



Molecular Flow Module

User's Guide

Molecular Flow Module User's Guide

© 1998–2023 COMSOL

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

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

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

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: CM024401

C o n t e n t s

Chapter 1: Introduction

About the Molecular Flow Module	8
Rarefied Gas Flows	8
What Can the Molecular Flow Module Do?.	8
The Molecular Flow Module Physics Interface Guide	9
Common Physics Interface and Feature Settings and Nodes	9
Where Do I Access the Documentation and Application Libraries?	10
 Overview of the User's Guide	 13

Chapter 2: Modeling Guidelines

Molecular and Rarefied Gas Flows	16
 Modeling Transitional Flows	 18
Boundary Conditions	19
Choosing a Mesh and a Quadrature.	19
Solving Transitional Flow Problems	20
 Modeling Free Molecular Flows	 21
Boundary Conditions for Molecular Flows	22
Interior Number Density Reconstruction	23
Meshing	24
Integration Resolution	24
Results Processing	25
Experimental Measurements of Number Density/Pressure	25
Multiple Species	27
References for the Theory of Rarefied Gas Flows	27

Chapter 3: Rarefied Flow Interfaces

The Transitional Flow Interface	30
Domain and Boundary Nodes for the Transitional Flow Interface	32
Flow Properties	34
Initial Values	35
Wall	35
Continuity on Interior Boundary	35
Inlet.	35
Outlet.	36
High Vacuum Pump	36
Diffuse Flux	37
Total Vacuum	38
Reservoir.	38
Outgassing Wall	39
 Theory for the Transitional Flow Interface	 40
Overview of the Lattice Boltzmann Method.	40
Relaxation Time and Mean Free Path	43
About the Boundary Conditions for Transitional Flow.	44
Velocity Quadratures	47
References for the Transitional Flow Interface.	49
 The Free Molecular Flow Interface	 50
Domain, Boundary, Edge, Point, Pair, and Global Nodes for the Free Molecular Flow Interface	54
Number Density Reconstruction	55
Molecular Flow	55
Surface Temperature	55
Evaporation	56
Initial Values	56
Wall	57
Diffuse Flux	59
Total Vacuum	59
Reservoir.	59
Vacuum Pump	60
Plane Symmetry	61

Symmetry 61

Theory for the Free Molecular Flow Interface 64

Solving the Flow 64

Calculating the Particle Flux. 68

Calculating the Pressure 71

Calculating the Number Density 72

Calculating the Heat Flux. 74

About Boundary Conditions for Pressure and Number Density 75

Adsorption and Desorption. 78

Evaporation. 80

References for the Free Molecular Flow Interface 80

Chapter 4: Glossary

Glossary of Terms 82

Index 85

Introduction

This guide describes the Molecular Flow Module, an optional add-on package that extends the COMSOL Multiphysics® modeling environment with customized physics interfaces for modeling kinetic gas flows.

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.

In this chapter:

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

About the Molecular Flow Module

These topics are included in this section:

- [Rarefied Gas Flows](#)
- [What Can the Molecular Flow Module Do?](#)
- [The Molecular Flow 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*

Rarefied Gas Flows

Gas rarefaction is of critical importance for both high velocity gas flows around aircraft traveling at high altitudes and for low velocity flows at low gas pressures, such as those encountered in vacuum systems. Historically most academic research has been focused on aerospace applications, and consequently modeling software appropriate for such flows is readily available. Unfortunately the techniques usually employed to model the high velocity flows around aircraft are inappropriate for the modeling of low velocity flows, such as those typically encountered in vacuum systems, since the particle-based methods employed produce an unacceptable level of statistical noise at low velocities.

The Molecular Flow Module is designed to offer previously unavailable simulation capabilities for the accurate modeling of low pressure, low velocity gas flows. The module facilitates accurate simulations of molecular flows in complex geometries.

What Can the Molecular Flow Module Do?

The Molecular Flow Module is a collection of tailored physics interfaces for the simulation of kinetic gas flows. The module includes two physics interfaces: the Transitional Flow interface and the Molecular Flow interface. The Transitional Flow interface uses a *discrete velocity/lattice Boltzmann* approach to solve low velocity gas flows in the transitional flow regime. The Molecular Flow interface uses the *angular coefficient* method to solve molecular flows.





The Molecular Flow Module Application Library and supporting documentation explain how to use the physics interfaces to model a range of problems encountered by vacuum practitioners.

