the Electrostatics interface:

- Change Cross Section
- Change Thickness (Out-of-Plane)
- Electric Potential
- Ground
- Periodic Condition

In the *COMSOL Multiphysics Reference Manual*, see Table 2-4 for links to common sections and Table 2-5 for common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

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

Table 11-1 lists the interior and exterior boundary conditions available with this physics interface. It also includes edge, point, and pair availability.

TABLE 11-2: INTERIOR AND EXTERIOR BOUNDARY CONDITIONS (INCLUDING EDGE, POINT, AND PAIR AVAILABILITY) FOR THE ELECTRIC CURRENTS INTERFACE.

NODE	INTERIOR	EXTERIOR	ALSO AVAILABLE FOR
Boundary Current Source	x		pairs
Contact Impedance	x		pairs
Distributed Impedance	x	x	not applicable
Electric Insulation	x	x	pairs
Electric Potential	x	x	edges, points, and pairs
Floating Potential	x	x	pairs
Ground	x	x	edges, points, and pairs
Normal Current Density		x	not applicable
Periodic Condition		x	not applicable
Symmetry Plane		x	not applicable
Terminal	x	x	pairs, domains

Current Conservation

The **Current Conservation** node adds the continuity equation for the electrical potential and provides an interface for defining the electric conductivity as well as the constitutive relation and the relative permittivity for the displacement current.

MATERIAL TYPE

The **Material type** setting decides how materials behave and how material properties are interpreted when the mesh is deformed. Select **Solid** for materials whose properties change as functions of material strain, material orientation, and other variables evaluated in a material reference configuration (material frame). Select **Nonsolid** for materials whose properties are defined only as functions of the current local state at each point in the spatial frame, and for which no unique material reference configuration can be defined. Select **From material** to pick up the corresponding setting from the domain material on each domain.

CONSTITUTIVE RELATION JC-E

By default, the **Electrical conductivity** σ (SI unit: S/m) for the media is defined **From material**. Or select **User defined** or **Linearized resistivity**.

User Defined

For **User defined** select **Isotropic**, **Diagonal**, **Symmetric**, or **Full** depending on the characteristics of the electrical conductivity, and then enter values or expressions for the electrical conductivity σ in the field or matrix. The default is 0 S/m. If type of temperature dependence is used other than a linear temperature relation, enter any expression for the conductivity as a function of temperature.

Linearized Resistivity

Select **Linearized resistivity** for a temperature-dependent conductivity (this occurs in, for example, Joule heating, and is also called resistive heating). The equation describing the conductivity:

$$\sigma = \frac{1}{\rho_0(1+\alpha(T-T_{\mathrm{ref}}))}$$

where ρ_0 is the resistivity at the reference temperature T_{ref}, and α is the temperature coefficient of resistance, which describes how the resistivity varies with temperature.

The default **Reference resistivity** ρ_0 (SI unit: Ω·m), **Reference temperature** T_{ref} (SI unit: K), and **Resistivity temperature coefficient** α (SI unit: 1/K) are taken **From material**, which means that the values are taken from the domain (or boundary) material. T is the current temperature, which can be a value that is specified as a model input or the temperature from a heat transfer interface. The definition of the temperature field is in the **Model Inputs** section.

To specify other values for any of these properties, select **User defined** from the list and then enter a value or expression for each. The default values are:

- 1 Ω·m for the Reference resistivity
- 273.15 K for the Reference temperature, and
- 0 1/K for the Resistivity temperature coefficient

CONSTITUTIVE RELATION D-E

Select a **Dielectric model** to describe the macroscopic properties of the medium (relating the electric displacement $\mathbf{D}$ with the electric field $\mathbf{E}$) and the applicable material properties, such as the relative permittivity. For a description of the constitutive relations **Relative permittivity**, **Polarization**, and **Remanent electric displacement**, see **Constitutive Relation D-E** as described for the Charge Conservation node for the Electrostatics interface. The constitutive relations specific to Electric Currents are:

- **Dielectric losses**: uses the constitutive relation $\mathbf{D} = \varepsilon_0(\varepsilon' - j\varepsilon'')\mathbf{E}$. Specify that the **Relative permittivity (real part)** ε' (dimensionless) and the **Relative permittivity (imaginary part)** ε'' (dimensionless) must be taken **From material** or be **User defined**. For **User defined**, select **Isotropic**, **Diagonal**, **Symmetric**, or **Full** and enter values or expressions in the field or matrix. The default is 1. Note that the material parameters **Relative permittivity (real part)** ε' and the **Relative permittivity (imaginary part)** ε'' form the complex relative permittivity $\varepsilon_r = \varepsilon' - j\varepsilon''$. The time-harmonic Sign Convention requires a lossy material to have a positive material parameter ε''.
- **Loss tangent, loss angle**: uses the constitutive relation $\mathbf{D} = \varepsilon_0\varepsilon'(1 - j\tan\delta)\mathbf{E}$. Specify the **Relative permittivity (real part)** ε' (dimensionless) and **Loss angle** δ (SI unit: rad).
- **Loss tangent, dissipation factor**: uses the constitutive relation $\mathbf{D} = \varepsilon_0\varepsilon'(1 - j\tan\delta)\mathbf{E}$. Specify the **Relative permittivity (real part)** ε' (dimensionless) and the **Dissipation factor** $\tan\delta$ (dimensionless).

Initial Values

The **Initial Values** node adds an initial value for the electric potential that can serve as an initial condition for a transient simulation or as an initial guess for a nonlinear solver. If more than one set of initial values is required, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions and current sources. Add more **Initial Values** nodes from the **Physics** toolbar.

INITIAL VALUES

Enter a value or expression for the initial value of the **Electric potential** V (SI unit: V). The default value is 0 V.

External Current Density

The **External Current Density** node adds an externally generated current density $\mathbf{J}_\mathrm{e}$, which appears in Ohm's law

$$\mathbf{J} = \sigma\mathbf{E} + \mathbf{J}_\mathrm{e}$$

and in the equation that the physics interface defines.

The external current density does not contribute to the losses (due to Joule heating), since there is no electric field associated with it. To include the contribution to the losses from the external current density, select the **Add contribution of the external current density to the losses** check box. Then select an option from the **External losses** list — **From domain conductivity** (the default) or **User defined**. If **From domain conductivity** is selected, the heat source is computed using the conductivity specified in the material model feature (such as **Current Conservation**) that is applied in the domain. For **User defined**, enter a value for Q_e (SI unit: W/m^3) to specify a user-defined heat source.

Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or clicking **Harmonic Perturbation** in the **Physics** toolbar. For more information see Harmonic Perturbation — Exclusive and Contributing Nodes in the *COMSOL Multiphysics Reference Manual*.

EXTERNAL CURRENT DENSITY

Based on space dimension, enter the coordinates (**x**, **y**, and **z** for 3D components, for example) of the **External current density** $\mathbf{J}_e$ (SI unit: A/m^2). The defaults are 0 A/m^2.

Current Source

The **Current Source** node adds a distributed current source Q_j in the equation that the physics interface defines. Use this node with caution as it can violate the current conservation law that is inherent in Maxwell–Ampère's law.

Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or clicking **Harmonic Perturbation** in the **Physics** toolbar. For more information see Harmonic Perturbation — Exclusive and Contributing Nodes in the *COMSOL Multiphysics Reference Manual*.

CURRENT SOURCE

Enter a value or expression for the **Current source** Q_j (SI unit: $\mathrm{A/m^3}$). The default is $0\ \mathrm{A/m^3}$.

Electric Insulation

The **Electric Insulation** node, which is the default boundary condition, adds electric insulation as the boundary condition:

$$\mathbf{n} \cdot \mathbf{J} = 0$$

This boundary condition means that no electric current flows into the boundary. At interior boundaries, it means that no current can flow through the boundary and that the electric potential is discontinuous across the boundary. It is also applicable at symmetric boundaries where the potential is known to be symmetric with respect to the boundary.

Electric insulation as the default boundary condition is not applicable to interior boundaries. To add electric insulation to an interior boundary, add an **Electric Insulation** node in addition to the one that represents the default boundary condition.

Floating Potential

The **Floating Potential** node is used when modeling a metallic electrode at floating potential. It applies a constant voltage V_0 on the boundary (for domain features, this is the boundary enclosing the selected domain), such that the total normal electric current density $\mathbf{J}$ equals a specific current I_0:

$$V = V_0$$

$$\int_{\partial\Omega} (-\mathbf{n}\cdot\mathbf{J})dS = I_0,$$

where Ω represents the boundary and **n** refers to the surface normal. The constant boundary voltage implies the tangential electric field equals zero; the electric field will be perpendicular to the boundary:

$$-\nabla_t V = \mathbf{E}_t = 0$$

$$\mathbf{n}\times\mathbf{E} = 0.$$

In case of $I_0 = 0$ (the default case), the boundary will behave as an unconnected perfect conductor (a floating equipotential). This is a good approximation when the conductivity of the electrode is many orders of magnitude larger than that of the surrounding medium. Although locally the current density may vary, the total current entering or leaving the boundary equals zero.

If the floating potential touches a point, boundary, or domain feature that is not floating (a **Terminal** or **Ground** feature), the floating potential will acquire that feature's potential. If the floating potential is set to a certain current, or connected to a circuit, it behaves like a terminal.

The **Harmonic Perturbation** subnode (it is of the exclusive type) is available from the context menu (right-click the parent node) or in the **Physics** toolbar, click the **Attributes** menu and select **Harmonic Perturbation**. For more information see Harmonic Perturbation — Exclusive and Contributing Nodes in the *COMSOL Multiphysics Reference Manual*.

FLOATING POTENTIAL

The Floating potential identifier text area shows the unique identifier for the floating potential feature. It is used to identify the global variables created by the node, such as the voltage. The **Floating potential group** check box in the **Settings** window for **Floating Potential** controls how potentials are assigned to boundary segments in the feature's boundary selection. If this check box is not selected, a single potential is applied to all boundaries in the selection. If the check box is selected, each group of contiguous boundaries in the selection is given a unique potential. This simplifies the setup of models with many floating electrodes. The values of the potential at each group of boundaries are then made available in postprocessing, collected in a vector variable.

The following options are not available if the **Floating potential group** check box is selected. Select a specification for the **Floating potential current** — choose **User defined** (the default) to specify a total **Current** I_0 (SI unit: A) that flows from the electrode. The default is 0 A, corresponding to an unconnected electrode. Select **Circuit** to connect the floating potential to an Electrical Circuit interface.

Initial values

The initial values subsection is used to provide initial values for the internal state variables in the Floating Potential. These settings can be useful to provide an initial guess to the nonlinear solver or the initial value in a time-dependent study when the initial values of the other dependent variables (such as the electric potential) are computed from a separate physics or study. In most cases, the default values are appropriate.

Enter an **Initial value for voltage** V_{init} (SI unit: V). The default is 0 V.

CONSTRAINT SETTINGS

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

Settings and Properties Windows for Feature Nodes in the *COMSOL Multiphysics Reference Manual*

Boundary Current Source

The **Boundary Current Source** node adds a current source Q_{j} on the boundary.

$$\mathbf{n} \cdot (\mathbf{J}_1 - \mathbf{J}_2) = Q_{\text{j}}$$

It is applicable to interior boundaries that represent either a source or a sink of current. Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or click **Harmonic Perturbation** in the **Physics** toolbar. For more information see Harmonic Perturbation — Exclusive and Contributing Nodes in the *COMSOL Multiphysics Reference Manual*.

BOUNDARY CURRENT SOURCE

Enter a value or expression for the **Boundary current source** Q_{j} (SI unit: A/m^2). The default is 0 A/m^2.

Normal Current Density

The **Normal Current Density** node is applicable to exterior boundaries that represent either a source or a sink of current. It provides a condition for specifying the normal current density as an inward or outward current flow:

$$-\mathbf{n}\cdot\mathbf{J} = J_n$$

Or alternatively, as a current density $\mathbf{J}_0$:

$$\mathbf{n}\cdot\mathbf{J} = \mathbf{n}\cdot\mathbf{J}_0$$

The normal current density is positive when the current flows inward in the domain. Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or clicking **Harmonic Perturbation** in the **Physics** toolbar. For more information see Harmonic Perturbation — Exclusive and Contributing Nodes in the *COMSOL Multiphysics Reference Manual*.

NORMAL CURRENT DENSITY

Select a **Type** — **Inward current density** (the default) or **Current density**.

- For **Inward current density** enter a value or expression for the **Normal current density** J_n (SI unit: A/m^2). Use a positive value for an inward current flow or a negative value for an outward current flow. The default is 0 A/m^2.
- For **Current density** enter values or expressions for the components of the **Current density** $\mathbf{J}_0$ (SI unit: A/m^2). The defaults are 0 A/m^2.

Distributed Impedance

The **Distributed Impedance** node adds a distributed impedance boundary condition to a model.

The **Harmonic Perturbation** subnode (it is of the exclusive type) is available from the context menu (right-click the parent node) or in the **Physics** toolbar, click the **Attributes** menu and select **Harmonic Perturbation**. For more information see Harmonic Perturbation — Exclusive and Contributing Nodes in the *COMSOL Multiphysics Reference Manual*.

Use this boundary condition to model a thin sheet of a resistive material connected to a reference potential V_{ref}.

The layer impedance can be specified either with the bulk material conductivity σ_s, the relative permittivity ε_r and layer thickness d_s, or directly with the surface resistance ρ_s and capacitance C_s. Assuming DC currents, the equation is:

$$\mathbf{n}\cdot(\mathbf{J}_1-\mathbf{J}_2) = \frac{\sigma_s}{d_s}(V-V_{\mathrm{ref}})$$

$$\mathbf{n}\cdot(\mathbf{J}_1-\mathbf{J}_2) = \frac{1}{\rho_s}(V-V_{\mathrm{ref}})$$

For the Frequency Domain and Time Dependent study types, this boundary condition is slightly more sophisticated and accounts also for capacitive coupling. The equations are:

$$\mathbf{n}\cdot(\mathbf{J}_1-\mathbf{J}_2) = \frac{(\sigma+j\omega\varepsilon_0\varepsilon_{\mathrm{r}})}{d_s}(V-V_{\mathrm{ref}})$$

$$\mathbf{n}\cdot(\mathbf{J}_1-\mathbf{J}_2) = \left(\frac{1}{r_s}+j\omega C_s\right)(V-V_{\mathrm{ref}})$$

$$\mathbf{n}\cdot(\mathbf{J}_1-\mathbf{J}_2) = \frac{1}{d_L}\left(\sigma(V-V_{\mathrm{ref}})+\varepsilon_0\varepsilon_{\mathrm{r}}\frac{\partial}{\partial t}(V-V_{\mathrm{ref}})\right)$$

$$\mathbf{n}\cdot(\mathbf{J}_1-\mathbf{J}_2) = \left(\frac{1}{\rho_s}(V-V_{\mathrm{ref}})+C_s\frac{\partial}{\partial t}(V-V_{\mathrm{ref}})\right)$$

DISTRIBUTED IMPEDANCE

Enter the reference potential V_{ref} (SI unit: V). The default is 0 V.

Select a potentially complex-valued **Layer specification** — **Thin layer** (the default) or **Surface impedance**.

- For **Thin layer**, enter values or expressions for the:
  - **Surface thickness** d_{s} (SI unit: m). The default is $5\cdot10^{-3}$ m (5 mm).
  - **Electrical conductivity** σ (SI unit: S/m) and **Relative permittivity** ε_r (dimensionless). The defaults take values **From material**. For **User defined**, enter different values or expressions. The default electrical conductivity is $1\cdot10^{-2}$ S/m and the default relative permittivity is 1.
- For **Surface impedance**, enter values or expressions for the **Surface resistance** ρ_s (SI unit: $\Omega\cdot\mathrm{m}^2$) and the **Surface capacitance** C_{s} (SI unit: $\mathrm{F/m}^2$). The default surface impedance is $1\cdot10^{-8}$ $\Omega\cdot\mathrm{m}^2$ and the default surface capacitance is 0 $\mathrm{F/m}^2$.

Terminal

The **Terminal** node provides a boundary or domain condition for connection to external circuits, to transmission lines, or with a specified voltage or current. By specifying zero current, a floating potential condition is obtained.

> ⚠ In most cases, it is not recommended to include the interior of good conductors in the simulation. When using the **Terminal** on the domain level, the interior unknowns for the electric potential are replaced by a single constant, thus avoiding numerical problems arising from a very high material contrast. If one instead uses the **Terminal** on the boundary level, it is recommended to exclude the interior of the associated conductor from the domain selection for the main Electric Currents node.

The **Terminal** node also enables the computation of the lumped parameter of the system, such as the impedance.

The **Harmonic Perturbation** subnode (it is of the exclusive type) is available from the context menu (right-click the parent node) or in the **Physics** toolbar, click the **Attributes** menu and select **Harmonic Perturbation**. The **Harmonic Perturbation** subnode only has effect if **Current** or **Voltage** are selected as terminal type. For more information see Harmonic Perturbation — Exclusive and Contributing Nodes in the *COMSOL Multiphysics Reference Manual*.

An operation named as **Split by Connectivity** (✎) is available by right-clicking the parent node. The operation is useful when there are several terminal domains or boundaries that are not geometrically connected. Clicking the operation node will automatically generate several **Terminal** nodes where each selection is geometrically connected.

TERMINAL

Specify the terminal's properties. The **Terminal name** identifies the terminal and the variables it creates. The name should be numeric for sweeps to work properly.

Select a **Terminal type** — **Current** (the default), **Voltage**, **Circuit**, **Power**, **Terminated (power)**, and **Terminated (voltage)**. Select:

> ! The **Circuit** and **Power** types cannot be used together with a terminal sweep.

- **Current** to specify a total current flowing from the terminal. Enter the current I_0 (SI unit: A). The default is 0 A for an open circuit.
- **Voltage** to specify a fixed electric potential at the terminal. Enter the value of the electric potential V_0 (SI unit: V). The default is 1 V.
- **Circuit** to specify a terminal connected to an Electrical Circuit interface that models an external circuit.
- **Power** to specify a fixed electrical power for the terminal. The power is computed as the product of the terminal current and voltage and, in the frequency domain, is the cycle-averaged power. When **Power** is selected:
  - Enter a **Terminal power** P_0 (SI unit: W) to specify the input power at this terminal. The default is 1 W.
- **Terminated (power)** to connect the terminal to an impedance that might represent a load or a transmission line. When **Terminated (power)** is selected, the scattering parameters (S-parameters) are computed.
  - Enter a **Terminal power** P_0 (SI unit: W) to specify the input power at this terminal from the transmission line. This excitation can be toggled using a terminal sweep. The default is 1 W.
  - Select an option from the **Characteristic impedance from** list to define the value of the impedance — **Physics interface** (the default) or **User defined**. For **Physics interface** the **Reference impedance** Z_{ref} defined in the **Settings** window for the

physics interface under Sweep Settings is used. For **User defined** enter a specific impedance Z_{ref} (SI unit: Ω) for this terminal. The default is 50 Ω.

- **Terminated (voltage)** to connect the terminal to an impedance that might represent a load or a transmission line. The **Terminated (voltage)** option is essentially the same as **Terminated (power)**, with the power defined as: $P_0 = 0.5 \cdot |V_0|^2 / Z_{\mathrm{ref}}$.

!

Using Power excitation causes the equation system to become nonlinear. COMSOL Multiphysics automatically adjusts the solver settings when Power is selected.

Be aware that, in general, the values of the voltage and the current might not uniquely be determined by this constraint and may depend on the initial value provided for the current and voltage variables. Particularly in the frequency domain, the absolute phase of the quantities can be arbitrary. This indeterminacy can have an impact on the solution process. See Compile Equations in the *COMSOL Multiphysics Reference Manual* for a possible solution to this problem.

Initial Values

The initial values subsection is used to provide initial values for the internal state variables in the Terminal. These settings can be useful to provide an initial guess to the nonlinear solver or the initial value in a time-dependent study when the initial values of the other dependent variables (such as the electric potential) are computed from a separate physics or study. In most cases, the default values are appropriate.

The **Initial value for voltage** is available for all Terminal types except Voltage. Enter an **Initial value for voltage** V_{init} (SI unit: V). The default is 0 V.

The **Initial value for current** is available for Power only. In this case, it provides an initial guess for the nonlinear solver, and should not be set to zero. The initial value provided may affect the solution. Enter an **Initial value for current** I_{init} (SI unit: A). The default is 1 mA.

CONSTRAINT SETTINGS

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

Contact Impedance

Use the **Contact Impedance** node on interior boundaries to model a thin layer of resistive material. It can also be added as a pair using a **Pair Contact Impedance** node. The feature allows specifying the contact impedance either by entering the properties of the material together with the layer thickness, or by entering the impedance properties of the thin layer directly.

The feature, in stationary studies, applies the following conditions that relate the normal electric current density with the jump in the electric potential:

$$\mathbf{n}\cdot\mathbf{J}_1 = \frac{\sigma}{d_{\mathrm{s}}}(V_1 - V_2)$$

$$\mathbf{n}\cdot\mathbf{J}_2 = \frac{\sigma}{d_{\mathrm{s}}}(V_2 - V_1)$$

$$\mathbf{n}\cdot\mathbf{J}_1 = \frac{1}{\rho_s}(V_1 - V_2)$$

$$\mathbf{n}\cdot\mathbf{J}_2 = \frac{1}{\rho_s}(V_2 - V_1)$$

The first two equations refer to a layer impedance specified using the bulk material conductivity σ_s and the layer thickness d_s, while the last two equations refer to the case in which the surface resistance ρ_s is specified. The indices 1 and 2 refer to the two sides of the boundary. These parameters work the same as with Distributed Impedance.

For the frequency domain and Time Dependent study types, this boundary condition is slightly more sophisticated and accounts also for capacitive coupling. The bulk material relative permittivity ε_r or the layer capacitance C_s can be specified.

The equations used in Frequency Domain studies are:

$$\mathbf{n}\cdot\mathbf{J}_1 = \frac{(\sigma + j\omega\varepsilon_0\varepsilon_r)}{d_s}(V_1 - V_2)$$

$$\mathbf{n}\cdot\mathbf{J}_2 = \frac{(\sigma + j\omega\varepsilon_0\varepsilon_r)}{d_s}(V_2 - V_1)$$

$$\mathbf{n}\cdot\mathbf{J}_1 = \left(\frac{1}{\rho_s} + j\omega C_s\right)(V_1 - V_2)$$

$$\mathbf{n}\cdot\mathbf{J}_2 = \left(\frac{1}{\rho_s} + j\omega C_s\right)(V_2 - V_1)$$

In Time Dependent studies:

$$\mathbf{n}\cdot\mathbf{J}_1 = \frac{1}{d_s}\left(\sigma(V_1 - V_2) + \varepsilon_0\varepsilon_\mathrm{r}\frac{\partial}{\partial t}(V_1 - V_2)\right)$$

$$\mathbf{n}\cdot\mathbf{J}_2 = \frac{1}{d_s}\left(\sigma(V_2 - V_1) + \varepsilon_0\varepsilon_\mathrm{r}\frac{\partial}{\partial t}(V_2 - V_1)\right)$$

$$\mathbf{n}\cdot\mathbf{J}_1 = \left(\frac{1}{\rho_s}(V_1 - V_2) + C_s\frac{\partial}{\partial t}(V_1 - V_2)\right)$$

$$\mathbf{n}\cdot\mathbf{J}_2 = \left(\frac{1}{\rho_s}(V_2 - V_1) + C_s\frac{\partial}{\partial t}(V_2 - V_1)\right)$$

CONTACT IMPEDANCE

Select a potentially complex-valued **Layer specification** — **Thin layer** (the default) or **Surface impedance**.

- For **Thin layer**, enter values or expressions for the:
  - **Surface thickness** d_s (SI unit: m). The default is $5{\cdot}10^{-3}$ m (5 mm).
  - **Electrical conductivity** σ (SI unit: S/m) and **Relative permittivity** ε_r (dimensionless). The defaults take values **From material**. For **User defined**, enter different values or expressions. The default electrical conductivity is $1{\cdot}10^{-2}$ S/m and the default relative permittivity is 1.
- For **Surface impedance**, enter values or expressions for the **Surface resistance** ρ_s (SI unit: $\Omega{\cdot}\mathrm{m}^2$) and the **Surface capacitance** C_s (SI unit: $\mathrm{F/m}^2$). The default surface impedance is $1{\cdot}10^{-8}$ $\Omega{\cdot}\mathrm{m}^2$ and the default surface capacitance is 0 $\mathrm{F/m}^2$.

Thin-Film Resistance: Application Library path **COMSOL_Multiphysics/ Electromagnetics/thin_film_resistance**

Sector Symmetry

Select **Sector Symmetry** at interfaces between rotating objects where sector symmetry is used. It is only available for pairs. A default subnode is added. Right-click to select additional features from the **Fallback Features** submenu. In 2D, this feature assumes rotation around the origin.

This feature is always used in conjunction with a Periodic Condition on adjacent radial sector boundaries. Note that the same periodic condition feature cannot be used on both sides of where the sector symmetry boundaries connect with the periodic boundaries. At least two periodic condition features are required for the model to compute correctly.

PAIR SELECTION

When using nonconforming meshes on the source and destination of a pair, for numerical stability, a finer mesh should be applied on the destination side for any pair with a condition that imposes a coupling or a constraint across the pair. The sector symmetry feature falls into this category.

SECTOR SETTINGS

Enter the **Number of sectors (<50)** n_{sect}. The default is 2.

Select a **Type of periodicity** — **Continuity** (the default) or **Antiperiodicity**.

Based on space dimension, enter values or expressions in the table for the **Axis of rotation** $\mathbf{a}_{\mathrm{rot}}$.

CONSTRAINT SETTINGS

To display this section, click the **Show More Options** button ( ) and select **Advanced Physics Options** in the **Show More Options** dialog box. The constraint setting can be set to **Pointwise constraints** (default), or **Weak constraints**. Additionally, the **Nitsche constraints** option is available for the scalar potential in the **Electric Currents**, **Magnetic Field, No Currents**, and **Rotating Machinery, Magnetic** physics interfaces. The displacement current term is excluded from the **Nitsche constraints** in the **Electric Currents interface** and is therefore valid in the quasistationary approximation.

The **Nitsche constraints** option assumes the quasistatic approximation and is applicable only for pairs adjacent to domains with small or no displacement currents or induced eddy currents normal to the pair boundary. In addition, nonlinear material models in the neighboring domains are not supported.

Line Current Source

The **Line Current Source** node adds a line source to edges in 3D components and to points in 2D and 2D axisymmetric components. The line source represents electric current per unit length.

EDGE OR POINT SELECTION

⚠	Beware that constraining the potential on edges or points usually yields a current outflow that is mesh dependent.

LINE CURRENT SOURCE

Enter a value or expression to apply a **Line current source** Q_j (SI unit: A/m). This source represents electric current per unit length. The default is 0 A/m.

🔍	Line Current Source (on Axis) for 2D axisymmetric components.

Line Current Source (on Axis)

The **Line Current Source (on Axis)** node adds a line source to boundaries in 2D axisymmetric components. The line source represents electric current per unit length.

LINE CURRENT SOURCE (ON AXIS)

Enter a value or expression to apply a **Line current source** Q_j (SI unit: A/m) to boundaries. This source represents electric current per unit length.

🔍	Line Current Source

Point Current Source

The **Point Current Source** node adds a point source and represents an electric current flowing out of the point. Add point sources to 3D components from the **Points** menu. Add a contribution as a **Harmonic Perturbation** by right-clicking the parent node or clicking **Harmonic Perturbation** in the **Physics** toolbar. For more information see

Harmonic Perturbation — Exclusive and Contributing Nodes in the *COMSOL Multiphysics Reference Manual*.

POINT SELECTION

Beware that constraining the potential on points usually yields a current outflow that is mesh dependent.

POINT CURRENT SOURCE

Enter a value or expression to apply a **Point current source** Q_j (SI unit: A) to points. This source represents an electric current flowing out of the point.

- Line Current Source for 2D components
- Point Current Source (on Axis) for 2D axisymmetric components

Point Current Source (on Axis)

The **Point Current Source (on Axis)** node adds a point source and represents an electric current flowing out of the point in 2D axisymmetric components.

POINT SELECTION

Beware that constraining the potential on points usually yields a current outflow that is mesh dependent.

POINT CURRENT SOURCE

Enter a value or expression to apply a **Point current source** Q_j (SI unit: A) to points. This source represents an electric current flowing out of the point.

- Point Current Source for 3D components
- Line Current Source for 2D components

The Electrical Circuit Interface

The **Electrical Circuit (cir)** interface ( ), found under the **AC/DC** branch ( ) when adding a physics interface, is used to model currents and voltages in circuits including voltage and current sources, resistors, capacitors, inductors, and semiconductor devices. Models created with the Electrical Circuit interface can include connections to distributed field models. The physics interface supports stationary, frequency-domain and time-domain modeling and solves Kirchhoff's conservation laws for the voltages, currents, and charges associated with the circuit elements.

When this physics interface is added, it adds a default **Ground Node** feature and associates that with node zero in the electrical circuit.

! Circuit nodes are nodes in the electrical circuit (electrical nodes) and should not be confused with nodes in the Model Builder tree of the COMSOL Multiphysics software. Circuit node names are not restricted to numerical values but can contain alphanumeric characters.

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Refer to such physics interface variables in expressions using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `cir`.

RESISTANCE IN PARALLEL TO PN JUNCTIONS

For numerical stability, a large resistance is added automatically in parallel to the pn junctions in diodes and BJT devices. Enter a default value for the **Resistance in parallel to pn junctions** R_j (SI unit: Ω). The default value is $1\cdot 10^{12}$ Ω.

CREATE UNIQUE NODES FOR NEW DEVICES

When this setting is selected (the default), newly added devices will be assigned unused node names. The devices will be disconnected from the rest of the circuit and the nodes should be updated to reflect the actual circuit connections. When this setting is deselected, new devices will be connected to the lowest-numbered nodes starting from 0.

- Theory for the Electrical Circuit Interface
- Connecting to Electrical Circuits

Electrical Circuit Toolbar

The following nodes are available from the **Electrical Circuit** ribbon toolbar (Windows users), **Electrical Circuit** context menu (Mac or Linux users), or right-click to access the context menu (all users):

For step-by-step instructions and general documentation descriptions, this is the **Electrical Circuit** toolbar.

- Ground Node
- Voltmeter
- Ampère Meter
- Resistor
- Capacitor
- Inductor
- Voltage Source
- Current Source
- Diode
- Switch
- Voltage-Controlled Voltage Source[1]
- Voltage-Controlled Current Source[1]
- Current-Controlled Voltage Source[1]
- Current-Controlled Current Source[1]
- Subcircuit Definition
- Subcircuit Instance
- Mutual Inductance
- Transformer
- NPN BJT and PNP BJT[2]
- n-Channel MOSFET and p-Channel MOSFET[2]
- External I vs. U[3]
- External U vs. I[3]
- External I-Terminal[3]
- SPICE Circuit Import
- SPICE Circuit Export

[1] Selected from the **Dependent Sources** submenu when you right-click main node.

[2] Selected from the **Transistors** submenu when you right-click main node.

[3] Selected from the **External Couplings** submenu when you right-click main node.

Ground Node

The **Ground Node** (⏚) feature adds a ground node with the default node number zero to the electrical circuit. This is the default node in the Electrical Circuit interface. More ground nodes can be added but those must have unique node numbers and are by default given higher node numbers.

GROUND CONNECTION

Set the **Node name** for the ground node in the circuit. The convention is to use 0 (zero) for the ground node. If adding more ground nodes, each must have a unique node name (number).

Voltmeter

The **Voltmeter** (Ⓥ) feature connects a voltmeter (voltage measurement device) between two nodes in the electrical circuit. A voltmeter behaves electrically as an open circuit. The voltmeter node adds a Probe sampling the voltage across it.

NODE CONNECTIONS

Set the two **Node names** for the connecting nodes for the resistor.

Ampère Meter

The **Ammeter** (Ⓐ) feature connects an ammeter (current measurement device) between two nodes in the electrical circuit. An ammeter behaves electrically as a short circuit. The ammeter node adds a Probe sampling the current through it.

NODE CONNECTIONS

Set the two **Node names** for the connecting nodes for the resistor.

DEVICE PARAMETERS

Enter the **Resistance** of the resistor.

Resistor

The **Resistor** (▭) feature connects a resistor between two nodes in the electrical circuit.

NODE CONNECTIONS

Set the two **Node names** for the connecting nodes for the resistor.

DEVICE PARAMETERS

Enter the **Resistance** of the resistor.

Capacitor

The **Capacitor** ( ) feature connects a capacitor between two nodes in the electrical circuit.

NODE CONNECTIONS

Set the two **Node names** for the connecting nodes for the capacitor.

DEVICE PARAMETERS

Enter the **Capacitance** of the capacitor.

Inductor

The **Inductor** ( ) feature connects an inductor between two nodes in the electrical circuit.

NODE CONNECTIONS

Set the two **Node names** for the connecting nodes for the inductor.

DEVICE PARAMETERS

Enter the **Inductance** of the inductor.

Voltage Source

The **Voltage Source** ( ) feature connects a voltage source between two nodes in the electrical circuit.

NODE CONNECTIONS

Set the two **Node names** for the connecting nodes for the voltage source. The first node represents the positive reference terminal.

DEVICE PARAMETERS

Enter the **Source type** that should be adapted to the selected study type. It can be **General source**, **AC-source**, or a time-dependent **Sine source** or **Pulse source**. Depending on the choice of source, also specify the following parameters:

- For a General source, the **Voltage V_{src}** (default value: 1 V). General sources are active in Stationary, Time-Dependent and Frequency Domain studies.
- For an AC-source: the **Voltage V_{src}** (default value: 1 V) and the **Phase** Θ (default value: 0 rad). AC-sources are active in Frequency Domain studies only.
- For a sine source: the **Voltage V_{src}** (default value: 1 V), the **Offset V_{off}** (default value: 0 V), the **Frequency** (default value: 1 kHz), and the **Phase** Θ (default value: 0 rad). The sine sources are active in Time-Dependent studies and also in Stationary studies, providing that a value for t has been provided as a model parameter or global variable.
- For a pulse source: the **Voltage V_{src}** (default value: 1 V), the **Offset V_{off}** (default value: 0 V), the **Delay t_d** (default value: 0s), the **Rise time t_r** and **Fall time t_f** (default values: 0 s), the **Pulse width p_w** (default value: 1 μs), and the **Period T_{per}** (default value: 2 μs). The pulse sources are active in Time-Dependent studies and also in Stationary studies, providing that a value for t has been provided as a model parameter or global variable.

All values are peak values rather than RMS.

For the AC source, the frequency is a global input set by the solver. AC sources should be used in Frequency-domain studies only. Do not use the **Sine source** unless the model is time dependent.

Current Source

The **Current Source** ( ) feature connects a current source between two nodes in the electrical circuit.

NODE CONNECTIONS

Set the two **Node names** for the connecting nodes for the current source. The first node represents the positive reference terminal from where the current flows through the source to the second node.

DEVICE PARAMETERS

Enter the **Source type** that should be adapted to the selected study type. It can be **General source**, **AC-source**, or a time-dependent **Sine source** or **Pulse source**. Depending on the choice of source, also specify the following parameters:

- For a General source, the **Current** i_{src} (default value: 1 A). General sources are active in Stationary, Time-Dependent and Frequency Domain studies.
- For an AC-source: the **Current** i_{src} (default value: 1 A) and the **Phase** Θ (default value: 0 rad). AC-sources are active in Frequency Domain studies only.
- For a sine source: the **Current** i_{src} (default value: 1 A), the **Offset** i_{off} (default value: 0 A), the **Frequency** (default value: 1 kHz), and the **Phase** Θ (default value: 0 rad). The sine sources are active in Time-Dependent studies and also in Stationary studies, providing that a value for t has been provided as a model parameter or global variable.
- For a pulse source: the **Current** i_{src} (default value: 1 A), the **Offset** i_{off} (default value: 0 A), the **Delay** t_d (default value: 0 s), the **Rise time** t_r and **Fall time** t_f (default values: 0 s), the **Pulse width** p_w (default value: 1 μs), and the **Period** T_{per} (default value: 2 μs). The pulse sources are active in Time-Dependent studies and also in Stationary studies, providing that a value for t has been provided as a model parameter or global variable.

All values are peak values rather than RMS.

For the AC source, the frequency is a global input set by the solver. AC sources should be used in frequency-domain studies only. Do not use the **Sine source** unless the model is time dependent.

Voltage-Controlled Voltage Source

The **Voltage-Controlled Voltage Source** (◇) feature connects a voltage-controlled voltage source between two nodes in the electrical circuit. A second pair of nodes define the input control voltage.

NODE CONNECTIONS

Specify four **Node names**: the first pair for the connection nodes for the voltage source and the second pair defining the input control voltage. The first node in a pair represents the positive reference terminal.

DEVICE PARAMETERS

There are two options to define the relationship between the control voltage and resulting voltage. The **Use gain** method defines the resulting voltage to be the control voltage multiplied by the gain. The **Custom expression** method can define the relationship with an arbitrary expression.

Voltage-Controlled Current Source

The **Voltage-Controlled Current Source** (◇) feature connects a voltage-controlled current source between two nodes in the electrical circuit. A second pair of nodes define the input control voltage.

NODE CONNECTIONS

Specify four **Node names**: the first pair for the connection nodes for the current source and the second pair defining the input control voltage. The first node in a pair represents the positive voltage reference terminal or the one from where the current flows through the source to the second node.

DEVICE PARAMETERS

There are two options to define the relationship between the control voltage and resulting current. The **Use gain** method defines the resulting current to be the control voltage multiplied by the gain (SI units: S). The **Custom expression** method can define the relationship with an arbitrary expression.

Current-Controlled Voltage Source

The **Current-Controlled Voltage Source** (◇) feature connects a current-controlled voltage source between two nodes in the electrical circuit. The input control current is the one flowing through a two-pin device.

NODE CONNECTIONS

Set two **Node names** for the connection nodes for the voltage source. The first node in a pair represents the positive reference terminal.

DEVICE PARAMETERS

There are two options to define the relationship between the control current and resulting voltage. The **Use gain** method defines the resulting voltage to be the control current multiplied by the gain (SI units: Ω). The **Custom expression** method can define the relationship with an arbitrary expression.