The Molecular Flow Module Physics Interface Guide

The physics interfaces in the Molecular Flow Module form a complete set of simulation tools, which 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 Molecular Flow 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
 Fluid Flow				
 Rarefied Flow				
Free Molecular Flow		fmf	3D, 2D, 2D axisymmetric	stationary; time dependent
Transitional Flow		tran	3D, 2D	stationary; time dependent

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	<p>To open the Help window:</p> <ul style="list-style-type: none"> • 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	<p>To open the Help window:</p> <ul style="list-style-type: none"> • 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.
Linux	

Opening the Documentation Window

Mac	<p>To open the Documentation window:</p> <ul style="list-style-type: none"> • Press Ctrl+F1. • In the main toolbar, click the Documentation () button. • From the main menu, select Help>Documentation.
Linux	

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



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

TABLE OF CONTENTS AND INDEX

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

MODELING IN MOLECULAR FLOW

The [Modeling Guidelines](#) chapter discusses these topics: [Molecular and Rarefied Gas Flows](#), [Modeling Transitional Flows](#), and [Modeling Free Molecular Flows](#).

RAREFIED FLOW

[Rarefied Flow Interfaces](#) describes [The Transitional Flow Interface](#) and [The Free Molecular Flow Interface](#) including the underlying theory for the physics interfaces.

Modeling Guidelines

In this chapter:

- [Molecular and Rarefied Gas Flows](#)
- [Modeling Transitional Flows](#)
- [Modeling Free Molecular Flows](#)

Molecular and Rarefied Gas Flows

The Molecular Flow Module provides tools specifically designed to assist with the modeling of vacuum systems. Historically such tools have not been available because most academic research on rarefied gas flows has been focused on aerospace applications. Gases at low pressures cannot be modeled with conventional fluid dynamics tools because kinetic effects become important as the mean free path of the gas molecules becomes comparable to the length scale of the flow. For gases the ratio of the molecular mean free path to the flow geometry size is given by the *Knudsen number* ($Kn = \lambda/l$). Rarefied gas flows occur when the mean free path, λ , of the molecules becomes comparable with the length scale of the flow, l . There are four flow regimes depending on the value of the Knudsen number ([Ref. 1](#)):

- Continuum flow ($Kn < 0.01$)
- Slip flow ($0.01 < Kn < 0.1$)
- Transitional flow ($0.1 < Kn < 10$)
- Free molecular flow ($Kn > 10$)

These flow regimes are shown in [Figure 2-1](#).

In the *continuum flow* regime the Navier–Stokes equations are applicable. Gases flowing in the slip flow regime show continuum behavior except in a thin layer, the *Knudsen layer*, close to the surfaces of the containing geometry. The effect of the Knudsen layer can be modeled using special boundary conditions for the Navier–Stokes equations. The Molecular Flow Module is designed to address kinetic gas flows (Knudsen numbers greater than 0.1) and includes the Transitional Flow and Free Molecular Flow interfaces, which are described in more detail next.



The Microfluidics Module includes the Slip Flow interface, which can be used to model gas flows at moderate Knudsen numbers, as well as a range of tools for modeling laminar continuum flows.

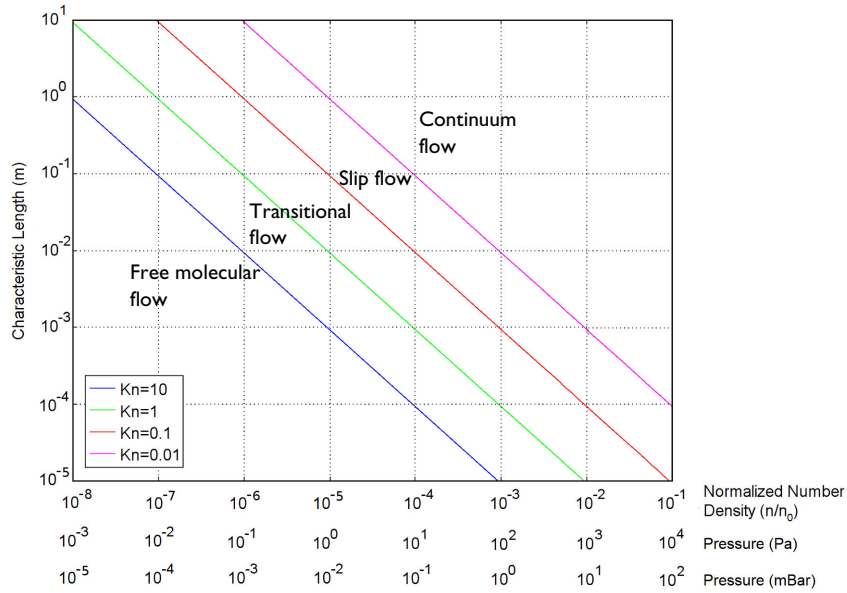


Figure 2-1: A plot showing the main fluid flow regimes for rarefied gas flows. Different regimes are separated by lines of constant Knudsen numbers. The number density of the gas is normalized to the number density of an ideal gas at a pressure of 1 atmosphere and a temperature of 0°C (n_0). Equivalent pressures are shown in various units.