Current-Controlled Current Source

The **Current-Controlled Current Source** ( ) feature connects a current-controlled current source between two nodes in the electrical circuit. The input control current is the one flowing through a named device that must be a two-pin device.

NODE CONNECTIONS

Specify two **Node names** for the connection nodes for the current source. The first node in a pair represents the positive reference terminal from where the current flows through the source to the second node.

DEVICE PARAMETERS

There are two options to define the relationship between the control current and resulting current. The **Use gain** method defines the resulting current to be the control current multiplied by the gain. The **Custom expression** method can define the relationship with an arbitrary expression.

Switch

The **Switch** ( ) feature is used to connect or disconnect the conducting path in a circuit under specific conditions.

NODE CONNECTIONS

Specify two **Node names** for the connection nodes for the current source. The first node in a pair represents the positive reference terminal from where the current flows through the source to the second node.

SWITCH CONDITIONS

There are three types of conditions, **Voltage controlled**, **Current controlled**, and **Custom expressions**. For each type of condition there are two conditions, one for turn on and one for turn off. The on condition is true if the **On condition** expression is larger than zero, while the off condition is true if the **Off condition** is less than zero.

The **Initial state** list has three options, **Use on condition**, **Use off condition**, and **Boolean expression**. The two former options mean that the switch will have an initial state matching to the on or off condition. The third option makes the switch's initial state match a custom Boolean expression. Separating on, off, and initial states makes the switch more flexible and can support Schmitt-trigger style switches and various latches.

For the **Voltage controlled** switch, it is necessary to specify two nodes that defines the voltage `sens.v` that the switch state depends on. The conditions must be written as a function of this variable. Similarly, for the **Current controlled** switch it is necessary to specify a two-pin device that defines the current `sens.i` that the switch state depends on.

SWITCH PARAMETERS

When the switch is in the on state it has a nonzero resistance specified by the **On resistance** expression. For the off state no current flows through the switch (infinite resistance). There is also a transition time for the switch to turn on and off set by the **Switching time** expression. The switch triggers an implicit event that updates a discrete state variable (with `suffix _state`).

Subcircuit Definition

The **Subcircuit Definition** ( ) feature is used to define subcircuits, which can be inserted as devices into the main circuit using Subcircuit Instance nodes. Create the subcircuit by adding subnodes to the **Subcircuit Definition** node, either by using the **Physics** toolbar, or by right-clicking the **Subcircuit Definition**.

SUBCIRCUIT PINS

Define the **Pin names** at which the subcircuit connects to the main circuit or to other subcircuits when referenced by a Subcircuit Instance node. The **Pin names** refer to circuit nodes in the subcircuit. The order in which the **Pin names** are defined is the order in which they are referenced by a Subcircuit Instance node. The devices constituting the subcircuit should be connected only to the subcircuit's pins and to themselves.

INPUT PARAMETERS

To display this section, click the **Show More Options** button ( ) and select **Advanced Physics Options**. Specify input parameters to a subcircuit that can be changed from a subcircuit instance. These input parameters can be used in all expression-style edit fields that affect the parameters of a device, for example, resistance, capacitance, and current gain. In this way, a subcircuit can represent a parameterized custom device model.

Subcircuit Instance

The **Subcircuit Instance** (IC) feature represents an instance of a subcircuits defined by a Subcircuit Definition feature.

NODE CONNECTIONS

Select the **Name of subcircuit link** from the list of defined subcircuits in the circuit model and the circuit **Node names** at which the subcircuit instance connects to the main circuit or to another subcircuit if used therein.

INPUT PARAMETERS

To display this section, click the **Show More Options** button (👁) and select **Advanced Physics Options**. Specify input parameters to a subcircuit that can be changed from a subcircuit instance. These input parameters can be used in all expression-style edit fields that affect the parameters of a device, for example, resistance, capacitance, and current gain. In this way, a subcircuit can represent a parameterized custom device model.

NPN BJT and PNP BJT

The **NPN BJT** ( ) and the **PNP BJT** ( ) device models are large-signal models for bipolar junction transistors (BJT). It is an advanced device model and no thorough description and motivation of the many input parameters are attempted here. Many device manufacturers provide model input parameters for this BJT model. For any particular make of BJT, the device manufacturer should be the primary source of information.

NODE CONNECTIONS

Specify three **Node names** for the connection nodes for the **BJT** device. These represent the *collector*, *base*, and *emitter* nodes for the **NPN** transistor, and the *emitter*, *base*, and *collector* nodes for the **PNP** transistor.

MODEL PARAMETERS

Specify the **Model Parameters**. Reasonable defaults are provided but for any particular BJT, the device manufacturer should be the primary source of information.

The interested reader is referred to Ref. 1 for more details on semiconductor modeling within circuits.

For an explanation of the **Model Parameters** see Bipolar Transistors.

n-Channel MOSFET and p-Channel MOSFET

The **n-Channel MOSFET** ( ) and the **p-Channel MOSFET** ( ) device models are large-signal models for, respectively, an n-Channel MOS field-effect transistor (MOSFET) and p-Channel MOSFET. These are advanced device models and no thorough description and motivation of the many input parameters are attempted here. Many device manufacturers provide model parameters for the MOSFET models. For any particular make of MOSFET, the device manufacturer should be the primary source of information.

NODE CONNECTIONS

Specify four **Node names** for the connection nodes for the **n-Channel MOSFET** or **p-Channel MOSFET** device. These represent the *drain*, *gate*, *source*, and *bulk* nodes, respectively.

MODEL PARAMETERS

Specify the **Model Parameters**. Reasonable defaults are provided but for any particular MOSFET, the device manufacturer should be the primary source of information.

The interested reader is referred to Ref. 1 for more details on semiconductor modeling within circuits.

For an explanation of the **Model Parameters** see MOSFET Transistors.

Mutual Inductance

The **Mutual Inductance** allows specifying a coupling between two existing Inductor features in the circuit. The mutual inductance of the coupling is

$$M = k\sqrt{L_1L_2}$$

where k is the coupling factor and L_1 and L_2 are the inductances of the inductors.

DEVICE PARAMETERS

Enter values or expressions for the:

- **Coupling factor** k (dimensionless). The value must be between 0 and 1, and the default is 0.98.
- **First inductance** L_1 (SI unit: H) and **Second inductance** L_2 (SI unit: H). These must be set to two different Inductor features in the circuit.

Transformer

The **Transformer** feature represents either a combination of two Inductor and a Mutual Inductance features, or an ideal transformer.

NODE CONNECTIONS

Enter or edit the table in the **Node names** column for the **primary** and **secondary** node connections.

DEVICE PARAMETERS

Choose a **Transformer model** — **Specify inductors** (the default) or **Ideal transformer**.

For **Specify inductors** enter values or expressions for the:

- **Coupling factor** k (dimensionless). The default is 0.98.
- **First inductance** L_1 (SI unit: H). The default is 1 mH.
- **Second inductance** L_2 (SI unit: H). The default is 1 mH.

For **Ideal transformer** enter values or expressions for the **Winding ratio** N_1/N_2 (dimensionless). The default is 10.

Diode

The **Diode** device model (◁) is a large-signal model for a diode. It is an advanced device model and no thorough description and motivation of the many input parameters are attempted here. The interested reader is referred to Ref. 1 for more details on semiconductor modeling within circuits. Many device manufacturers provide model parameters for this diode model. For any particular make of diode, the device manufacturer should be the primary source of information.

NODE CONNECTIONS

Specify two **Node names** for the positive and negative nodes for the **Diode** device.

MODEL PARAMETERS

Specify the **Model Parameters**. Reasonable defaults are provided but for any particular diode, the device manufacturer should be the primary source of information.

> For an explanation of the **Model Parameters** see Diode.

External I vs. U

The **External I vs. U** (I(U)) feature connects an arbitrary voltage measurement (for example, a circuit terminal or circuit port boundary or a coil domain from another physics interface) as a voltage source between two nodes in the electrical circuit. The resulting circuit current from the first node to the second node is typically coupled back as a prescribed current source in the context of the voltage measurement.

NODE CONNECTIONS

Specify the two **Node names** for the connecting nodes for the voltage source. The first node represents the positive reference terminal.

EXTERNAL DEVICE

Enter the source of the **Voltage**. If circuit or current excited terminals or circuit ports are defined on boundaries or domains or a multiturn coil domains is defined in other physics interfaces, these display as options in the **Voltage** list. Also select the **User defined** option and enter your own voltage variable, for example, using a suitable coupling operator. For inductive or electromagnetic wave propagation models, the voltage measurement must be performed as an integral of the electric field because the electric potential only does not capture induced EMF. Also the integration must be performed over a distance that is short compared to the local wavelength.

Except when coupling to a circuit terminal, circuit port, or coil, the current flow variable must be manually coupled back in the electrical circuit to the context of the voltage measurement. This applies also when coupling to a current excited terminal. The name of this current variable follows the convention `cirn.IvsUm_i`, where `cirn` is the tag of the Electrical Circuit interface node and `IvsUm` is the tag of the **External I vs. U** node. The tags are typically displayed within curly brackets `{}` in the Model Builder.

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

External U vs. I

The **External U vs. I** (U(I)) feature connects an arbitrary current measurement (for example, a coil domain from another physics interface) as a current source between two nodes in the electrical circuit. The resulting circuit voltage between the first node and the second node is typically coupled back as a prescribed voltage source in the context of the current measurement.

NODE CONNECTIONS

Specify the two **Node names** for the connecting nodes for the current source. The current flows from the first node to the second node.

EXTERNAL DEVICE

Enter the source of the **Current**. Voltage excited terminals or lumped ports defined on boundaries in other physics interfaces are natural candidates but do not appear as options in the **Voltage** list because those do not have an accurate built-in current measurement variable. A **User defined** option must be selected and a current variable entered, for example, using a suitable coupling operator.

> The voltage variable must be manually coupled back in the electrical circuit to the context of the current measurement. This applies also when coupling to a voltage excited terminal or lumped port. The name of this voltage variable follows the convention `cirn.UvsIm_v`, where `cirn` is the tag of the Electrical Circuit interface node and `UvsIm` is the tag of the **External U vs. I** node. The tags are typically displayed within curly brackets `{}` in the Model Builder.

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

External I-Terminal

The **External I-Terminal** ( ) feature connects an arbitrary voltage-to-ground measurement (for example, a circuit terminal from another physics interface) as a voltage-to-ground assignment to a node in the electrical circuit. The resulting circuit current from the node is typically coupled back as a prescribed current source in the context of the voltage measurement. This node does not apply when coupling to inductive or electromagnetic wave propagation models because then voltage must be defined as a line integral between two points rather than a single point measurement of electric potential. For such couplings, use the External I vs. U node instead.

NODE CONNECTIONS

Set the **Node name** for the connecting node for the voltage assignment.

EXTERNAL TERMINAL

Enter the source of the **Voltage**. If circuit- or current-excited terminals are defined on boundaries in other physics interfaces, these display as options in the **Voltage** list. Also select the **User defined** option and enter a voltage variable, for example, using a suitable coupling operator.

!

- Except when coupling to a circuit terminal, the current flow variable must be manually coupled back in the electrical circuit to the context of the voltage measurement. This applies also when coupling to a current excited terminal. The name of this current variable follows the convention `cirn.termIm_i`, where `cirn` is the tag of the Electrical Circuit interface node and `termIm` is the tag of the **External I-Terminal** node. The tags are typically displayed within curly brackets `{}` in the Model Builder.
- When connecting the finite element model between two circuit nodes that both are not grounded, the **External I-Terminal** node cannot be used. In this case, use External I vs. U or External U vs. I instead.

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

SPICE Circuit Import

Right-click the **Electrical Circuit** ( ) feature node to import an existing SPICE netlist (select **Import Spice Netlist**). A window opens — enter a file location or browse your directories to find one. The default file extension for a SPICE netlist is `.cir`. The SPICE circuit import translates the imported netlist into Electrical Circuit interface nodes so these define the subset of SPICE features that can be imported.

See SPICE Import and Export about the supported SPICE commands.

SPICE Circuit Export

Right-click the **Electrical Circuit** ( ) feature node to export the current circuit to the SPICE netlist file format (select **Export Spice Netlist**). A window opens — enter a file location or browse your directories to find one. The default file extension for a SPICE netlist is `.cir`. The compatible circuit nodes are exported as SPICE devices

See SPICE Export for more details on the supported SPICE commands.

Theory for the Electrostatics Interface

The Electrostatics Interface is available for 3D, 2D in-plane, and 2D axisymmetric components. Applications with electrostatic equations include high-voltage apparatus, electronic devices, and capacitors. The term “statics” is not to be interpreted literally — it is the observation time, or time scale at which the applied excitation changes, that is short compared to the charge relaxation time; also, the electromagnetic wavelength and skin depth are very large compared to the size of the domain of interest.

If you do not know whether to use the Electric Currents or the Electrostatics interface, which both solve for the scalar electric potential *V*, consider using an explicit charge transport model. See Charge Relaxation Theory.

Charge Relaxation Theory

COMSOL Multiphysics includes physics interfaces for the modeling of static electric fields and currents. Deciding what specific physics interface and study type to select for a particular modeling situation requires a basic understanding of the charge dynamics in conductors.

The different physics interfaces involving only the scalar electric potential can be interpreted in terms of the charge relaxation process. The fundamental equations involved are *Ohm’s law* for the conduction current density

$$\mathbf{J}_c = \sigma\mathbf{E}$$

the *equation of continuity*

$$\frac{\partial\rho}{\partial t} + \nabla\cdot\mathbf{J}_c = 0$$

and *Gauss’ law*

$$\nabla\cdot(\varepsilon\mathbf{E}) = \rho$$

By combining these, one can deduce the following differential equation for the space charge density in a homogeneous medium

$$\frac{\partial\rho}{\partial t} + \frac{\sigma}{\varepsilon}\rho = 0$$

This equation has the solution

$$\rho(t) = \rho_0 e^{-t/\tau}$$

where

$$\tau = \frac{\varepsilon}{\sigma}$$

is called the charge relaxation time. For a good conductor like copper, τ is of the order of 10^{-19} s, whereas for a good insulator like silica glass, it is of the order of 10^{3} s. For a pure insulator, it becomes infinite.

When modeling real-world devices, there is not only the *intrinsic time scale* of the charge relaxation time but also an *external time scale* t at which a device is energized or the observation time. It is the relation between the external time scale and the charge relaxation time that determines what physics interface and study type to use. The results are summarized in Table 11-3 below,

TABLE 11-3: SUITABLE PHYSICS INTERFACE AND STUDY TYPE FOR DIFFERENT TIME-SCALE REGIMES.

CASE	PHYSICS INTERFACE	STUDY TYPE
τ>>t	Electrostatics	Stationary
τ<<t	Electric Currents	Stationary
τ~t	Electric Currents	Time Dependent or Frequency Domain

FIRST CASE: τ >> T

If the external time scale is short compared to the charge relaxation time, the charges do not have time to redistribute to any significant degree. Thus the charge distribution can be considered as a given model input. The best approach is to solve the Electrostatics formulation using the electric potential V.

By combining the definition of the potential with Gauss' law, you can derive the classical Poisson's equation. Under static conditions, the electric potential V is defined by the equivalence $\mathbf{E} = -\nabla V$. Using this together with the constitutive relation $\mathbf{D} = \varepsilon_0\mathbf{E} + \mathbf{P}$ between $\mathbf{D}$ and $\mathbf{E}$, you can rewrite Gauss' law as a variant of Poisson's equation

$$-\nabla\cdot(\varepsilon_0\nabla V - \mathbf{P}) = \rho$$

This equation is used in the Electrostatics interface. It is worth noting that Gauss' law does not require the charge distribution to be static. Thus, provided dynamics are slow enough that induced electric fields can be neglected and hence a scalar electric potential is justified, the formulation can be used also in the Time Dependent study type. That typically involves either prescribing the charge dynamics or coupling a separate formulation for this.

Such separate charge transport formulations can be found in the Plasma Module, the Semiconductor Module, and the Chemical Reaction Engineering Module.

SECOND CASE: $\tau << T$

If the external time scale is long compared to the charge relaxation time, the stationary solution to the equation of continuity has been reached. In a stationary coordinate system, a slightly more general form of Ohm's law than above states that

$$\mathbf{J}_c = \sigma\mathbf{E} + \mathbf{J}_e$$

where $\mathbf{J}_\mathrm{e}$ is an externally generated current density. The static form of the equation of continuity then reads

$$\nabla \cdot \mathbf{J}_c = -\nabla \cdot (\sigma\nabla V - \mathbf{J}_e) = 0$$

To handle current sources, the equation can be generalized to

$$-\nabla \cdot (\sigma\nabla V - \mathbf{J}_e) = Q_j$$

This equation is used in the static study type for the Electric Currents interface.

Electrostatics Equations

Under static conditions, the electric potential, V, is defined by the relationship:

$$\mathbf{E} = -\nabla V$$

Combining this equation with the constitutive relationship $\mathbf{D} = \varepsilon_0\mathbf{E} + \mathbf{P}$ between the electric displacement $\mathbf{D}$ and the electric field $\mathbf{E}$, it is possible to represent Gauss' law as the following equation:

$$-\nabla \cdot (\varepsilon_0\nabla V - \mathbf{P}) = \rho$$

In this equation, the physical constant, ε_0 (SI unit: F/m) is the permittivity of vacuum, $\mathbf{P}$ (SI unit: C/m^2) is the electric polarization vector, and ρ (SI unit: C/m^3) is a space charge density. This equation describes the electrostatic field in dielectric materials.

For in-plane 2D modeling, the Electrostatics interface assumes a symmetry where the electric potential varies only in the x and y directions and is constant in the z direction. This implies that the electric field, $\mathbf{E}$, is tangential to the xy-plane. With this symmetry, the same equation is solved as in the 3D case. The physics interface solves the following equation where d is the thickness in the z direction:

$$-\nabla\cdot d(\varepsilon_0\nabla V-\mathbf{P}) = \rho$$

The axisymmetric version of the physics interface considers the situation where the fields and geometry are axially symmetric. In this case, the electric potential is constant in the φ direction, which implies that the electric field is tangential to the rz-plane.

The Electrostatics Interface in Time Dependent or Frequency Domain Studies

The Electrostatics Interface can also be solved in a dynamic study (Time Dependent or Frequency Domain). The equation system solved, however, is typically always the one presented in the previous section for the stationary case, in which no transient electromagnetic effects are taken into account. The difference is that the sources of the problem (charge densities, electric potential) are assumed to be time-varying (in a Time Dependent study) or time-harmonic (in a Frequency Domain study).

! Users that have a license supporting the piezoelectric material model can also include conduction losses in the frequency domain.

The support for dynamic studies simplifies the coupling of the Electrostatics interface with other physics interfaces. Using the physics interface in a dynamic study is a valid approximation only if the time-scale (or the frequency) of the study is so slow that transient electromagnetic effects can be neglected; for example, in acoustic or structural problems.

Most of the features in the Electrostatics interface have the same behavior in all study types, with the important exception of the Terminal node.

The Electrostatics interface also supports the small-signal analysis study sequence, which can be used when a time-harmonic perturbation is superposed on a static bias charge or voltage.

Theory for the Electric Currents Interface

The Electric Currents Interface solves a current conservation problem for the scalar electric potential V and is available for 3D, 2D in-plane, and 2D axisymmetric components. Electrolysis, conductors, and the computation of resistances of grounding plates are examples that involve conductive media with electrical conductivities and electric currents. If you are uncertain of whether to use the Electric Currents or the Electrostatics interface, which both solve for the scalar electric potential V, refer to the section on .

The Electric Currents Interface assumes that there is only one type of charge carrier. When there are multiple charge carriers, it is often necessary to solve the transport equation for each charge carrier as well as the Poisson's equation with Electrostatics interface.

In this section:

- Electric Currents Equations in Steady State
- Dynamic Electric Currents Equations

Electric Currents Equations in Steady State

When handling stationary electric currents in conductive media you must consider the stationary equation of continuity. In a stationary coordinate system, the point form of *Ohm's law* states that:

$$\mathbf{J} = \sigma\mathbf{E} + \mathbf{J}_\mathrm{e}$$

where σ is the electrical conductivity (SI unit: S/m), and $\mathbf{J}_\mathrm{e}$ is an externally generated current density (SI unit: $\mathrm{A/m^2}$). The static form of the equation of continuity then states:

$$\nabla \cdot \mathbf{J} = -\nabla \cdot (\sigma\nabla V - \mathbf{J}_\mathrm{e}) = 0$$

To handle current sources, you can generalize the equation to:

$$-\nabla \cdot (\sigma\nabla V - \mathbf{J}_\mathrm{e}) = Q_j$$

In planar 2D the Electric Currents interface assumes that the model has a symmetry where the electric potential varies only in the x and y directions and is constant in the z direction. This implies that the electric field, $\mathbf{E}$, is tangential to the xy-plane. The Electric Currents interface then solves the following equation, where d is the thickness in the z direction:

$$-\nabla\cdot d(\sigma\nabla V-\mathbf{J}_\mathrm{e}) = dQ_j \tag{11-1}$$

In 2D axisymmetry, the Electric Currents interface considers the situation where the fields and geometry are axially symmetric. In this case, the electric potential is constant in the φ direction, which implies that the electric field is tangential to the rz-plane.

Dynamic Electric Currents Equations

In the Frequency Domain and Time Dependent study types dynamic formulations accounting for both conduction currents and displacement currents are used.

Combining the time-harmonic equation of continuity

$$\nabla\cdot(\sigma\mathbf{E}+\mathbf{J}_e)+j\omega\rho = 0$$

with the equation

$$\nabla\cdot\mathbf{D} = \rho$$

and generalized to handle current sources yields the following equation:

$$-\nabla\cdot((\sigma+j\omega\varepsilon_0)\nabla V-(\mathbf{J}_e+j\omega\mathbf{P})) = Q_j$$

For the transient case, using the transient equation of continuity

$$\nabla\cdot(\sigma\mathbf{E}+\mathbf{J}_e)+\frac{\partial\rho}{\partial t} = 0$$

and generalized to handle current sources the resulting equation becomes:

$$-\nabla\cdot\frac{\partial}{\partial t}(\varepsilon_0\nabla V-\mathbf{P})-\nabla\cdot(\sigma\nabla V-\mathbf{J}_e) = Q_j \tag{11-2}$$

In planar 2D the dynamic formulations also involves the thickness d in the z direction:

$$-\nabla\cdot d((\sigma+j\omega\varepsilon_0)\nabla V-(\mathbf{J}_e+j\omega\mathbf{P})) = dQ_j$$

$$-\nabla\cdot d\frac{\partial}{\partial t}((\varepsilon_0\nabla V-\mathbf{P})-\nabla\cdot d(\sigma\nabla V-\mathbf{J}_e)) = dQ_j$$

Theory for the Electrical Circuit Interface

The Electrical Circuit Interface theory is discussed in this section:

- Electric Circuit Modeling and the Semiconductor Device Models
- Bipolar Transistors
- MOSFET Transistors
- Diode
- Reference for the Electrical Circuit Interface

Connecting to Electrical Circuits

Electric Circuit Modeling and the Semiconductor Device Models

Electrical circuit modeling capabilities are useful when simulating all sorts of electrical and electromechanical devices ranging from heaters and motors to advanced plasma reactors in the semiconductor industry. There are two fundamental ways that an electrical circuit model relates to a physical field model.

- The field model is used to get a better, more accurate description of a single device in the electrical circuit model.
- The electrical circuit is used to drive or terminate the device in the field model in such a way that it makes more sense to simulate both as a tightly coupled system.

The Electrical Circuit interface makes it possible to add nodes representing circuit elements directly to the Model Builder tree in a COMSOL Multiphysics model. The circuit variables can then be connected to a physical device model to perform co-simulations of circuits and multiphysics. The model acts as a device connected to the circuit so that its behavior is analyzed in larger systems.

The fundamental equations solved by the Electrical Circuit interface are Kirchhoff's circuit laws, which in turn can be deduced from Maxwell's equations. The supported study types are Stationary, Frequency Domain, and Time Dependent.

There are three more advanced large-signal semiconductor device features available in the Electrical Circuit interface. The equivalent circuits and the equations defining their nonideal circuit elements are described in this section. For a more complete treatise on semiconductor device modeling, see Ref. 1.

Bipolar Transistors

Figure 11-1 illustrates the equivalent circuit for the npn bipolar junction transistor.

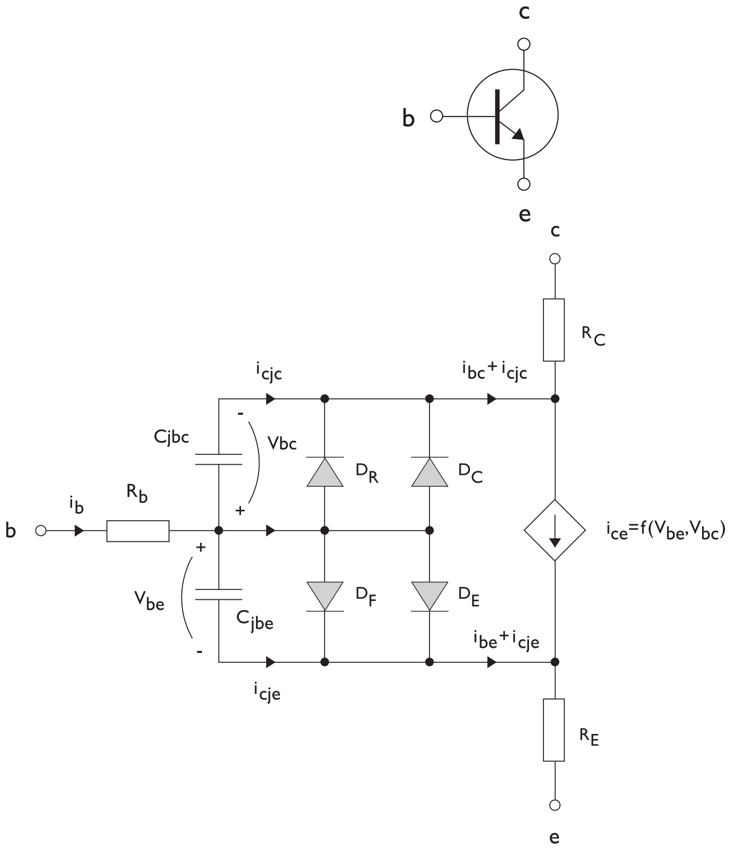


Figure 11-1: A circuit for the bipolar transistor.

The pnp transistor model is similar in all regards to the npn transistor, with the difference that the polarities of the currents and voltages involved are reversed. The following equations are used to compute the relations between currents and voltages in the circuit.

$$v_{rb} = \frac{1}{A}\left(R_{BM} - \frac{R_B - R_{BM}}{f_{bq}}\right)i_b$$

$$f_{bq} = \frac{1}{2\left(1 - \frac{v_{bc}}{V_{AF}} - \frac{v_{be}}{V_{AR}}\right)}\left(1 + \sqrt{1 + 4I_S\left(\frac{e^{\frac{v_{be}}{N_F V_T}} - 1}{I_{KF}A} + \frac{e^{\frac{v_{bc}}{N_R V_T}} - 1}{I_{KR}A}\right)}\right)$$

$$i_{be} = A\left(\frac{I_S}{B_F}\left(e^{\frac{v_{be}}{N_F V_T}} - 1\right) + I_{SE}\left(e^{\frac{v_{be}}{N_E V_T}} - 1\right)\right)$$

$$i_{bc} = A\left(\frac{I_S}{B_R}\left(e^{\frac{v_{bc}}{N_R V_T}} - 1\right) + I_{SC}\left(e^{\frac{v_{bc}}{N_C V_T}} - 1\right)\right)$$

$$i_{ce} = A\left(\frac{I_S}{f_{bq}}\left(e^{\frac{v_{be}}{N_F V_T}} + e^{\frac{v_{bc}}{N_C V_T}}\right)\right)$$

$$V_T = \frac{k_B T_{NOM}}{q}$$

There are also two capacitances that use the same formula as the junction capacitance of the diode model. In the parameter names below, replace *x* with *C* for the base-collector capacitance and *E* for the base-emitter capacitance.

$$C_{jbx} = AC_{Jx} \times \begin{cases} \left(1 - \frac{v_{bx}}{V_{Jx}}\right)^{-M_{Jx}} & v_{bx} < F_C V_{Jx} \\ (1 - F_C)^{-1 - M_{Jx}}\left(1 - F_C(1 + M_{Jx}) + M_{Jx}\frac{v_{bx}}{V_{Jx}}\right) & v_{bx} \geq F_C V_{Jx} \end{cases}$$

The model parameters are listed in the table below.

TABLE 11-1: BIPOLAR TRANSISTOR MODEL PARAMETERS.

PARAMETER	DEFAULT	DESCRIPTION
B_F	100	Ideal forward current gain
B_R	1	Ideal reverse current gain
C_{JC}	0 F/m^2	Base-collector zero-bias depletion capacitance
C_{JE}	0 F/m^2	Base-emitter zero-bias depletion capacitance
F_C	0.5	Breakdown current
I_{KF}	Inf (A/m^2)	Corner for forward high-current roll-off
I_{KR}	Inf (A/m^2)	Corner for reverse high-current roll-off

TABLE 11-1: BIPOLAR TRANSISTOR MODEL PARAMETERS.

PARAMETER	DEFAULT	DESCRIPTION
I_S	10^{-15} A/m^2	Saturation current
I_SC	0 A/m^2	Base-collector leakage saturation current
I_SE	0 A/m^2	Base-emitter leakage saturation current
M_{JC}	1/3	Base-collector grading coefficient
M_{JE}	1/3	Base-emitter grading coefficient
N_C	2	Base-collector ideality factor
N_E	1.4	Base-emitter ideality factor
N_F	1	Forward ideality factor
N_R	1	Reverse ideality factor
R_B	0 Ω·m^2	Base resistance
R_{BM}	0 Ω·m^2	Minimum base resistance
R_C	0 Ω·m^2	Collector resistance
R_E	0 Ω·m^2	Emitter resistance
T_{NOM}	298.15 K	Device temperature
V_{AF}	Inf (V)	Forward Early voltage
V_{AR}	Inf (V)	Reverse Early voltage
V_{JC}	0.71 V	Base-collector built-in potential
V_{JE}	0.71 V	Base-emitter built-in potential

MOSFET Transistors

Figure 11-2 illustrates an equivalent circuit for the n-channel MOSFET transistor. The p-channel MOSFET transistor is treated similarly, but the polarities of the involved voltages are reversed.

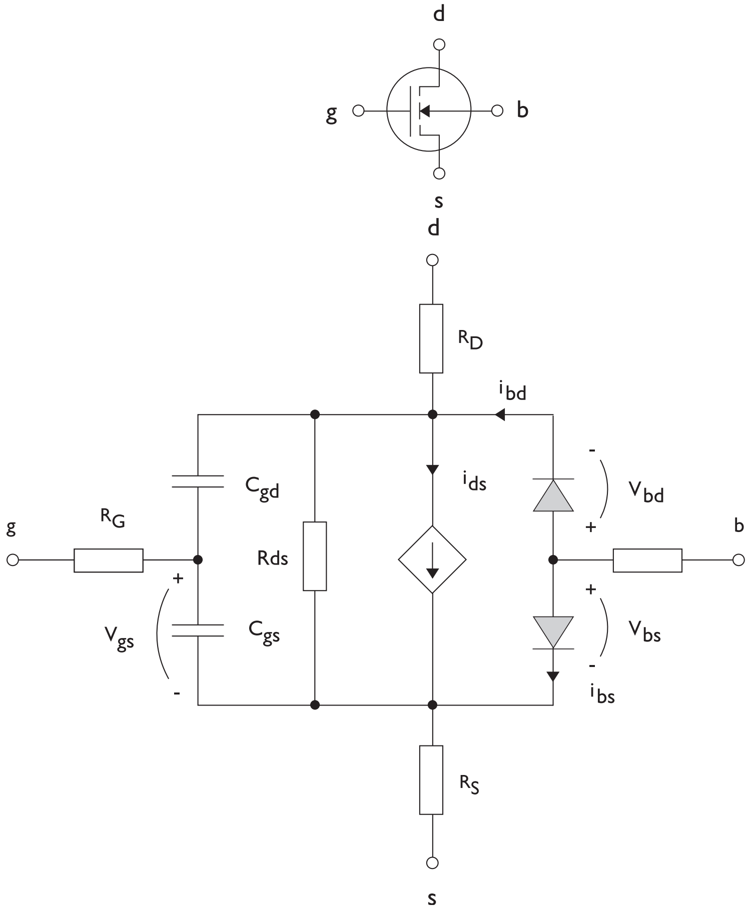


Figure 11-2: A circuit for the MOSFET transistor.

The following equations are used to compute the relations between currents and voltages in the circuit.

$$i_{ds} = \begin{cases} \frac{W}{L}\frac{K_P}{2}(1+\Lambda v_{ds})v_{ds}(2v_{th}-v_{ds}) & v_{ds} < v_{th} \\ \frac{W}{L}\frac{K_P}{2}(1+\Lambda v_{ds})v_{th}^2 & v_{ds} \geq v_{th} \\ 0 & v_{ds} < v_{th} \leq 0 \end{cases}$$

$$v_{th} = v_{gs} - (V_{TO} + \Gamma(\sqrt{\Phi - v_{bs}} - \sqrt{\Phi}))$$

$$i_{bd} = I_S\left(e^{\frac{v_{bd}}{NV_T}} - 1\right)$$

$$i_{bs} = I_S\left(e^{\frac{v_{bs}}{NV_T}} - 1\right)$$

$$V_T = \frac{k_B T_{NOM}}{q}$$

There are also several capacitances between the terminals

$$C_{gd} = C_{gd0}W$$
$$C_{gs} = C_{gs0}W$$

$$C_{jbd} = C_{BD} \times \begin{cases} \left(1 - \frac{v_{bd}}{P_B}\right)^{-M_J} & v_{bx} < F_C P_B \\ (1-F_C)^{-1-M_J}\left(1 - F_C(1+M_J) + M_J\frac{v_{bx}}{P_B}\right) & v_{bx} \geq F_C P_B \end{cases}$$

The model parameters are as follows:

TABLE 11-2: MOSFET TRANSISTOR MODEL PARAMETERS.

PARAMETER	DEFAULT	DESCRIPTION
C_{BD}	0 F/m	Bulk-drain zero-bias capacitance
C_{GDO}	0 F/m	Gate-drain overlap capacitance
C_{GSO}	0 F/m	Gate-source overlap capacitance
F_C	0.5	Capacitance factor
I_S	1e-13 A	Bulk junction saturation current
K_P	2e-5 A/V^2	Transconductance parameter
L	50e-6 m	Gate length
M_J	0.5	Bulk junction grading coefficient

TABLE 11-2: MOSFET TRANSISTOR MODEL PARAMETERS.

PARAMETER	DEFAULT	DESCRIPTION
N	1	Bulk junction ideality factor
P_B	0.75 V	Bulk junction potential
R_B	0 Ω	Bulk resistance
R_D	0 Ω	Drain resistance
R_{DS}	Inf (Ω)	Drain-source resistance
R_G	0 Ω	Gate resistance
R_S	0 Ω	Source resistance
T_{NOM}	298.15 K	Device temperature
V_{TO}	0 V	Zero-bias threshold voltage
W	50e-6 m	Gate width
Γ (GAMMA)	1 $V^{0.5}$	Bulk threshold parameter
Φ (PHI)	0.5 V	Surface potential
Λ (LAMBDA)	0 1/V	Channel-length modulation

Diode

Figure 11-3 illustrates equivalent circuit for the diode.

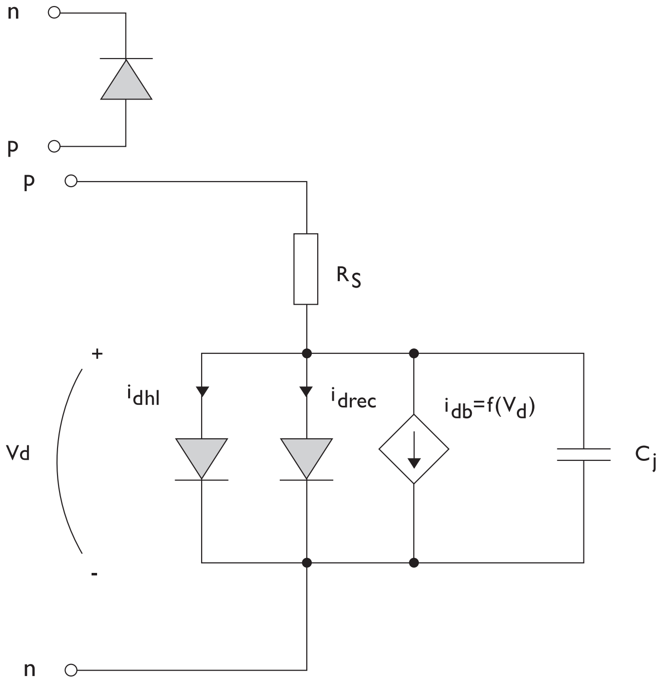


Figure 11-3: A circuit for the diode.

The following equations are used to compute the relations between currents and voltages in the circuit.

$$i_d = i_{dhl} + i_{drec} + i_{db} + i_c$$

$$i_{dhl} = I_S\left(e^{\frac{v_d}{NV_T}} - 1\right)\frac{1}{\sqrt{1 + \frac{I_S}{I_{KF}}\left(e^{\frac{v_d}{NV_T}} - 1\right)}}$$

$$i_{drec} = I_{SR}\left(e^{\frac{v_d}{N_R V_T}} - 1\right)$$

$$i_{db} = I_{BV} e^{-\frac{v_d + B_V}{N_{BV} V_T}}$$

$$C_j = C_{J0} \times \begin{cases} \left(1 - \frac{v_d}{V_J}\right)^{-M} & v_d < F_C V_J \\ (1 - F_C)^{-1-M}\left(1 - F_C(1+M) + M\frac{v_d}{V_J}\right) & v_d \ge F_C V_J \end{cases}$$

$$V_T = \frac{k_B T_{NOM}}{q}$$

where the following model parameters are required.

TABLE 11-3: DIODE TRANSISTOR MODEL PARAMETERS.