Modeling Transitional Flows

In the *transitional flow regime*, the continuum approximation for the gas breaks down completely, and the Knudsen layer occupies a significant fraction — if not all — of the flow domain. Historically flows in this regime have been modeled by the *direct simulation Monte Carlo (DSMC) method*, which computes the trajectories of large numbers of randomized particles through the system. The *discrete velocity method*, which represents the *Boltzmann equation*, has also been used. The discrete velocity method has dependent variables that exist in six dimensions (three in real space and three in velocity space) by a discrete number of three-dimensional equations, each corresponding to a discrete velocity (the set of discrete velocities is sometimes referred to as the *quadrature*). This approach produces an increased number of degrees of freedom in the problem (corresponding to the number of discrete velocities used) but restricts the problem to only three dimensions. The velocity degrees of freedom are coupled together through scattering terms in the equation system. More recently the *lattice Boltzmann* technique has been shown to be an optimized form of the discrete velocity method (close to the continuum limit) to solve a simplified form of the Boltzmann equation known as the *Boltzmann BGK equation*. COMSOL Multiphysics uses a modified form of the lattice Boltzmann method to solve transitional flows. Unlike the DSMC method, the lattice Boltzmann method is not subject to statistical noise — this is an advantage for low velocity gas flows. The implementation available in COMSOL Multiphysics is limited to isothermal flows because the BGK equation has a greatly simplified scattering model with a single parameter: the relaxation time. The relaxation time can be chosen to produce the correct viscosity (the approach used in the Transitional Flow interface) or the correct thermal conductivity but not both simultaneously. More complicated scattering models (with more than one parameter) are required for nonisothermal modeling. Diffuse reflection of gas molecules is also assumed at all surfaces.

The [Transitional Flow Interface](#) (☒☒☒) should be used to compute fluid flows with Knudsen numbers greater than 0.1. Lower Knudsen number flows can be solved, but ultimately the problem becomes very highly coupled in the Navier–Stokes regime, and a fully coupled solver must be used. For three-dimensional flows and two-dimensional

flows with quadratures larger than the D2Q12 quadrature, it becomes impractical to use the fully coupled solver with the computing power that is typically available.



It is currently not possible to couple the Transitional Flow interface to the Molecular Flow or Laminar Flow interfaces, so transitional flows must be modeled in their entirety within the physics interface.

Boundary Conditions

In the transitional flow regime there are no simple boundary conditions available for inflow and outflow. [The Transitional Flow Interface](#) includes boundary conditions that are applicable either in the Navier–Stokes limit (the [Inlet](#) and [Outlet](#) conditions) or in the molecular flow regime (for example, the [Reservoir](#) condition). If the flow throughout the region of interest is transitional then it is necessary to introduce additional domains on either side of the region of interest to allow the flow to reach a physical solution before it enters the region of interest. When this approach is taken then it is reasonable to use (with care) the boundary conditions available, even if they are not strictly physical. Generally it is best to use the molecular flow boundary conditions because they do not require the fully coupled solver.

The [Wall](#) boundary condition is valid across the full range of Knudsen numbers but is limited to the case of diffuse reflection of molecules incident on the boundary.

Choosing a Mesh and a Quadrature

[The Transitional Flow Interface](#) can be thought of as having a mesh in physical space (represented by the mesh generated by COMSOL Multiphysics) and a mesh in velocity space (represented by the quadrature). Accurate solutions should not vary greatly with the physical mesh or with the quadrature. Close to the slip flow regime, smaller quadratures provide a good degree of accuracy, but a finer mesh in physical space is required. As the flow becomes progressively more rarefied, more degrees of freedom are required to capture the behavior in velocity space more accurately, and consequently a larger quadrature is required. A coarser spatial mesh can, however, be employed.

For solutions to be accurate they should be both mesh and quadrature independent; that is, both increasing the size of the quadrature and decreasing the element size in the mesh should have an insignificant effect on the solution.