PARAMETER	DEFAULT	DESCRIPTION
B_V	Inf (V)	Reverse breakdown voltage
C_{J0}	0 F	Zero-bias junction capacitance
F_C	0.5	Forward-bias capacitance coefficient
I_{BV}	1e-09 A	Current at breakdown voltage
I_{KF}	Inf (A)	Corner for high-current roll-off
I_{S}	1e-13 A	Saturation current
M	0.5	Grading coefficient
N	1	Ideality factor
N_{BV}	1	Breakdown ideality factor
N_R	2	Recombination ideality factor
R_S	0 Ω	Series resistance
T_{NOM}	298.15 K	Device temperature
V_J	1.0 V	Junction potential

Reference for the Electrical Circuit Interface

1. P. Antognetti and G. Massobrio, *Semiconductor Device Modeling with Spice*, 2nd ed., McGraw Hill, 1993.

Connecting to Electrical Circuits

In this section:

About Connecting Electrical Circuits to Physics Interfaces

This section describes the various ways electrical circuits can be connected to other physics interfaces in COMSOL Multiphysics. If you are not familiar with circuit modeling, it is recommended that you review the Theory for the Electrical Circuit Interface.

In general electrical circuits connect to other physics interfaces via one or more of three special circuit features:

- External I vs. U
- External U vs. I
- External I-Terminal

These features either accept a voltage measurement from the connecting noncircuit physics interface and return a current from an Electrical Circuit interface or the other way around.

The “External” features are considered “ideal” current or voltage sources by the Electrical Circuit interface. Hence, you cannot connect them directly in parallel (voltage sources) or in series (current sources) with other ideal sources. This results in the error message *The DAE is structurally inconsistent*. A workaround is to provide a suitable parallel or series resistor, which can be tuned to minimize its influence on the results.

Connecting Electrical Circuits Using Predefined Couplings

In addition to these circuit features, interfaces in the AC/DC Module, RF Module, MEMS Module, Plasma Module, and Semiconductor Module (the modules that include the Electrical Circuit interface) also contain features that provide couplings to the Electrical Circuit interface by accepting a voltage or a current from one of the specific circuit features (External I vs. U, External U vs. I, and External I-Terminal).

This coupling is typically activated when:

- A choice is made in the **Settings** window for the noncircuit physics interface feature, which then announces (that is, includes) the coupling to the Electrical Circuit interface. Its voltage or current is then included to make it visible to the connecting circuit feature.
- A voltage or current that has been announced (that is, included) is selected in a feature node’s **Settings** window.

These circuit connections are supported in Terminals.

Connecting Electrical Circuits by User-Defined Couplings

A more general way to connect a physics interface to the Electrical Circuit interface is to:

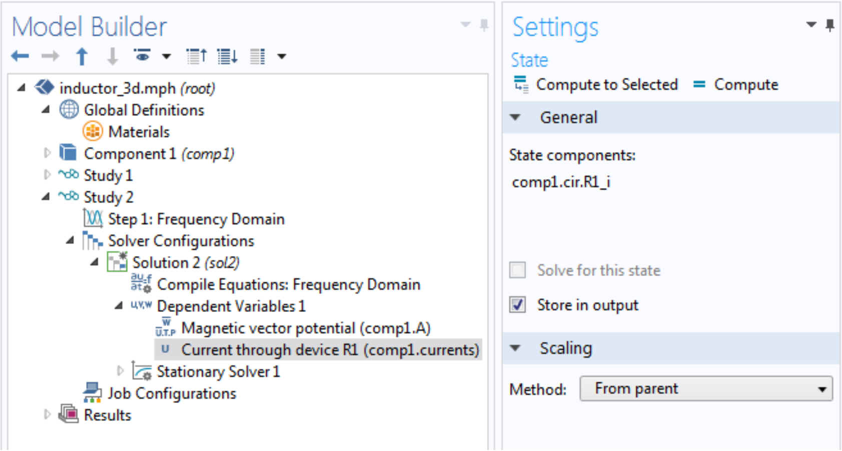
- Apply the voltage or current from the connecting “External” circuit feature as an excitation in the noncircuit physics interface.
- Define your own voltage or current measurement in the noncircuit physics interface using variables, coupling operators and so forth.
- In the **Settings** window for the Electrical Circuit interface feature, selecting the User-defined option and entering the name of the variable or expression using coupling operators defined in the previous step.

DETERMINING A CURRENT OR VOLTAGE VARIABLE NAME

To determine a current or voltage variable name, look at the **Dependent Variables** node under the **Study** node. To do this:

1 In the **Model Builder**, right-click the **Study** node and select **Show Default Solver**.

2 Expand the **Solver>Dependent Variables** node and click the state node, in this example, **Current through device R1 (comp1.currents)**. The variable name is shown in the **Settings** window for **State**.



Model Builder

- inductor_3d.mph *(root)*
  - Global Definitions
    - Materials
  - Component 1 *(comp1)*
  - Study 1
  - Study 2
    - Step 1: Frequency Domain
    - Solver Configurations
      - Solution 2 *(sol2)*
        - Compile Equations: Frequency Domain
        - Dependent Variables 1
          - Magnetic vector potential (comp1.A)
          - Current through device R1 (comp1.currents)
        - Stationary Solver 1
    - Job Configurations
  - Results

Settings

State

Compute to Selected = Compute

General

State components:

comp1.cir.R1_i

☐ Solve for this state

☑ Store in output

Scaling

Method: From parent

Typically, voltage variables are named `cir.Xn_v` and current variables `cir.X`*n*`_i`, where *n* is the “External” device number — `1`, `2`, and so on.

Solving

Some modeling errors lead to the error message *The DAE is structurally inconsistent* being displayed when solving. This error typically occurs from having an open current loop, from connecting voltage sources in parallel, or connecting current sources in series.

In this respect, the predefined coupling features are also treated as (ideal) voltage or current sources. The remedy is to close current loops and to connect resistors in series with voltage sources or in parallel with current sources.

Postprocessing

The Electrical Circuits interface, unlike most of the other physics interfaces, solves for a relatively large number of global dependent variables (such as voltages and currents), instead of solving for a few space-varying fields (such as temperature or displacement). For this reason, the Electrical Circuit interface does not provide default plots when computing a study.

The physics interface defines a number of variables that can be used in postprocessing. All variables defined by the Electrical Circuit interface are of a global scope, and can be evaluated in a **Global Evaluation** node (under **Derived Values**). In addition, the time evolution or dependency on a parameter can be plotted in a **Global** plot (under a **1D Plot Group** node).

The physics interface defines a Node voltage variable for each electrical node in the circuit, with name `cir.v_name`, where `cir` is the physics interface Label and `<name>` is the node Name. For each two-pin component, the physics interface also defines variables containing the voltage across it and the current flowing through it. For resistors it adds a heat source variable as well.

In the *COMSOL Multiphysics Reference Manual:*

- Derived Values, Evaluation Groups, and Tables and Global Evaluation
- Plot Groups and Plots and Global

SPICE Import and Export

SPICE Import

The circuit definition in COMSOL Multiphysics adheres to the SPICE format developed at the University of California, Berkeley (Ref. 1). SPICE netlists can be imported and the corresponding circuit nodes are generated in the COMSOL Multiphysics model. Most circuit simulators can export to this format or some version of it.

The Electrical Circuit interface supports the following device models:

TABLE 11-4: SUPPORTED SPICE DEVICE MODELS.

STATEMENT	DEVICE MODEL
R	Resistor
C	Capacitor
L	Inductor
V	Voltage Source
I	Current Source
E	Voltage-Controlled Voltage Source
F	Current-Controlled Current Source
G	Voltage-Controlled Current Source
H	Current-Controlled Voltage Source
D	Diode
Q	NPN BJT and PNP BJT
M	n-Channel MOSFET and p-Channel MOSFET
X	Subcircuit Instance

Statements corresponding to multiple devices are resolved by parsing the associated `.model` statement. The physics interface also supports the `.subckt` statement, which is represented in COMSOL by a Subcircuit Definition node, and the `.include` statement. SPICE commands are interpreted case-insensitively. The statement defining each device is also interpreted as the **Device name**.

According to SPICE specification, the first line in the netlist file is assumed to be the title of the netlist and it is ignored by the parser.

SPICE Export

The **SPICE Export** functionality creates a SPICE netlist file containing a description of the circuit represented by the physics interface. This functionality can be accessed from the physics interface context menu (right-click the physics interface node and select **Export SPICE Netlist**). After specifying a filename, the circuit is exported and messages from the export process display in the **Messages** window. During the export process, a series of operations are performed:

- In order to avoid conflicts, each component must be identified by a unique **Device name**. If one or more components have the same device name, the export operation fails and an error message is displayed. All characters in a **Device name** that are not letters, digits, or underscores are replaced by underscores.
- According to the SPICE specification, each circuit must have a node with name `0`, which is assumed to be the only ground node. When exporting a circuit, any node with name `0` that is not connected to a **Ground** component is exported with a different node name. All nodes that are connected to a Ground components are exported as a merged node with name `0`. The Messages window shows a log message if these operations are performed, showing the name of the renamed or merged nodes.
- All characters in node names that are not letters, digits, or underscores are replaced by underscores.
- Some components (most notably, the **External** components used to couple to other physics interfaces) cannot be exported to a SPICE netlist. These components are ignored during the export process, and a message is shown in the Messages window. Note that this can change the exported circuit, since some components are then missing.
- Subcircuit definitions are added as `.subckt` statements in the netlist. Semiconductor devices (such as MOSFETs, BJTs, and diodes) are exported as a SPICE device with a corresponding `.model` statement.

The title of the exported netlist file is the model's filename, and the time, date, and version of COMSOL Multiphysics is added as a comment in the netlist file.

Reference

1. http://bwrcs.eecs.berkeley.edu/Classes/IcBook/SPICE/

12

Fluid Flow Interface

This chapter has information about the enhanced Laminar Flow interface found under the **Fluid Flow>Single-Phase Flow** branch (≋) when adding a physics interface.

In this chapter:

The Single-Phase Flow, Laminar Flow Interface

In this section:

- The Laminar Flow Interface
- Domain, Boundary, Pair, and Point Nodes for Single-Phase Flow

- Theory for the Single-Phase Flow Laminar Interface

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

The Laminar Flow Interface

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

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

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

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

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

SETTINGS

The **Label** is the default physics interface name.

The **Name** is used primarily as a scope prefix for variables defined by the physics interface. Physics interface variables can be referred to using the pattern `<name>.<variable_name>`. In order to distinguish between variables belonging to different physics interfaces, the `name` string must be unique. Only letters, numbers, and underscores (_) are permitted in the **Name** field. The first character must be a letter.

The default **Name** (for the first physics interface in the model) is `spf`.

PHYSICAL MODEL

Compressibility

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

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

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

When the **Compressible flow (Ma<0.3)** option is selected, the compressible form of the Navier–Stokes and continuity equations is applied. $\mathrm{Ma} < 0.3$ indicates that the inlet and outlet conditions, as well as the stabilization, may not be suitable for transonic and supersonic flow. For more information, see The Mach Number Limit.

Porous Media Domains

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

Include Gravity

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

Also, when the **Include gravity** check box is selected, the **Use reduced pressure** option changes the pressure formulation from using the total pressure (default) to using the reduced pressure. This option is suitable for configurations where the density changes are very small; otherwise, the default formulation can be used. For more information, see Gravity

Reference Values

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

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

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

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

DEPENDENT VARIABLES

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

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

CONSISTENT STABILIZATION

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

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

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

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

INCONSISTENT STABILIZATION

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

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

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

- Numerical Stability — Stabilization Techniques for Fluid Flow

ADVANCED SETTINGS

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

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

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

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

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

DISCRETIZATION

The default discretization for Laminar Flow is **P1+P1** elements — that is, piecewise linear interpolation for velocity and pressure. This is suitable for most flow problems.

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

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

Flow Past a Cylinder: Application Library path **COMSOL_Multiphysics/Fluid_Dynamics/cylinder_flow**

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

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

- Boundary Stress
- Flow Continuity
- Fluid Properties
- Gravity
- Initial Values
- Inlet
- Line Mass Source[1]
- Open Boundary
- Outlet
- Periodic Flow Condition
- Point Mass Source[1]
- Porous Interface
- Pressure Point Constraint
- Symmetry
- Vacuum Pump
- Volume Force
- Wall

[1] A feature that may require an additional license

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

In the *COMSOL Multiphysics Reference Manual*, see Table 2-4 for links to common sections and Table 2-5 to common feature nodes. You can also search for information: press F1 to open the **Help** window or Ctrl+F1 to open the **Documentation** window.

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

Fluid Properties

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

MODEL INPUTS

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

Temperature

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

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

Absolute Pressure

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

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

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

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

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

FLUID PROPERTIES

Density

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

Dynamic Viscosity

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

Volume Force

The **Volume Force** node specifies the volume force **F** on the right-hand side of the momentum equation.

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

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

Initial Values

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

INITIAL VALUES

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

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

Wall

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

BOUNDARY CONDITION

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

- No Slip
- Slip
- Slip Velocity
- Leaking Wall

No Slip

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

Slip

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

Slip Velocity

In the microscale range, the flow condition at a boundary is seldom strictly no slip or slip. Instead, the boundary condition is something in between, and there is a **Slip velocity** at the boundary. Two phenomena account for this velocity: noncontinuum effects and the flow induced by a thermal gradient along the boundary.

When the **Use viscous slip** check box is selected, the default **Slip length** L_s is **User defined**. Another value or expression may be entered if the default value is not applicable. For **Maxwell's model** values or expressions for the **Tangential momentum accommodation coefficient** a_v and the **Mean free path** λ should be specified. Tangential accommodation coefficients are typically in the range of 0.85 to 1.0 and can be found in G. Kariadakis, A. Beskok, and N. Aluru, *Microflows and Nanoflows*, Springer Science and Business Media, 2005.

When the **Use thermal creep** check box is selected, a thermal creep contribution with **Thermal slip coefficient** σ_T is activated. Thermal slip coefficients are typically between 0.3 and 1.0 and can be found in G. Kariadakis, A. Beskok, and N. Aluru, *Microflows and Nanoflows*, Springer Science and Business Media, 2005.

Leaking Wall

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

WALL MOVEMENT

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

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

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

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

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

- For 3D components, values or expressions for the **Velocity of sliding wall** $\mathbf{u}_\mathrm{w}$ should be specified. If the velocity vector entered is not in the plane of the wall, COMSOL Multiphysics projects it onto the tangential direction. Its magnitude is adjusted to be the same as the magnitude of the vector entered.
- For 2D components, the tangential direction is unambiguously defined by the direction of the boundary. For this reason, the sliding wall boundary condition has different definitions in different space dimensions. A single entry for the **Velocity of the tangentially moving wall** U_w should be specified in 2D.

CONSTRAINT SETTINGS

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

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

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

- Theory for the Wall Boundary Condition
- The Moving Mesh Interface in the *COMSOL Multiphysics Reference Manual*

Inlet

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

Specifying the velocity vector at both the inlet and the outlet may cause convergence difficulties.

BOUNDARY CONDITION

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

VELOCITY

The **Normal inflow velocity** is specified as $\mathbf{u} = -\mathbf{n}\mathrm{U}_0$, where $\mathbf{n}$ is the boundary normal pointing out of the domain and U_0 is the normal inflow speed.

The **Velocity field** option sets the velocity vector to $\mathbf{u} = \mathbf{u}_{0.}$ The components of the inlet velocity vector $\mathbf{u}_0$ should be defined for this choice.

PRESSURE CONDITIONS

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

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

  When **Include gravity** is selected and **Use reduced pressure** not selected in the interface **Physical model** section, the **Compensate for hydrostatic pressure approximation** (named **Compensate for hydrostatic pressure** for incompressible flows) check box is available and selected by default. When it is selected, the hydrostatic pressure is automatically added to the pressure entered in p_0 user input.
- The **Suppress backflow** option adjusts the inlet pressure locally in order to reduce the amount of fluid exiting the domain through the boundary. If you clear the suppress backflow option, the inlet boundary can become an outlet depending on the pressure field in the rest of the domain.

- **Flow direction** controls in which direction the fluid enters the domain.
  - For **Normal flow**, it prescribes zero tangential velocity component.
  - For **User defined**, an **Inflow velocity direction** $\mathbf{d_u}$ (dimensionless) should be specified. The magnitude of $\mathbf{d_u}$ does not matter, only the direction. $\mathbf{d_u}$ must point into the domain.

MASS FLOW

The mass flow at an inlet can be specified by the **Mass flow rate**, the **Pointwise mass flux**, the **Standard flow rate**, or the **Standard flow rate (SCCM)**.

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

Mass Flow Rate

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

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

Pointwise Mass Flux

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

Standard Flow Rate

The **Standard flow rate** Q_{sv} sets a standard volumetric flow rate, according to the SEMI standard E12-0303. The mass flow rate is specified as the volumetric flow rate of a gas at standard density — the **Mean molar mass** M_n divided by a **Standard molar volume** V_{m} (that is, the volume of one mole of a perfect gas at standard pressure and standard temperature). The flow occurs across the whole boundary in the direction of the boundary normal and is computed by a surface (3D) or line (2D) integral. The tangential flow velocity is set to zero.

The standard density can be defined directly, or by specifying a standard pressure and temperature, in which case the ideal gas law is assumed. The options in the **Standard flow rate defined by** list are:

- **Standard density**, for which the **Standard molar volume** V_{m} should be specified.
- **Standard pressure and temperature**, for which the **Standard pressure** P_{st} and the **Standard temperature** T_{st} should be defined.

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

Standard Flow Rate (SCCM)

The **Standard flow rate (SCCM)** boundary condition is equivalent to the **Standard flow rate** boundary condition, except that the flow rate is entered directly in SCCMs (standard cubic centimeters per minute) without the requirement to specify units. Here, the dimensionless **Number of SCCM units** Q_{sccm} should be specified.

FULLY DEVELOPED FLOW

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

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

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

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

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

CONSTRAINT SETTINGS

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

- Prescribing Inlet and Outlet Conditions
- Normal Stress Boundary Condition

Fully Developed Flow (Inlet)

Outlet

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

BOUNDARY CONDITION

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

PRESSURE CONDITIONS

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

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

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

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

VELOCITY

See the **Inlet** node Velocity section for the settings.

FULLY DEVELOPED FLOW

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

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

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

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

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

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

CONSTRAINT SETTINGS

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

🔍 Prescribing Inlet and Outlet Conditions

🔍 Fully Developed Flow (Outlet)

Symmetry

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

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

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

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

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

$$\mathbf{u}\cdot\mathbf{n} = \mathbf{0}, \qquad \mathbf{K}-(\mathbf{K}\cdot\mathbf{n})\mathbf{n} = \mathbf{0}$$
$$\mathbf{K} = \mu(\nabla\mathbf{u}+(\nabla\mathbf{u})^{\mathrm{T}})\mathbf{n}$$

BOUNDARY SELECTION

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

CONSTRAINT SETTINGS

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

Open Boundary

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

BOUNDARY CONDITIONS

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

Normal Stress

The **Normal stress** f_0 condition implicitly imposes $p \approx \mathrm{f}_0$.

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

No Viscous Stress

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

The **No viscous stress** condition prescribes:

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

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

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

Boundary Stress

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

BOUNDARY CONDITION

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

General Stress

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

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

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

for the compressible and the incompressible formulations.