Larger machines, with 10s of GB of memory, or clusters, are recommended for use with the Transitional Flow interface. Due to the large number of degrees of freedom required to solve a practical problem, significant memory is required, particularly in three dimensions.

Solving a stationary problem can take over 24 hours even within these constraints. For highly rarefied flows it can also be necessary to adjust the solver tolerances. For a stationary problem the default solver relative tolerance of 10^{-7} is reasonable for flows with Knudsen numbers of approximately 1. For a highly rarefied flow the solver tolerance might need to be tightened significantly (if the **Neglect Scattering (Free Molecular Flow)** check box is selected under [Flow Properties](#), the relative tolerance is automatically changed to 10^{-10}).



When solving a large transitional flow it is often convenient to use a probe to monitor a quantity that characterizes the flow. This is described next.

To create a probe to monitor the maximum velocity in the solution, first set up a Maximum coupling on the model domains (for example, `maxop1`) and add a Global Variable Probe for `maxop1(tran.U)`. Adjust the solver settings to display the probe output at a step taken by the solver during convergence (this requires changing the **Results While Solving** settings on the **Segregated** solver node in the solver sequence).

If the tolerance has been set too tight, the solver can then be stopped once the probe value stops changing with successive iterations. It should also be clear if the tolerance needs to be further tightened, if the probe value fluctuates wildly. You can then change the solver tolerance and continue solving, starting from the existing solution.

To do this, in the **Model Builder** under the **Study>Solver Configurations>Solver** node, click **Dependent Variables**. In the **Initial Values of Variables Solved For** section, choose **Solution** from the **Method** list and select the solver itself in the **Solution** list.




For more information about probes, nonlocal couplings, and solvers, see these topics in the *COMSOL Multiphysics Reference Manual*:

- [Nonlocal Couplings and Coupling Operators](#) and [Probes](#)
- [Getting Results While Solving](#) and [Studies and Solvers](#)

Modeling Free Molecular Flows

In the *free molecular flow regime* the Knudsen number is significantly greater than unity. The mean free path is therefore much greater than the length scale of the flow, and molecules collide with surfaces bounding the flow more frequently than they collide with one another. In this regime there are two common approaches to modeling the flow: the direct simulation Monte Carlo (DSMC) method (which computes the trajectories of large numbers of randomized particles through the system) and the *angular coefficient method*. COMSOL Multiphysics uses the angular coefficient method. Completely diffuse scattering (total accommodation) and emission are assumed at all surfaces in the geometry (in practice this is often a reasonable assumption for both the reflection and emission of molecules from most surfaces and sources). The flow is computed by integrating the flux arriving at a surface from all other surfaces in its line-of-sight. Because only surfaces are considered in the calculation, the dependent variables exist only on the surfaces of the model. This method is limited to quasistatic flows, in which the fluxes change on time scales that are large compared to the average time that the molecules take to traverse the geometry. It is much faster than the DSMC method and is not subject to statistical scatter.

The **Free Molecular Flow Interface** () should be used to compute quasistatic molecular flows with Knudsen numbers greater than 10. The flux, pressure, and number density are solved for on all surfaces. The effect of surface temperature is also accounted for, and the heat flux due to the molecules themselves is available. It is also possible to compute the number density of gas molecules on points, lines, and even domains interior to the bounding surfaces of the flow.



It is very computationally expensive to compute the number density within the entire domain of the flow in three dimensions.



Although COMSOL Multiphysics currently does not offer a physics interface for Monte Carlo modeling of molecular flows, it is possible to use the Particle Tracing interface for limited Monte-Carlo computations (this requires the Particle Tracing Module). The *Molecular Flow Through an S-Bend* shows how to do this (Application Library path: **Molecular_Flow_Module/Benchmarks/s_bend_benchmark**).

In this section:

- [Boundary Conditions for Molecular Flows](#)
- [Interior Number Density Reconstruction](#)
- [Meshing](#)
- [Integration Resolution](#)
- [Results Processing](#)
- [Experimental Measurements of Number Density/Pressure](#)
- [Multiple Species](#)
- [References for the Theory of Rarefied Gas Flows](#)

Boundary Conditions for Molecular Flows

The [Free Molecular Flow Interface](#) works by computing the incoming molecular flux for each species, G (always a dependent variable), at each element by integrating the outgoing flux, J (available as `fmf.J`), arriving from all boundaries within the line of sight. Similar integrals with different weightings on J are used to compute the number density, pressure, and net heat flux on the surfaces. These quantities do not normally affect the solution (unless, for example, the heat flux is coupled into a heat transfer computation that in turn effects the surface temperature). Each of the boundary conditions specifies the outgoing flux, J , usually in terms of the incoming flux, G .

Boundary conditions are available to describe a range of surfaces including:

- Walls (including outgassing walls, walls with *adsorption* and *desorption*, and walls on which deposition occurs)
- Pumps
- Incoming flux (specified directly by the user with an arbitrary expression or a constant value)
- Molecular *effusion* from a large, adjacent reservoir and total vacuum (in practice a large chamber with a pressure low enough that molecules entering the domain from the chamber can be neglected)

The [Surface Temperature](#) feature can be used to set the surface temperature on different surfaces if the system under consideration is nonisothermal.

Interior Number Density Reconstruction

It is often desirable to know the number density at some point within the vacuum chamber (for example, along a beam line). There are two ways to compute the number density within a domain in COMSOL Multiphysics.

The first method is to add a [Number Density Reconstruction](#) feature to the model. COMSOL Multiphysics then computes the number density for each species (which is available as the dependent variable N) on the feature in addition to on the exterior surfaces of the model at the time of solving.



This method is recommended for large regions, such as surfaces and domains, where the calculation of the number density is computationally intensive, or along curved lines that are explicitly drawn within the geometry (such lines are more difficult to define in results processing).

The [Free Molecular Flow Interface](#) also defines a number density operator, which is called in the following manner:

```
comp1.fmf.nop_name(x,y,z) (3D)
comp1.fmf.nop_name(x,y)   (2D)
comp1.fmf.nop_name(r,z)   (2D axisymmetric)
```

where name is the dependent variable name for a given species. The default is G , so `comp1.fmf.nop_G(x,y,z)` would be the default operator to compute the number density in 3D.



If multiple components or Free Molecular Flow interfaces are added to the model, change `comp1` and `fmf` to the corresponding component and physics interface names.

This operator computes the number density at the time it is called, and consequently if a number of computations are required (for example, for plotting the number density on a surface) it takes a long time to return the result (the physics interface is unresponsive during this time).



This method is recommended if the number density at a point or on a straight line not drawn in the geometry is required.



The number density operator and the number density reconstruction features produce unreliable results adjacent to the exterior surfaces of a Molecular Flow domain. In this region the number density on the surface is much more accurate.

Meshing

To solve molecular flow problems it is only necessary to define degrees of freedom on the surfaces of the flow domain. Consequently the Free Molecular Flow interface supports two methods for meshing:

- The entire domain selected in the Free Molecular Flow interface can be meshed.
- Exterior boundaries of the selected domain can be meshed.



The physics interface does not support a mixture of these two approaches, so, for example, volume meshing only a part of the flow domain and surface meshing the rest can lead to incorrect results.

COMSOL Multiphysics does not return an error message, but in some cases the results generated are incorrect (in a mixed approach, the method used to find the normals that point into the molecular flow domain can fail in certain circumstances).

By default the Free Molecular Flow interface uses constant elements for all the degrees of freedom. In general, using constant elements with a fine mesh provides better accuracy than using higher-order elements on a coarser mesh with the same number of degrees of freedom. It is important to check that solutions obtained do not change significantly (within the desired degree of accuracy) when the mesh is further refined.

The method used by COMSOL Multiphysics to integrate the flux arriving at a particular mesh element can be inaccurate near interior or convex corners. If the results in such regions are important, it can be necessary to use a highly refined mesh and a high integration resolution. See [Integration Resolution](#) for further information.

Integration Resolution

The integrations performed to calculate the flux arriving at a particular element use a computational technique known as the hemicube method, which constructs a 2D view of the other elements from the perspective of each element. The resolution employed

is determined by the [Integration Settings](#) on the Free Molecular Flow interface node. A larger number indicates a greater integration resolution, and consequently more accurate results. For the results of a model to be accurate the solution should not change significantly when the integration resolution is increased.



For 2D axisymmetric components, the number of **Integration Sectors** must also be specified. To compute the integrals in an axisymmetric component COMSOL Multiphysics must construct a virtual geometry in three dimensions. The number of sectors in the geometry are determined by this setting. The sensitivity of the solution to this setting should be evaluated.

Results Processing

By default the Free Molecular Flow interface uses constant shape functions for all the computed quantities. Constant shape functions produce poor results with the default refinement method used during results processing. When constant shape functions are used, it is therefore recommended that the **Resolution** setting in the **Quality** section of most plot types is set to **No Refinement**.