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

$$p = 2\mu\frac{\partial u_n}{\partial n}-\mathbf{n}\cdot\mathbf{F} \qquad (12\text{-}1)$$

If $\partial u_n/\partial n$ is small, Equation 12-1 states that $p \approx -\mathbf{n}\cdot\mathbf{F}$.

Normal Stress

Normal Stress is described for the Open Boundary node.

Normal Stress, Normal Flow

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

$$\mathbf{n}^T\left(-p\mathbf{I}+\left(\mu(\nabla\mathbf{u}+(\nabla\mathbf{u})^T)-\frac{2}{3}\mu(\nabla\cdot\mathbf{u})\mathbf{I}\right)\right)\mathbf{n} = -\mathrm{f}_0, \qquad \mathbf{t}\cdot\mathbf{u} = 0$$

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

for the compressible and the incompressible formulations.

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

$$p = 2\mu\frac{\partial u_n}{\partial n}+\mathrm{f}_0 \qquad (12\text{-}2)$$

If $\partial u_n/\partial n$ is small, Equation 12-2 states that $p \approx \mathrm{f}_0$.

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

CONSTRAINT SETTINGS

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

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

Vacuum Pump

The **Vacuum Pump** node models the effective outlet pressure created by a vacuum pump device that is attached to the outlet.

PARAMETERS

The **Flow condition** option defines a lumped curve — **linear**, **static pressure curve data**, or **user defined**.

Static Pressure Curve, Linear

For the **Static pressure curve, linear** option, values or expressions should be specified for the **Static pressure at no flow** p_{nf} and the **Free delivery flux** $V_{0,\mathrm{fd}}$.

The static pressure curve value is equal to the static pressure at no flow when $V_0 < 0$ and equal to 0 when the flux is larger than the free delivery flux.

Static Pressure Curve, Data

The **Static pressure curve, data** option can be used to enter or load data under the **Static Pressure Curve Data** section below. The interpolation between points given in the **Static Pressure Curve Data** table is defined using the **Interpolation function type** list in the **Static Pressure Curve Interpolation** section.

Static Pressure Curve, User Defined

The **User defined** option can be used to enter a different value or expression. The flux across the selection where this boundary condition is applied is defined by `*phys_id*.V0` where *`phys_id`* is the name (for example, *`phys_id`* is `spf` by default for this physics interface). In order to avoid unexpected behavior, the function used for the fan curve is the maximum of the user-defined function and 0.

STATIC PRESSURE CURVE DATA

This section displays when **Static pressure curve, data** is selected as the **Flow condition**. In the table, values or expressions for the **Flux** and **Static pressure curve** should be specified. Alternatively, a text file can be imported by clicking the **Load from file** button (📂) under the table.

UNITS

This section is available when **Static pressure curve, data** is selected as the **Flow condition**. Units can be specified here for the **Flux** and **Static pressure curve**.

STATIC PRESSURE CURVE INTERPOLATION

This section is available when **Static pressure curve, data** is selected as the **Flow condition**. The available options for the **Interpolation function type** are **Linear**, **Piecewise cubic**, and **Cubic spline**.

The extrapolation method always returns a constant value. In order to avoid problems with undefined data, the function used for the boundary condition is the maximum of the interpolated function and 0.

Vacuum Pump Boundary Condition

Periodic Flow Condition

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

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

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

BOUNDARY SELECTION

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

DESTINATION SELECTION

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

FLOW CONDITION

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

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

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

ORIENTATION OF SOURCE

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

ORIENTATION OF DESTINATION

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

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

MAPPING BETWEEN SOURCE AND DESTINATION

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

Porous Interface

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

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

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

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

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

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

Vena contracta model allows two options:

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

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

Flow Continuity

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

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

When the condition is set for pairs adjacent to domains with different porosity ε_p, it imposes conditions identical to the **Porous Interface** feature, which is designed to properly account for a sudden jump in porosity. See the section Porous Interface above for further details (note that the **Flow Continuity** does not account for the density-jump corrections),

About Identity and Contact Pairs

Pressure Point Constraint

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

PRESSURE CONSTRAINT

The relative pressure value is set by specifying the **Pressure** p_0. If the reference pressure p_ref defined at the physics interface level is equal to zero, p_0 represents the absolute pressure.

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

CONSTRAINT SETTINGS

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

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

Point Mass Source

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

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

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

SOURCE STRENGTH

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

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

Mass Sources for Fluid Flow in the *COMSOL Multiphysics Reference Manual*

Line Mass Source

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

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

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

SELECTION

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

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

SOURCE STRENGTH

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

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

Mass Sources for Fluid Flow in the *COMSOL Multiphysics Reference Manual*

Gravity

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

ACCELERATION OF GRAVITY

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

Theory for the Single-Phase Flow Laminar Interface

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

- Particle Tracing in Fluid Flow
- References for the Single-Phase Flow, Laminar Flow Interfaces

The theory about most boundary conditions is found in Ref. 2.

General Single-Phase Flow Theory

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

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \tag{12-3}$$

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u} \cdot \nabla)\mathbf{u} = \nabla \cdot [-p\mathbf{I} + \mathbf{K}] + \mathbf{F} \tag{12-4}$$

$$\rho C_p\left(\frac{\partial T}{\partial t} + (\mathbf{u} \cdot \nabla)T\right) = -(\nabla \cdot \mathbf{q}) + \mathbf{K}:\mathbf{S} - \frac{T}{\rho}\frac{\partial \rho}{\partial T}\bigg|_p\left(\frac{\partial p}{\partial t} + (\mathbf{u} \cdot \nabla)p\right) + Q \tag{12-5}$$

where

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

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

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

$$\mathbf{a}:\mathbf{b} = \sum_n \sum_m a_{nm} b_{nm} \tag{12-6}$$

This is sometimes referred to as the double dot product.

Equation 12-3 is the continuity equation and represents conservation of mass. Equation 12-4 is a vector equation which represents conservation of momentum. Equation 12-5 describes the conservation of energy, formulated in terms of temperature. This is an intuitive formulation that facilitates boundary condition specifications.

To close the equation system, Equation 12-3 through Equation 12-5, constitutive relations are needed.

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

$$\mathbf{K} = 2\mu\mathbf{S} - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I} \tag{12-7}$$

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

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

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

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

Many applications describe isothermal flows for which Equation 12-5 is decoupled from Equation 12-3 and Equation 12-4.

2D AXISYMMETRIC FORMULATIONS

A 2D axisymmetric formulation of Equation 12-3 and Equation 12-4 requires $\partial/\partial\phi$ to be zero. That is, there must be no gradients in the azimuthal direction. A common additional assumption is, however, that $u_\phi = 0$. In such cases, the ϕ-equation can be removed from Equation 12-4. The resulting system of equations is both easier to converge and computationally less expensive compared to retaining the ϕ-equation. The default 2D axisymmetric formulation of Equation 12-3 and Equation 12-4 therefore assumes that

$$\partial/\partial\phi = 0$$
$$u_\phi = 0$$

Compressible Flow

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

$$\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\mathbf{u}) = 0 \tag{12-8}$$

and the momentum equation:

$$\rho\frac{\partial\mathbf{u}}{\partial t} + \rho\mathbf{u}\cdot\nabla\mathbf{u} = -\nabla p + \nabla\cdot\left(\mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla\cdot\mathbf{u})\mathbf{I}\right) + \mathbf{F} \tag{12-9}$$

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

Weakly Compressible Flow

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

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

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

The Mach Number Limit

An important dimensionless number in fluid dynamics is the Mach number, Ma, defined by

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

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

The momentum equation, Equation 12-9, is parabolic for unsteady flow and elliptic for steady flow, whereas the continuity equation, Equation 12-8, is hyperbolic for both steady and unsteady flow. The combined system of equations is thus hybrid parabolic-hyperbolic for unsteady flow and hybrid elliptic-hyperbolic for steady flow. An exception occurs when the viscous term in Equation 12-9 becomes vanishingly small, such as at an outflow boundary, in which case the momentum equation becomes locally hyperbolic. The number of boundary conditions to apply on the boundary then depends on the number of characteristics propagating into the computational domain. For the purely hyperbolic system, the number of characteristics propagating from the boundary into the domain changes as the Mach number passes through unity. Hence, the number of boundary conditions required to obtain a numerically well-posed system must also change. The compressible formulation of the laminar and turbulent interfaces uses the same boundary conditions as the incompressible formulation, which implies that the compressible interfaces are not suitable for flows with a Mach number larger than or equal to one. Yet, the practical Mach number limit is lower than one. The main reason is that the numerical scheme (stabilization and boundary conditions) of the Laminar Flow interface does not recognize the direction and speed of pressure waves. The fully compressible Navier–Stokes equations do, for example, start to display very sharp gradients already at moderate Mach numbers. But the stabilization for the single-phase flow interface does not necessarily capture these gradients. It is impossible to give an exact limit where the low Mach number regime ends and the moderate Mach number regime begins, but a rule of thumb is that the Mach number effects start to appear at $\mathrm{Ma} = 0.3$. For this reason, the compressible formulation is referred to as *Compressible flow (Ma<0.3)* in COMSOL Multiphysics.

Incompressible Flow

When the temperature variations in the flow are small, a single-phase fluid can often be assumed incompressible; that is, ρ is constant or nearly constant. This is the case for all liquids under normal conditions and also for gases at low velocities. For constant ρ, Equation 12-8 reduces to

$$\rho\nabla\cdot\mathbf{u} = 0 \qquad (12\text{-}10)$$

and Equation 12-9 becomes

$$\rho\frac{\partial\mathbf{u}}{\partial t} + \rho(\mathbf{u}\cdot\nabla)\mathbf{u} = \nabla\cdot[-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)] + \mathbf{F} \qquad (12\text{-}11)$$

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

The Reynolds Number

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

$$\mathrm{Re} = \frac{\rho UL}{\mu}$$

where U denotes a velocity scale, and L denotes a representative length. The Reynolds number represents the ratio between inertial and viscous forces. At low Reynolds numbers, viscous forces dominate and tend to damp out all disturbances, which leads to laminar flow. At high Reynolds numbers, the damping in the system is very low, giving small disturbances the possibility to grow by nonlinear interactions. If the Reynolds number is high enough, the flow field eventually ends up in a chaotic state called turbulence.

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

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

Gravity

DEFINITION

The gravity force is defined from the acceleration of gravity vector, **g**, and the density, ρ. Under usual conditions and in Cartesian coordinates with the z-axis in the vertical direction,

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

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

$$\rho\frac{\partial\mathbf{u}}{\partial t} + \rho\mathbf{u}\cdot\nabla\mathbf{u} = \nabla\cdot\left(-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla\cdot\mathbf{u})\mathbf{I}\right) + \mathbf{F} + \rho\mathbf{g} \qquad (12\text{-}12)$$

Introducing a constant reference density ρ_{ref}, and assuming that **g** is homogeneous, this equation is equivalently written:

$$\begin{aligned}&\rho\frac{\partial\mathbf{u}}{\partial t} + \rho\mathbf{u}\cdot\nabla\mathbf{u} = \\ &\nabla\cdot\left(-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla\cdot\mathbf{u})\mathbf{I} + \rho_{\mathrm{ref}}\mathbf{g}\cdot(\mathbf{r}-\mathbf{r}_{\mathrm{ref}})\right) + \mathbf{F} + (\rho - \rho_{\mathrm{ref}})\mathbf{g}\end{aligned} \qquad (12\text{-}13)$$

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

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

$$\begin{aligned}&\rho\frac{\partial\mathbf{u}}{\partial t} + \rho\mathbf{u}\cdot\nabla\mathbf{u} = \nabla\cdot\left(-\tilde{p}\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla\cdot\mathbf{u})\mathbf{I}\right) + \\ &\mathbf{F} + (\rho - \rho_{\mathrm{ref}})\mathbf{g}\end{aligned} \qquad (12\text{-}14)$$

In Equation 12-14, the gravity force is written $(\rho - \rho_{\text{ref}})\mathbf{g}$.

PRESSURE FORMULATION

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

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

PRESSURE BOUNDARY CONDITION

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

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

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

Theory for the Wall Boundary Condition

See Wall for the node settings.

SLIP

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

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

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

$$\mathbf{u} \cdot \mathbf{n} = 0, \qquad \mathbf{K}_\mathrm{n} - (\mathbf{K}_\mathrm{n} \cdot \mathbf{n})\mathbf{n} = \mathbf{0}$$

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

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

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

SLIDING WALL

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

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

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

  where $\mathbf{t} = (n_y, -n_x)$ for 2D and $\mathbf{t} = (n_z, -n_r)$ for axial symmetry.
- For 3D components, the velocity is set equal to a given vector $\mathbf{u}_\mathrm{w}$ projected onto the boundary plane:

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

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

SLIP VELOCITY

In the microscale range, the flow at a boundary is seldom strictly no slip or slip. Instead, the boundary condition is something in between, and there is a slip velocity at the boundary. Two phenomena account for this velocity: violation of the continuum hypothesis for the viscosity and flow induced by a thermal gradient along the boundary.

The following equation relates the viscosity-induced jump in tangential velocity to the tangential shear stress along the boundary:

$$\Delta u = \frac{1}{\beta}\tau_{n,t}$$

For gaseous fluids, the coefficient β is given by

$$\beta = \frac{\mu}{\left(\frac{2-\alpha_{\mathrm{v}}}{\alpha_{\mathrm{v}}}\right)\lambda}$$

where μ is the fluid’s dynamic viscosity (SI unit: Pa·s), α_{v} represents the tangential momentum accommodation coefficient (TMAC) (dimensionless), and λ is the molecules’ mean free path (SI unit: m). The tangential accommodation coefficients are typically in the range of 0.85 to 1.0 and can be found in Ref. 17.

A simpler expression for β is

$$\beta = \frac{\mu}{L_s}$$

where L_s, the slip length (SI unit: m), is a straight channel measure of the distance from the boundary to the virtual point outside the flow domain where the flow profile extrapolates to zero. This equation holds for both liquids and gases.

Thermal creep results from a temperature gradient along the boundary. The following equation relates the thermally-induced jump in tangential velocity to the tangential gradient of the natural logarithm of the temperature along the boundary:

$$\Delta u = \sigma_{\mathrm{T}}\frac{\mu}{\rho}\nabla_t \log T$$

where σ_{T} is the thermal slip coefficient (dimensionless) and ρ is the density of the fluid. The thermal slip coefficients range between 0.3 and 1.0 and can be found in Ref. 17.

Combining the previous relationships results in the following equation:

$$\mathbf{u} - \mathbf{u}_{\mathrm{w},t} = \frac{L_s}{\mu}\tau_{n,t} + \sigma_{\mathrm{T}}\frac{\mu}{\rho T}\nabla_t T$$

Relate the tangential shear stress to the viscous boundary force by

$$\tau_{n,t} = \mathbf{K}_{\mathrm{n}} - (\mathbf{n}\cdot\mathbf{K}_{\mathrm{n}})\mathbf{n}$$

where the components of $\mathbf{K}$ are the Lagrange multipliers that are used to implement the boundary condition. Similarly, the tangential temperature gradient results from the difference of the gradient and its normal projection:

$$\nabla_t T = \nabla T - (\mathbf{n} \cdot \nabla T)\mathbf{n}$$

Use Viscous Slip

When viscous slip is used, select Maxwell's model to calculate L_s using:

$$L_s = \left(\frac{2-\alpha_\mathrm{v}}{\alpha_\mathrm{v}}\right)\lambda$$

Also see Wall for the node settings.

CONSTRAINT SETTINGS

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

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

  The main advantage of pointwise constraints is that they enforce the constraint exactly pointwise. This means that they do not introduce any leakage of the velocity across the wall, unless specified. The main disadvantage of pointwise constraints is that they introduce locking effects when trying to impose a no-penetration condition for the velocity, $\mathbf{u} \cdot \mathbf{n} = 0$, on curved walls or walls with sharp corners.
- **Weak constraints** is an alternative method to prescribe the velocity. It consists on enforcing the boundary condition for the velocity via *Lagrange multipliers*. Their main advantage is that the Lagrange multiplier can provide an accurate representation of the reaction flux at the wall. Their main disadvantage is that they introduce extra unknowns, and are usually difficult to combine with other constraint methods on adjacent boundaries. Moreover, they may require extra constraints for the Lagrange multipliers. For more information, see Weak Constraints in the *COMSOL Multiphysics Reference Manual*.
- **Nitsche constraints** use a numerical flux to prescribe the velocity at the wall. They impose the constraint in an integral sense rather than pointwise, and do not suffer

from the locking effects introduced by pointwise constraints when trying to prescribe a no penetration condition for the velocity. They are also better behaved when prescribing nonlinear constraints. Their main disadvantage is that the constraint is only imposed approximately, and may produce small leaks. For more information, see Nitsche Constraints.

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

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

  Nitsche constraints are used to impose the no penetration condition for Slip walls. When a No Slip condition is prescribed, pointwise constraints are used except for moving walls where Nitsche constraints are used.
- **Pointwise constraints**.
- **Nitsche constraints**.
- **Weak constraints**. They are not available on Interior Walls.
- **Mixed constraints**. This option is only available when both the tangential and normal components of the velocity need to be prescribed. The velocity on the wall normal direction is imposed via pointwise constraints. The constraint for the tangential directions is relaxed, and Nitsche constraints are used instead. This provides improved accuracy and performance when working with coarse boundary layer meshes. For more information, see Ref. 19.

Prescribing Inlet and Outlet Conditions

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

INLET CONDITIONS

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

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

$$-p + 2\mu\frac{\partial u_n}{\partial n} = F_n \qquad (12\text{-}15)$$

where $\partial u_n/\partial n$ is the normal derivative of the normal velocity component. Equation 12-15 is prescribed by the Pressure condition in the Inlet and Outlet features and the Normal stress condition in the Open Boundary and Boundary Stress features. Equation 12-15 is mathematically more stringent compared to specifying the pressure pointwise and at the same time cannot guarantee that p obtains the desired value. In practice, p is close to F_n, except for low Reynolds number flows where viscous effects are the only effects that balance the pressure. In addition to Equation 12-15, all but one velocity component must be specified. For low Reynolds numbers, this can be specified by a vanishing tangential stress condition:

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

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

OUTLET CONDITIONS

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

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

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

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

ALTERNATIVE FORMULATIONS

COMSOL provides several specialized boundary conditions that either provide detailed control over the flow at the boundary or that simulate specific devices. In practice they often prescribe a velocity or a pressure but calculate the prescribed values using, for example, ODEs.

Normal Stress Boundary Condition

Mass Flow

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

POINTWISE MASS FLUX

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

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

where m_f is the normal mass flux and ρ is the density.

MASS FLOW RATE

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

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

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

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

$$\mathbf{u} \times \mathbf{n} = \mathbf{0} \tag{12-16}$$

STANDARD FLOW RATE

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

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

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

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

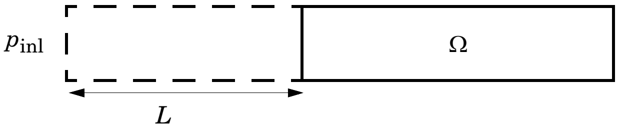
$$\rho_{\mathrm{st}} = \frac{p_{\mathrm{st}} M_n}{R T_{\mathrm{st}}}$$

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

Equation 12-16 or Equation 12-17 is also enforced for compressible and incompressible flow, respectively, ensuring that the normal component of the viscous stress and the tangential component of the velocity are zero at the boundary.