[Results Analysis and Plots](#) in the *COMSOL Multiphysics Reference Manual*

Experimental Measurements of Number Density/Pressure

When comparing simulation results to measurements it is important to understand whether pressure or number density is the relevant quantity to compare. In the vacuum industry it is common to use the ideal gas law to relate the pressure, p , to the number density, n :

$$p = nk_{\text{B}}T \quad (2-1)$$

where k_{B} is the Boltzmann constant, and T is the absolute temperature. [Equation 2-1](#) is not rigorously true in a molecular flow. The derivation of the ideal gas law assumes that molecules arrive at a surface from random directions (at high pressure this is true because of collisions between the molecules). Provided highly directional effects such as molecular beaming are absent from a molecular flow, [Equation 2-1](#) holds approximately in many circumstances (it is possible, for example, to compare the

pressure with $nk_{\text{B}}T$ in some of the model examples and often the two values are within 10% of each other). A counterexample that demonstrates when this relationship completely fails is provided by the model *Rotating Plate in a Unidirectional Molecular Flow* in the Molecular Flow Module Application Library.

COMSOL Multiphysics defines pressure as the normal force acting on a surface and number density as the number of molecules per unit volume. Unfortunately it is common practice in the vacuum industry to lump both of these quantities together under the concept of pressure. Many vacuum gauges that operate at low pressures actually measure number density (in the COMSOL Multiphysics sense) but are calibrated to give readings in units of pressure. For these gauges it is appropriate to compare the quantity $nk_{\text{B}}T$ from a simulation with the pressure that the gauge reads.

A practicing engineer needs to answer the question: “Does my gauge actually measure pressure or number density?”. [Ref. 2](#) discusses how the common types of vacuum gauge operate and classifies gauges as direct or indirect. Direct gauges usually measure the displacement of a wall, which is directly related to the pressure in a COMSOL Multiphysics simulation. Indirect gauges measure the “pressure” indirectly, via a gas property. Many indirect gauges are so-called ionization gauges, in which the gas is ionized by some mechanism, and the ion current generated by an electric field is measured. For these gauges, the quantity $nk_{\text{B}}T$ is appropriate for comparison with experimental gauge readings. Other indirect gauges operate on other principles that make it harder to associate them with either a pressure or a number density directly. For example, Pirani and thermocouple gauges measure the heat loss from a wire in the gas. Although this process could be modeled in detail by the Free Molecular Flow interface, it is often more practical to exercise engineering judgment when comparing simulation results with data from these types of gauges. [Table 2-1](#) classifies common vacuum gauges in terms of the quantity they measure.

TABLE 2-1: CLASSIFICATION OF COMMON VACUUM GAUGES BASED ON THE QUANTITY MEASURED.

GAUGES THAT MEASURE PRESSURE p	GAUGES THAT MEASURE $nk_B T$	GAUGES THAT MEASURE OTHER QUANTITIES
Gauges based on the deflection of a solid wall: a. Diaphragm b. Baratron/Capacitive gauge c. Bourdon gauge	Cold cathode ionization gauges: a. Penning b. Redhead c. Magnetron Hot cathode ionization gauges: a. Bayard–Alpert b. Extractor Radiation-based ionization gauges: a. Alpatron	Gauges based on tangential/viscous forces: a. Spinning rotor Gauges based on heat transfer: a. Thermocouple b. Pirani

Multiple Species

Since the gas molecules do not interact with each other in the flow domain, solving for many different species at once does not significantly increase the computational complexity of the problem. It is possible to model flows involving several different species simultaneously by adding more species in the Dependent Variables section.



For an example of using multiple species in a model, see the example *Ultrahigh Vacuum, Chemical Vapor Deposition* (Application Library path: **Molecular_Flow_Module/Industrial_Applications/uhv_cvd**).

Modeling multiple species is needed if adsorption and desorption of the molecules occurs. It is also possible for the different species to interact on the surfaces — for example, if they share common adsorption sites, the site occupancy can be set up as an expression involving the concentration of both gas molecules.

References for the Theory of Rarefied Gas Flows

1. G. Kariadakis, A. Beskok, and N. Aluru, *Microflows and Nanoflows*, Springer Science and Business Media, 2005.

2. J.F. O’Hanlon, *A User’s Guide to Vacuum Technology*, Wiley Interscience, 2003.

Rarefied Flow Interfaces

This chapter describes the physics interfaces available to model the flow of rarefied gases. These physics interfaces are found under the **Fluid Flow>Rarefied Flow** branch ([xxx](#)).