Fully Developed Flow (Inlet)

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



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

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

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

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

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

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

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

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

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

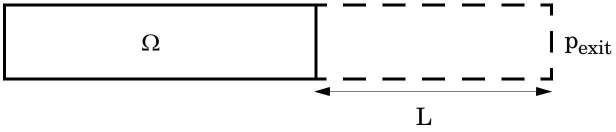
where <·> denotes the average over the inlet. Since the equation for P_{inl} is a DAE (the equation for P_{inl} does not contain P_{inl}), it must be solved coupled to Navier–Stokes and it must be treated by a Vanka presmoother and postsmoother if iterative solvers are used.

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

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

Fully Developed Flow (Outlet)

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



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

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

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

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

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

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

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

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

where <·> denotes the average over the inlet. Since the equation for P_{ext} is a DAE, it must be solved coupled to Navier–Stokes and it must be treated by a Vanka presmoother and postsmoother if iterative solvers are used.

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

No Viscous Stress

For this module, and in addition to the Pressure, No Viscous Stress boundary condition, the viscous stress condition sets the viscous stress to zero:

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

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

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

The condition is not a sufficient outlet condition since it lacks information about the outlet pressure. It must hence be combined with pressure point constraints on one or several points or lines surrounding the outlet.

This boundary condition is numerically the least stable outlet condition, but can still be beneficial if the outlet pressure is nonconstant due to, for example, a nonlinear volume force.

Normal Stress Boundary Condition

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

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

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

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

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

$$p = 2\mu\frac{\partial u_n}{\partial n}+f_0 \qquad (12\text{-}17)$$

If $\partial u_n/\partial n$ is small, Equation 12-17 states that $p \approx f_0$.

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

Pressure Boundary Condition

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

$$\left(-p\mathbf{I}+\mu(\nabla\mathbf{u}+(\nabla\mathbf{u})^T)-\frac{2}{3}\mu(\nabla\cdot\mathbf{u})\mathbf{I}\right)\mathbf{n} = -p_0\mathbf{n} \qquad (12\text{-}18)$$

$$(-p\mathbf{I}+\mu(\nabla\mathbf{u}+(\nabla\mathbf{u})^T))\mathbf{n} = -p_0\mathbf{n} \qquad (12\text{-}19)$$

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

This is a normal stress condition together with a no-tangential-stress condition. When $\mu > 0$, Equation 12-18 or Equation 12-19 can be supplemented with a tangential velocity condition

$$\mathbf{u}\cdot\mathbf{t} = 0 \qquad (12\text{-}20)$$

If so, the no-tangential-stress condition is overridden. An issue with Equation 12-18 or Equation 12-19 is that it does not strongly enforce unidirectional flow on the boundary. If the prescribed pressure on an outlet is too high, parts of the outlet can actually have inflow. This is not as much of an issue for the Navier–Stokes equations as it is an issue for scalar transport equations solved along with the Navier–Stokes equations. Hence, when applying the **Pressure** boundary condition at an outlet or inlet you can further constrain the flow. With the **Suppress backflow** option

$$\begin{gathered}\left(-p\mathbf{I}+\mu(\nabla\mathbf{u}+(\nabla\mathbf{u})^T)-\frac{2}{3}\mu(\nabla\cdot\mathbf{u})\mathbf{I}\right)\mathbf{n}=-\hat{p}_0\mathbf{n}\\ (-p\mathbf{I}+\mu(\nabla\mathbf{u}+(\nabla\mathbf{u})^T))\mathbf{n}=-\hat{p}_0\mathbf{n}\\ \hat{p}_0\le p_0\end{gathered}\quad, \tag{12-21}$$

the normal stress is adjusted to keep

$$\mathbf{u}\cdot\mathbf{n}\ge 0 \tag{12-22}$$

Equation 12-21 effectively means that the prescribed pressure is p_0 if $\mathbf{u}\cdot\mathbf{n}\ge 0$, but smaller at locations where $\mathbf{u}\cdot\mathbf{n}<0$. This means that Equation 12-21 does not completely prevent backflow, but the backflow is substantially reduced. Backflow is suppressed also when external forces are acting on the fluid, provided the magnitude of these forces are of the same order as the dynamic pressure at the outlet.

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

$$\begin{gathered}\mathbf{n}^T\left(-p\mathbf{I}+\mu(\nabla\mathbf{u}+(\nabla\mathbf{u})^T)-\frac{2}{3}\mu(\nabla\cdot\mathbf{u})\mathbf{I}\right)\mathbf{n}=-\hat{p}_0\\ \mathbf{n}^T(-p\mathbf{I}+\mu(\nabla\mathbf{u}+(\nabla\mathbf{u})^T))\mathbf{n}=-\hat{p}_0\\ \hat{p}_0\ge p_0\end{gathered} \tag{12-23}$$

together with the tangential condition in Equation 12-20, or, a general flow direction is prescribed.

$$\begin{gathered}\mathbf{r}_{\mathbf{u}}^T\left(-p\mathbf{I}+\mu(\nabla\mathbf{u}+(\nabla\mathbf{u})^T)-\frac{2}{3}\mu(\nabla\cdot\mathbf{u})\mathbf{I}\right)\mathbf{n}=-\hat{p}_0(\mathbf{r}_{\mathbf{u}}\cdot\mathbf{n})\\ \mathbf{r}_{\mathbf{u}}^T(-p\mathbf{I}+\mu(\nabla\mathbf{u}+(\nabla\mathbf{u})^T))\mathbf{n}=-\hat{p}_0(\mathbf{r}_{\mathbf{u}}\cdot\mathbf{n})\\ \hat{p}_0\ge p_0\\ \mathbf{u}-(\mathbf{u}\cdot\mathbf{r}_{\mathbf{u}})\mathbf{r}_{\mathbf{u}}=\mathbf{0},\ \mathbf{r}_{\mathbf{u}}=\frac{\mathbf{d}_{\mathbf{u}}}{\|\mathbf{d}_{\mathbf{u}}\|}\end{gathered} \tag{12-24}$$

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

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

$$p = p_{\mathrm{stat}} = p_{\mathrm{tot}} - \frac{1}{2}\rho|\mathbf{u}|^2 \tag{12-25}$$

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

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

$$\mathbf{n}^T(-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T))\mathbf{n} = \mathbf{n}^T(-p_{\mathrm{stat}}\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T))\mathbf{n} \tag{12-26}$$

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

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

For the second option, Equation 12-25 is prescribed pointwise.

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

Vacuum Pump Boundary Condition

Vacuum pumps (devices) can be represented using lumped curves implemented as boundary conditions. These simplifications also imply some assumptions. In particular, it is assumed that a given boundary can only be either an inlet or an outlet. Such a boundary should not be a mix of inlets/outlets nor switch between them during a simulation.

Manufacturers usually provide curves that describe the static pressure as a function of flow rate for a vacuum pump. Also see Vacuum Pump for the node settings.

DEFINING A DEVICE AT AN OUTLET

In this case (see Figure 12-1), the device's inlet is the interior face situated between the blue (cube) and green (circle) domains while its outlet is an external boundary, here the circular boundary of the green domain. The lumped curve gives the flow rate as a function of the pressure difference between the interior face and the external boundary. This boundary condition implementation follows the Pressure Boundary Condition for outlets with the Suppress backflow option:

$$\left(-p\mathbf{I}+\mu(\nabla\mathbf{u}+(\nabla\mathbf{u})^{\mathrm{T}})-\frac{2}{3}\mu(\nabla\cdot\mathbf{u})\mathbf{I}\right)\mathbf{n}=-\hat{p}_0\mathbf{n} \qquad (12\text{-}27)$$
$$\hat{p}_0\le p_{\text{vacuum pump}}(V_0)-p_{\text{ref}}$$

Here, V_0 is the flow rate across the boundary and $p_{\text{vacuum pump}}(V_0)$ is the static pressure function of flow rate for the vacuum pump. p_{ref} is the reference pressure that ensure that the absolute pressure is set to $p_{\text{vacuum pump}}$. In vacuum pump models the reference pressure is usually set to 0 Pa since the absolute pressure is close to 0 Pa by opposition to ambient conditions where p_{ref} is often set to 1 atm. Equation 12-27 corresponds to the compressible formulation. For incompressible flows, the term $-(2/3)\mu(\nabla\cdot\mathbf{u})$ vanishes. In 2D the thickness in the third direction, Dz, is used to define the flow rate. Vacuum pumps are modeled as rectangles in this case.

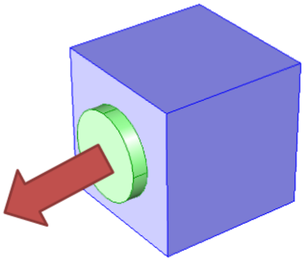


Figure 12-1: A vacuum pump at the outlet. The arrow represents the flow direction, the green circle represents the vacuum pump (that should not be part of the model), and the blue cube represents the modeled domain with an outlet boundary condition described by a lumped curve for the attached vacuum pump.

Porous Interface

This condition properly accounts for a sudden jump in porosity ε_p across boundary between different domains, giving the corresponding jump in stress across the interface. The emergence of a boundary layer on the upstream side of the interface is also revealed, which provides modifications to the behavior of the tangential velocity near the interface. The feature helps to avoid the need to add a very thin transition region, in which $\varepsilon_{p,\text{upstream}}$ changes to $\varepsilon_{p,\text{downstream}}$ continuously, while solving the

Brinkman equation in such situations. Below, ‘–’ and ‘+’ (used as subscripts and superscripts) refer to the upstream and downstream side of the interface, respectively.

Whatever the option for **Continuity conditions** is chosen, the velocity conditions are

$$[\rho(\mathbf{u}\cdot\mathbf{n})]\big|_{-}^{+} = 0, \qquad \mathbf{u}_{\mathrm{t}}\big|_{-}^{+} \equiv [\mathbf{u}-(\mathbf{u}\cdot\mathbf{n})\mathbf{n}]\Big|_{-}^{+} = 0$$

That is, the mass flux and tangential velocity across the interface are constant, and $\mathbf{n}$ is assumed to be a normal pointing from ‘–’ to ‘+’. **Velocity and stress** option just imposes usual stress continuity adding a correction for a possible jump in density:

$$\left[p\mathbf{n}-\mathbf{K}\mathbf{n}+\rho(\mathbf{u}\cdot\mathbf{n})\left(\left[\frac{1}{\varepsilon_{\mathrm{p}}}\right]_{\mathrm{av}}\right)^{2}\mathbf{u}\right]\Bigg|_{-}^{+} = 0$$

The option **Velocity and porosity-corrected stress** (default) adds a contribution due to the inertial term, which is non-trivial (gives non-zero effect) when ε_{p} jumps across the interface:

$$\left[p\mathbf{n}-\mathbf{K}\mathbf{n}+\rho(\mathbf{u}\cdot\mathbf{n})\left(\left[\frac{1}{\varepsilon_{\mathrm{p}}}\right]_{\mathrm{av}}\right)^{2}\mathbf{u}+\rho(\mathbf{u}\cdot\mathbf{n})\left(\frac{1}{2\varepsilon_{\mathrm{p}}^{2}}\right)\mathbf{u}_{\mathrm{av}}\right]\Bigg|_{-}^{+} = 0$$

Notice that $\left[\frac{1}{\varepsilon_{\mathrm{p}}}\right]_{\mathrm{av}} = \left(\left[\frac{1}{\varepsilon_{\mathrm{p}}}\right]_{-}+\left[\frac{1}{\varepsilon_{\mathrm{p}}}\right]_{+}\right)/2, \quad \mathbf{u}_{\mathrm{av}} = (\mathbf{u}_{-}+\mathbf{u}_{+})/2$ and so on.
Then, the **Velocity and porosity-corrected stress with loss** option has a ‘pressure loss’ term:

$$\left[p\mathbf{n}-\mathbf{K}\mathbf{n}+\rho(\mathbf{u}\cdot\mathbf{n})\left(\left[\frac{1}{\varepsilon_{\mathrm{p}}}\right]_{\mathrm{av}}\right)^{2}\mathbf{u}+\rho(\mathbf{u}\cdot\mathbf{n})\left(\frac{1}{2\varepsilon_{\mathrm{p}}^{2}}\right)\mathbf{u}_{\mathrm{av}}\right]\Bigg|_{-}^{+} + p^{(\mathrm{loss})}\mathbf{n} = 0$$

This loss occurs in the downstream direction. Its mechanism is similar to the mechanism of common engineering ‘total pressure loss’. Indeed, ‘expansion’ and ‘contraction’ across the interface experienced by the flow on the microscopic level (due to increasing or decreasing porosity, ε_{p}, respectively), are roughly equivalent to expansion and contraction of flow in a pipe with geometrical variations. Accordingly, it is assumed that only the normal velocity component is responsible for the loss. In the case of normal flow and negligible viscous term the pipe-analogy is quite exact.

Subsequently, **Pressure loss model** should be specified. Choosing **User-defined loss coefficients** requires values of **Expansion loss coefficient** and **Contraction loss coefficient** ($K_{exp} = 1$ and $K_{contr} = 0.5$ are default values). The resulting losses on the 'expansion' and 'contraction' boundaries are:

$$p^{(\text{loss,expansion})} = K_{\text{exp}}\left(\left(\frac{\rho(\mathbf{u}\cdot\mathbf{n})^2}{2\varepsilon_p^2}\right)_-\right),$$

$$p^{(\text{loss,contraction})} = K_{\text{contr}}\left(\frac{\rho(\mathbf{u}\cdot\mathbf{n})^2}{2\varepsilon_p^2}\right)_+$$

Borda–Carnot option results in the following losses:

$$p^{(\text{loss,expansion})} = \frac{\rho_{\text{av}}}{2}\left(\left[\frac{(\mathbf{u}\cdot\mathbf{n})}{\varepsilon_p}\right]_+ - \left[\frac{(\mathbf{u}\cdot\mathbf{n})}{\varepsilon_p}\right]_-\right)^2,$$

$$p^{(\text{loss,contraction})} = \frac{1}{2}\left(\frac{\rho(\mathbf{u}\cdot\mathbf{n})^2}{\varepsilon_p^2}\right)_+\left(\frac{1}{c_{\text{vc}}} - 1\right)^2$$

and requires to choose **Vena contracta model**. The **User defined** option asks for value of **Vena contracta coefficient** ($c_{\text{vc}} = 0.5$ by default). **Weisbach** option computes the coefficient based on the ratio of porosities across the interface:

$$c_{\text{vc}} = c_{\text{vc,ll}} + (1 - c_{\text{vc,ll}})\left(\frac{\varepsilon_{p,+}}{\varepsilon_{p,-}}\right)^{n_{\text{vc}}}$$

and requires inputs for **Lower limit** and **Exponent** ($c_{\text{vc,ll}} = 0.63$ and $n_{\text{vc}} = 3$ are default values).

Although the velocity conditions formally look the same for all **Continuity conditions** options, the latter two (containing the inertial correction due to the porosity jump) lead to an extra non-trivial effect. Namely, the tangential velocity exponentially evolves over a boundary layer formed on the upstream side of the interface boundary. This boundary layer has a characteristic length equal to

$$l_- = \frac{\mu\varepsilon_{p,-}}{\rho(\mathbf{u}\cdot\mathbf{n})}$$

The ratio of the tangential velocities right after the interface ('downstream') and outside the boundary layer ('upstream infinity') is

$$\frac{|\mathbf{u}_{t,+}|}{|\mathbf{u}_{t,\text{upstream infinity}}|} = \frac{2\varepsilon_{p,+}/\varepsilon_{p,-}}{\varepsilon_{p,+}/\varepsilon_{p,-} + \varepsilon_{p,-}/\varepsilon_{p,+}}$$

In terms of the interstitial velocity, $\mathbf{u}^{(i)} = \mathbf{u}/\varepsilon_p$, this results in:

$$\frac{|\mathbf{u}^{(i)}{}_{t,+}|}{|\mathbf{u}^{(i)}{}_{t,\text{upstream infinity}}|} = \frac{2}{\varepsilon_{p,+}/\varepsilon_{p,-} + \varepsilon_{p,-}/\varepsilon_{p,+}} \le 1$$

Obviously, tangential component of the interstitial kinetic energy is lost across the interface. For low viscosity and high normal-flow component, when l_u is much lower than l_{geom} (the characteristic geometry length) there are apparent variations in $\mathbf{u}_t$ and $\mathbf{u}^{(i)}{}_t$ in a thin boundary layer on the upstream side of the interface. However, for the cases with high viscosity or nearly interface-parallel flow, l_u is much larger than l_{geom}, and any variations in $\mathbf{u}_t$ and $\mathbf{u}^{(i)}{}_t$ are virtually invisible.

Notice that the feature does not take into account possible underresolved porous scale, as is described in Theory for the Free and Porous Media Flow, Darcy Interface.

Mass Sources for Fluid Flow

There are two types of mass sources in a Single-Phase Flow interface: point sources and line sources.

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

POINT SOURCE

A point source is theoretically formed by taking a mass injection/ejection, $\dot{Q}$ (SI unit: kg/(m^3·s)), in a small volume δV and then letting the size of the volume tend to zero while keeping the total mass flux constant. Given a point source strength, $\dot{q}_p$ (SI unit: kg/s), this can be expressed as

$$\lim_{\delta V \to 0} \int_{\delta V} \dot{Q} = \dot{q}_p \qquad (12\text{-}28)$$

An alternative way to form a point source/sink is to assume that mass is injected/extracted through the surface of a small object. Letting the object surface area tend to zero while keeping the mass flux constant, results in the same point source. For this alternative approach, effects resulting from the physical object volume, such as drag and fluid displacement, need to be neglected.

The weak contribution

$$\dot{q}_{\mathrm{p}}\mathrm{test}(p)$$

is added to a point in the geometry. As can be seen from Equation 12-28, $\dot{Q}$ must tend to plus or minus infinity as δV tends to zero. This means that in theory the pressure also tends to plus or minus infinity.

Observe that "point" refers to the physical representation of the source. A point source can therefore only be added to points in 3D components and to points on the symmetry axis in 2D axisymmetry components. Other geometrical points in 2D components represent physical lines.

The finite element representation of Equation 12-28 corresponds to a finite pressure in a point with the effect of the point source spread out over a region around the point. The size of the region depends on the mesh and on the strength of the source. A finer mesh gives a smaller affected region, but also a more extreme pressure value. It is important not to mesh too finely around a point source since the resulting pressure can result in unphysical values for the density, for example. It can also have a negative effect on the condition number for the equation system.

LINE SOURCE

A line source can theoretically be formed by assuming a source of strength $\dot{Q}$ (SI unit: kg/(m^3·s)), located within a tube with cross-sectional area δS and then letting δS tend to zero, while keeping the total mass flux per unit length constant. Given a line source strength, $\dot{q}_1$ (SI unit: kg/(m·s)), this can be expressed as

$$\lim_{\delta S \to 0} \int_{\delta S} \dot{Q} = \dot{q}_1 \tag{12-29}$$

As in the point source case, an alternative approach is to assume that mass is injected/extracted through the surface of a small object. This results in the same mass source, but requires that effects on the fluid resulting from the physical object volume are neglected.