- [The Transitional Flow Interface](#)
- [Theory for the Transitional Flow Interface](#)
- [The Free Molecular Flow Interface](#)
- [Theory for the Free Molecular Flow Interface](#)

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

The Transitional Flow Interface

The **Transitional Flow (tran)** interface (☒☒☒), found under the **Rarefied Flow** branch (☒☒☒) when adding a physics interface, can be used to model isothermal flows across the full range of Knudsen numbers from the laminar flow limit to the molecular flow limit. The method is less computationally efficient than other physics interfaces targeting specific Knudsen number regimes. In the transitional flow regime, at Knudsen numbers between 0.1 and 10, this physics interface is the available solution in COMSOL Multiphysics. The physics interface solves the Boltzmann BGK equation, which is a simplified form of the Boltzmann equation, using the lattice Boltzmann method. When larger lattices are used in the lattice Boltzmann method it can be viewed as a discrete velocity method for solving the Boltzmann BGK equation.

Diffuse reflection from all surfaces is assumed (this is reasonable in many practical situations) with molecules from all directions effectively adsorbed onto the surface and subsequently reemitted according to Knudsen's law (that is, with an intensity that varies as the cosine of the angle of emission to the normal to the surface). This physics interface is available for 2D and 3D components.

When this physics interface is added, these default nodes are also added to the **Model Builder: Flow Properties, Wall, and Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions and volume forces. You can also right-click **Transitional Flow** to select physics features from the context menu.



The physics interface solves the Boltzmann BGK equation, which is a simplified form of the Boltzmann equation. This provides a good engineering model of a rarefied gas, but it cannot be used to model thermal flows accurately. Currently only one species of molecule can be modeled. For this physics interface, the accuracy of the results obtained depends strongly on the quadrature employed. Accurate solutions are independent of both mesh density and do not change as the number of velocities in the quadrature is increased. It is important to validate results to establish if the simulation is sufficiently accurate for a given application.



For a practical guide to solving transitional flow models, see [Modeling Transitional Flows](#). For more information on the theory of the physics interface, see [Theory for the Transitional Flow 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 `tran`.

TRANSITIONAL FLOW

Select the **Neglect scattering term (Molecular flow)** check box to neglect scattering of the gas molecules within the domain. This is appropriate for models in the molecular flow regime.


Select an option from the **Quadrature** list. Several predefined quadratures are available. The notation for the quadratures is of the form `DXQY` where *X* and *Y* are integers. *X* represents the number of dimensions whilst *Y* represents the number of velocities in the quadrature. A transitional flow simulation can be considered accurate if the results do not change with increased *Y* and with reduced mesh size. This dialog defines the dependent variables: `tran.f1 . . . tran.fn` (SI unit: $1/m^3$). These are defined on the flow properties node). There are *n* dependent variables defined, where *n* is the number of velocities in the quadrature.



If the quadrature is changed after a model has been solved, the settings for the solver are not automatically updated. To regenerate the correct default solver settings, under **Study**, right-click the **Solver Configurations** node, select **Delete Solvers** and click **OK**. To keep the existing plot groups and results settings, when prompted click **No**. After the model is solved again, select the appropriate **Solution** node under **Results>Datasets** and select the appropriate solver output in the **Solution** setting.

Select a **Form of equilibrium function** — **First order**, **Second order** (the default), or **Third order**. The *equilibrium function* is the discretized form of the Maxwell–Boltzmann distribution for the gas at its local velocity and temperature.

PERFORMANCE INDEX

To display this section, click the **Show More Options** button () and select **Advanced Physics Options** in the **Show More Options** dialog box. Select the **Performance index: 1, 2, 4** (the default), **8, 16**, or **32**. This setting determines the number of dependent variables in each segregated group generated by the default solver. Select the **Always use dependent variable grouping** check box to use the segregated solver even for smaller quadratures in 2 dimensions.



The **Performance index** setting should be selected on the basis of model size and available memory. Choosing a higher setting requires more memory to solve the model, but the model solves more quickly.

DISCRETIZATION

Select the element order for the dependent variable. The default is quadratic.



- [Modeling Transitional Flows](#)
- [Domain and Boundary Nodes for the Transitional Flow Interface](#)
- [Velocity Quadratures](#)
- [Theory for the Transitional Flow Interface](#)
- [Global Equations in the COMSOL Multiphysics Reference Manual](#)

Domain and Boundary Nodes for the Transitional Flow Interface

The [Transitional Flow Interface](#) has these domain and boundary nodes available from the **Physics** ribbon toolbar (Windows users), **Physics** context menu (Mac or Linux users), or by right-clicking the main physics interface node to access the context menu (all users).