The weak contribution

$$\dot{q}_1\text{test}(p)$$

is added to lines in 3D or to points in 2D (which represent cut-through views of lines). Line sources can also be added to the axisymmetry line in 2D axisymmetry components. It cannot, however, be added to geometrical lines in 2D because they represent physical planes.

As with a point source, it is important not to mesh too finely around the line source.

For feature node information, see Line Mass Source and Point Mass Source in the *COMSOL Multiphysics Reference Manual*.

For the Reacting Flow in Porous Media, Diluted Species interface, which is available with the CFD Module, Chemical Reaction Engineering Module, or Battery Design Module, these shared physics nodes are renamed as follows:

- The **Line Mass Source** node is available as two nodes, one for the fluid flow (**Fluid Line Source**) and one for the species (**Species Line Source**).
- The **Point Mass Source** node is available as two nodes, one for the fluid flow (**Fluid Point Source**) and one for the species (**Species Point Source**).

Numerical Stability — Stabilization Techniques for Fluid Flow

The momentum equation (Equation 12-9 or Equation 12-11) is a (nonlinear) convection-diffusion equation. Such equations can easily become unstable if discretized using the Galerkin finite element method. Stabilized finite element methods are usually necessary in order to obtain physical solutions. The stabilization settings are found in the main Fluid Flow interface features. To display this section, click the **Show More Options** button (👁) and select **Stabilization** in the **Show More Options** dialog box.

There are three types of stabilization methods available for Navier–Stokes — *streamline diffusion*, *crosswind diffusion*, and *isotropic diffusion*. Streamline diffusion and crosswind diffusion are consistent stabilization methods, whereas isotropic diffusion is an inconsistent stabilization method.

For optimal functionality, the exact weak formulations of and constants in the streamline diffusion and crosswind diffusion methods depend on the order of the shape functions (basis functions) for the elements. The values of constants in the streamline diffusion and crosswind diffusion methods follow Ref. 5 and Ref. 6.

STREAMLINE DIFFUSION

For strongly coupled systems of equations, the streamline diffusion method must be applied to the system as a whole rather than to each equation separately. These ideas were first explored by Hughes and Mallet (Ref. 8) and were later extended to Galerkin least-squares (GLS) applied to the Navier–Stokes equations (Ref. 9). This is the streamline diffusion formulation that COMSOL Multiphysics supports.

Streamline diffusion is active by default because it is necessary when convection is dominating the flow.

The governing equations for incompressible flow are subject to the Babuška–Brezzi condition, which states that the shape functions (basis functions) for pressure must be of lower order than the shape functions for velocity. If the incompressible Navier–Stokes equations are stabilized by streamline diffusion, it is possible to use equal-order interpolation. Hence, streamline diffusion is necessary when using first-order elements for both velocity and pressure. This applies also if the model is solved using geometric multigrid (either as a solver or as a preconditioner) and at least one multigrid hierarchy level uses linear Lagrange elements.

CROSSWIND DIFFUSION

Crosswind diffusion can also be formulated for systems of equations, and when applied to the Navier–Stokes equations it becomes a shock-capturing operator. COMSOL Multiphysics supports the formulation in Ref. 9 with a shock-capturing viscosity of the Hughes–Mallet type Ref. 8.

Incompressible flows do not contain shock waves, but crosswind diffusion is still useful for introducing extra diffusion in sharp boundary layers and shear layers that otherwise would require a very fine mesh to resolve.

Crosswind diffusion is active by default as it makes it easier to obtain a solution even if the problem is fully resolved by the mesh. Crosswind diffusion also enables the iterative solvers to use inexpensive presmoothers. If crosswind diffusion is deactivated, more expensive preconditioners must be used instead.

STABILIZATION TIME SCALES

The time-scale tensor for time-dependent problems should in theory depend on a time-scale that many references just set to the time-step taken by the time-solver, Δt (see, for example, Ref. 9 and Ref. 10). This simple approach does not, however, necessarily reflect on the actual time-scales in the physics. A typical example is reacting flows where the time step is often guided by fast reactions, while the flow develops relatively slowly. The COMSOL Multiphysics software can therefore replace Δt^2 in the time-scale tensor with measures of type $(1/(\Delta\tilde{t})^2)^{-1}$, which are calculated from projections of weak expressions in a fashion similar to those in Ref. 11. These measures of the time scale are used when **Use dynamic subgrid time scale** is activated.

The **Limit small time steps effect on stabilization time scale** option can be used to ramp down the time-step contribution in the stabilization when the time step becomes much smaller than the convective and diffusive time scales. This option prevents loss of pressure stabilization at such small time steps.

ISOTROPIC DIFFUSION

Isotropic diffusion adds diffusion to the Navier–Stokes equations. Isotropic diffusion significantly reduces the accuracy of the solution but does a very good job at reducing oscillations. The stability of the continuity equation is not improved.

Numerical Stabilization and Iterative in the *COMSOL Multiphysics Reference Manual*

Solvers for Laminar Flow

The Navier–Stokes equations constitute a nonlinear equation system. A nonlinear solver must hence be applied to solve the problem. The nonlinear solver iterates to reach the final solution. In each iteration, a linearized version of the nonlinear system is solved using a linear solver. In the time-dependent case, a time marching method must also be applied. The default suggestions for each of these solver elements are discussed below.

NONLINEAR SOLVER

The nonlinear solver method depends on if the model solves a stationary or a time-dependent problem.

Stationary Solver

In the stationary case, a fully coupled, damped Newton method is applied. The initial damping factor is low since a full Newton step can be harmful unless the initial values are close to the final solution. The nonlinear solver algorithm automatically regulates the damping factor in order to reach a converged solution.

For advanced models, the automatically damped Newton method might not be robust enough. A pseudo time-stepping algorithm can then be invoked. See Pseudo Time Stepping for Laminar Flow Models.

Time-Dependent Solver

In the time-dependent case, the initial guess for each time step is (loosely speaking) the previous time step, which is a very good initial value for the nonlinear solver. The automatic damping algorithm is then not necessary. The damping factor in the Newton method is instead set to a constant value slightly smaller than one. Also, for the same reason, it suffices to update the Jacobian once per time step.

It is seldom worth the extra computational cost to update the Jacobian more than once per time step. For most models it is more efficient to restrict the maximum time step or possibly lower the damping factor in the Newton method.

LINEAR SOLVER

The linearized Navier–Stokes equation system has saddle point character, unless the density depends on the pressure. This means that the Jacobian matrix has zeros on the diagonal. Even when the density depends on the pressure, the equation system effectively shares many numerical properties with a saddle point system.

For small 2D and 3D models, the default solver suggestion is a direct solver. Direct solvers can handle most nonsingular systems and are very robust and also very fast for small models. Unfortunately, they become slow for large models and their memory requirement scales as somewhere between $N^{1.5}$ and N^2, where N is the number of degrees of freedom in the model. The default suggestion for large 2D and 3D models is therefore the iterative GMRES solver. The memory requirement for an iterative solver optimally scales as N.

GMRES is accelerated by a multigrid method, per default the smoothed aggregation algebraic multigrid (SAAMG) method. The cost of SAAMG is typically very low compared to the number of GMRES iterations necessary if no multigrid method is used. As the name implies, SAAMG builds its coarser meshes algebraically, so the application requires no additional meshes in order to employ SAAMG. In contrast, the geometric multigrid (GMG) method requires actual meshes. If a sufficient number of multigrid levels can be constructed, GMG is often faster than SAAMG. GMG is also superior for cluster computations and for shared-memory computations with many cores. When the default linear solver is GMRES, an optional, but deactivated, linear solver node is available where GMRES is accelerated by GMG.

Multigrid methods need smoothers, but the saddle point character of the linear system restricts the number of applicable smoothers. The choices are further restricted by the anisotropic meshes frequently encountered in fluid-flow problems. The efficiency of the smoothers is highly dependent on the numerical stabilization. Iterative solvers perform at their best when both Streamline Diffusion and Crosswind Diffusion are active.

The default smoother for P1+P1 elements is SCGS. This is an efficient and robust smoother specially designed to solve saddle point systems on meshes that contain anisotropic elements. The SCGS smoother works well even without crosswind diffusion. SCGS can sometimes work for higher-order elements, especially if **Method** in the SCGS settings is set to **Mesh element lines**. But there is no guarantee for this, so the default smoother for higher order elements is an SOR Line smoother. SOR Line handles mesh anisotropy but does not formally address the saddle point character. It does, however, function in practice provided that streamline diffusion and crosswind diffusion are both active.

A different kind of saddle point character can arise if the equation system contains ODE variables. Some advanced boundary conditions can add equations with such variables. These variables must be treated with the Vanka algorithm. SCGS includes an option to invoke Vanka. Models with higher-order elements must apply SCGS or use the Vanka smoother. The latter is the default suggestion for higher-order elements, but it does not work optimally for anisotropic meshes.

TIME-DEPENDENT SOLVERS

The default time-dependent solver for Navier–Stokes is the BDF method with maximum order set to two. Higher BDF orders are not stable for transport problems in general nor for Navier–Stokes in particular.

BDF methods have been used for a long time and are known for their stability. However, they can have severe damping effects, especially the lower-order methods. Hence, if robustness is not an issue, a model can benefit from using the generalized-α method instead. Generalized-α is a solver that has properties similar to those of the second-order BDF solver but it is much less diffusive.

Both BDF and generalized-α are per default set to automatically adjust the time step. While this works well for many models, extra efficiency and accuracy can often be gained by specifying a maximum time step. It is also often beneficial to specify an initial time step to make the solver progress smoothly in the beginning of the time series.

In the *COMSOL Multiphysics Reference Manual:*

- Time-Dependent Solver
- Multigrid, Direct, Iterative, SCGS, SOR Line, and Vanka
- Stationary Solver

Pseudo Time Stepping for Laminar Flow Models

A stationary formulation has per definition no time derivatives and Equation 12-11 reduces to:

$$\rho(\mathbf{u}\cdot\nabla)\mathbf{u} = \nabla\cdot[-p\mathbf{I}+\mu(\nabla\mathbf{u}+(\nabla\mathbf{u})^T)]+\mathbf{F} \tag{12-30}$$

Solving Equation 12-30 requires a starting guess that is close enough to the final solution. If no such guess is at hand, the fully transient problem can be solved instead. This is, however, a rather costly approach in terms of computational time. An intermediate approach is to add a fictitious time derivative to Equation 12-30:

$$\rho\frac{\mathbf{u}-\mathrm{nojac}(\mathbf{u})}{\Delta\tilde{t}}+\rho(\mathbf{u}\cdot\nabla)\mathbf{u} = \nabla\cdot[-p\mathbf{I}+\mu(\nabla\mathbf{u}+(\nabla\mathbf{u})^T)]+\mathbf{F}$$

where $\Delta\tilde{t}$ is a *pseudo time step*. Since $\mathbf{u}$–`nojac`$(\mathbf{u})$ is always zero, this term does not affect the final solution. It does, however, affect the discrete equation system and effectively transforms a nonlinear iteration into a step of size $\Delta\tilde{t}$ of a time-dependent solver.

Pseudo time stepping is not active per default. The pseudo time step $\Delta\tilde{t}$ can be chosen individually for each element based on the local CFL number:

$$\Delta\tilde{t} = \mathrm{CFL}_{\mathrm{loc}}\frac{h}{|\mathbf{u}|}$$

where h is the mesh cell size. A small CFL number means a small time step. It is practical to start with a small CFL number and gradually increase it as the solution approaches steady state.

If the automatic expression for $\mathrm{CFL}_{\mathrm{loc}}$ is set to the built-in variable `CFLCMP`, then the automatic setting suggests a PID regulator for the pseudo time step in the default solver. The PID regulator starts with a small CFL number and increases $\mathrm{CFL}_{\mathrm{loc}}$ as the solution comes closer to convergence.

The default manual expression is

$$\begin{gathered}1.3^{\min(\mathrm{niterCMP},\,9)}+\\ \mathrm{if}(\mathrm{niterCMP}>20,\,9\cdot 1.3^{\min(\mathrm{niterCMP}-20,\,9)},\,0)+\\ \mathrm{if}(\mathrm{niterCMP}>40,\,90\cdot 1.3^{\min(\mathrm{niterCMP}-40,\,9)},\,0)\end{gathered} \tag{12-31}$$

The variable `niterCMP` is the nonlinear iteration number. It is equal to one for the first nonlinear iteration. $\mathrm{CFL}_{\mathrm{loc}}$ starts at 1.3 and increases by 30% each iteration until it reaches $1.3^9 \approx 10.6$. It remains there until iteration number 20 at which it starts to increase until it reaches approximately 106. A final increase after iteration number 40 then takes it to 1060. Equation 12-31 can, for some advanced flows, increase $\mathrm{CFL}_{\mathrm{loc}}$ too slowly or too quickly. $\mathrm{CFL}_{\mathrm{loc}}$ can then be tuned for the specific application.

For details about the CFL regulator, see Pseudo Time Stepping in the *COMSOL Multiphysics Reference Manual*.

Nitsche Constraints

Some boundary conditions are implemented using Nitsche constraints. These boundary conditions include

- Wall — Slip, sliding walls, and moving walls (that is, walls with nonzero translational velocity).
- Periodic Flow Condition
- Flow Continuity

The formulation used in the Fluid Flow interfaces in single-phase flow interfaces corresponds to the Symmetric Interior Penalty Galerkin method (SIPG). The SIPG method can be regarded to satisfy the boundary conditions in an integral sense rather than pointwise. More information on SIPG can be found in Ref. 15.

In particular, the SIPG formulation includes a penalty parameter that must be large enough for the formulation to be coercive. The higher the value, the better the boundary condition is fulfilled, but a too high value results in an ill-conditioned equation system. The penalty parameter is implemented according to Ref. 16.

Particle Tracing in Fluid Flow

The Particle Tracing Module is available to assist with these types of modeling problems.

It is possible to model particle tracing with COMSOL Multiphysics provided that the impact of the particles on the flow field is negligible. First compute the flow field, and then, as an analysis step, calculate the motion of the particles. The motion of a particle is defined by Newton's second law

$$m\frac{d^2\mathbf{x}}{dt^2} = \mathbf{F}\left(t, \mathbf{x}, \frac{d\mathbf{x}}{dt}\right)$$

where $\mathbf{x}$ is the position of the particle, m the particle mass, and $\mathbf{F}$ is the sum of all forces acting on the particle. Examples of forces acting on a particle in a fluid are the drag force, the buoyancy force, and the gravity force. The drag force represents the force that a fluid exerts on a particle due to a difference in velocity between the fluid and the particle. It includes the viscous drag, the added mass, and the Basset history term. Several empirical expressions have been suggested for the drag force. One of those is the one proposed by Khan and Richardson (Ref. 13). That expression is valid for spherical particles for a wide range of particle Reynolds numbers. The particle Reynolds number is defined as

$$\mathrm{Re}_p = \frac{|\mathbf{u} - \mathbf{u}_p| 2r\rho}{\mu}$$

where $\mathbf{u}$ is the velocity of the fluid, $\mathbf{u}_p$ the particle velocity, r the particle radius, ρ the fluid density, and μ the dynamic viscosity of the fluid. The empirical expression for the drag force according to Khan and Richardson is

$$\mathbf{F} = \pi r^2 \rho|\mathbf{u} - \mathbf{u}_p|(\mathbf{u} - \mathbf{u}_p)[1.84\mathrm{Re}_p^{-0.31} + 0.293\mathrm{Re}_p^{0.06}]^{3.45}$$

The model *Flow Past a Cylinder* (Application Library path **COMSOL_Multiphysics/Fluid_Dynamics/cylinder_flow**) demonstrates how to add and set up particle tracing in a plot group using the **Particle Tracing with Mass** node. It uses the predefined Khan–Richardson model for the drag force and neglects gravity and buoyancy forces.

References for the Single-Phase Flow, Laminar Flow Interfaces

1. G.G. Stokes, *Trans. Camb. Phil. Soc.*, vol. 8, pp. 287–305, 1845.

2. P.M. Gresho and R.L. Sani, *Incompressible Flow and the Finite Element Method, Volume 2: Isothermal Laminar Flow*, John Wiley & Sons, 2000.

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

4. R.L. Panton, *Incompressible Flow*, 2nd ed., John Wiley & Sons, 1996.

5. I. Harari and T.J.R. Hughes, "What are C and h? Inequalities for the Analysis and Design of Finite Element Methods," *Comput. Methods Appl. Mech. Eng.*, vol. 97, pp. 157–192, 1992.

6. Y. Bazilevs, V.M. Calo, T.E. Tezduyar, and T.J.R. Hughes, "$YZ\beta$ Discontinuity Capturing for Advection-dominated Processes with Application to Arterial Drug Delivery," *Int. J. Num. Meth. Fluids*, vol. 54, pp. 593–608, 2007.

7. R.B. Bird, W.E. Stewart, and E.N. Lightfoot, *Transport Phenomena*, 2nd ed., John Wiley & Sons, 2007.

8. T.J.R. Hughes and M. Mallet, "A New Finite Element Formulation for Computational Fluid Dynamics: III. The Generalized Streamline Operator for Multidimensional Advective-Diffusive System," *Comput. Methods Appl. Mech. Eng.*, vol. 58, pp. 305–328, 1986.

9. G. Hauke and T.J.R. Hughes, "A Unified Approach to Compressible and Incompressible Flows," *Comput. Methods Appl. Mech. Eng.*, vol. 113, pp. 389–395, 1994.

10. G. Hauke, "Simple Stabilizing Matrices for the Computation of Compressible Flows in Primitive Variables", *Comput. Methods Appl. Mech. Eng.*, vol. 190, pp. 6881–6893, 2001.

11. M.-C. Hsu, Y. Bazilevs, V.M. Cali, T.E. Tezduyar, and T.J.R. Hughes, "Improving stability of stabilized and multiscale formulations in flow simulations at small time steps," *Comput. Methods Appl. Mech. Eng.*, vol. 199, pp. 828–840, 2010.

12. D.J. Tritton, *Physical Fluid Dynamics*, 2nd ed., Oxford University Press, 1988.

13. J.M. Coulson and J.F. Richardson, "Particle Technology and Separation Processes," *Chemical Engineering, Volume 2*, Butterworth-Heinemann, 2002.

14. J.L. Guermond, P. Minev, and J. Shen, “An overview of projection methods for incompressible flows,” *Comput. Methods Appl. Mech. Eng.*, vol. 195, pp. 6011–6045, 2006.

15. B. Rivière, *Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations*, SIAM, 2008.

16. Y. Epshteyn and B. Rivière, “Estimation of penalty parameters for symmetric interior penalty Galerkin methods,” *J. Comput. Appl. Math.*, vol. 206, pp. 843–872, 2007.

17. G. Kariadakis, A. Beskok, and N. Aluru, *Microflows and Nanoflows*, Springer Science and Business Media, 2005.

18. R.P. Chhabra and J.F. Richardson, *Non-Newtonian Flow and Applied Rheology*, 2nd ed., Elsevier, 2008.

19. Y. Bazilevs and T.J.R. Hughes, “Weak imposition of Dirichlet boundary conditions in fluid mechanics,” *Comput. Fluids*, vol. 36, pp. 12–26, 2007.

20. G. Kariadakis, A. Beskok, and N. Aluru, *Microflows and Nanoflows*, Springer Science and Business Media, 2005.

Index