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

DOMAIN

- [Flow Properties](#)
- [Initial Values](#)

BOUNDARY

Across the full range of Knudsen number, Kn:

- [Wall](#)
- [Continuity on Interior Boundary](#)

In the Navier–Stokes limit ($Kn < 0.1$):

- [Inlet](#)
- [Outlet](#)



These boundary conditions are functionally equivalent to the Inlet and Outlet conditions for single-phase laminar flows (see [The Single-Phase Flow, Laminar Flow Interface](#) in the *COMSOL Multiphysics Reference Manual*). The mathematical form of the boundary conditions in the two different interfaces is, however, very different. See the [Theory for the Transitional Flow Interface](#) for more details.

In the Free Molecular Flow Limit ($Kn > 10$):

- [High Vacuum Pump](#)
- [Outgassing Wall](#)
- [Reservoir](#)
- [Diffuse Flux](#)
- [Total Vacuum](#)

Although the conditions on the incident and outgoing fluxes are equivalent, the method for computing the incident flux and for setting the outgoing flux differs significantly between the two interfaces. See [Theory for the Free Molecular Flow Interface](#) and [Theory for the Transitional Flow Interface](#) for more detail.

On all the boundaries of the Transitional Flow interface, a number of physical variables are defined. These include:

- `tran.G` — incident flux
- `tran.J` — emitted flux

- `tran.Nin` — incident number density
- `tran.Nout` — emitted number density
- `tran.n` — total local number density (inherited from domains)
- `tran.Pin` — pressure due to incident molecules
- `tran.Pout` — pressure due to emitted molecules
- `tran.p` — total pressure



The pressure is not defined in the domain as a scalar because it is a tensor quantity for a rarefied flow. Also note that if a boundary is parallel to one of the lattice velocities, the sum of the incident and emitted number densities can differ from the total local number density due to molecules traveling parallel to the boundary.

Flow Properties

Use the **Flow Properties** node to define the parameters for the Boltzmann BGK model. The dynamic viscosity, the temperature, and the mean molar mass must be specified. These parameters define the relaxation time, τ , the characteristic molecular speed, c_s , and the molecular mass, m , which are the fundamental material parameters in the BGK model.

FLOW PROPERTIES

The default **Dynamic viscosity** μ (SI unit: Pa·s) uses the value **From material**. For **User defined** define a different value or expression.

The default **Mean molar mass** M_n (SI unit: kg/mol) is **User defined** with a default value of 0.028 kg/mol.



A number of variables are defined in the flow properties node, enabling convenient analysis of the flow. These include:

- `tran.ux`, `tran.uy`, and `tran.uz` — velocity components
- `tran.n` — total number density
- `tran.rho` — density
- `tran.lambda` — mean free path

Initial Values

The **Initial Values** node adds initial values for the number density, pressure, and density. For the transient solver, these values define the state of the problem at the initial time step; for the stationary solver, they serve as a starting point for the nonlinear solver.

INITIAL VALUES

Select an option for each of the following from the **Initialization** list and then enter a value or expression in the field. Select:

- **Initial number density** to enter a **Number density** n_0 (SI unit: $1/\text{m}^3$).
- **Initial pressure** to enter a **Pressure** p_0 (SI unit: Pa).
- **Initial density** to enter a **Density** ρ_0 (SI unit: kg/m^3).

Wall

The **Wall** node sets the total incoming molecular flux, G , equal to the flux emitted from the wall, J . All the incident molecules are emitted diffusely, with an equilibrium distribution that is determined by the wall velocity and the local density.

WALL

Select a **Boundary condition**: **Stationary wall** (the default) or **Moving wall**. For **Moving wall** enter a value or expression for the **Wall velocity** u_w (SI unit: m/s).

Continuity on Interior Boundary

The **Continuity on Interior Boundary** node ensures continuity of the molecular flux across boundaries within the geometry. It is the default boundary condition for all interior boundaries.

Inlet

The **Inlet** node includes a set of boundary conditions describing the fluid flow condition at an inlet, in the Navier–Stokes limit.



In most cases the inlet boundary conditions appear, some of them slightly modified, in the **Outlet** type as well. This means that there is nothing in the mathematical formulations to prevent a fluid from leaving the domain through boundaries where the **Inlet** type is specified.

INLET

Select a **Boundary condition** — **Velocity** (the default), **Pressure**, **Density**, or **Number density**. For each of the following enter a value or expression:

- For **Velocity** enter a **Normal inflow velocity** U_0 (SI unit: m/s).
- For **Pressure** enter a **Pressure** p_0 (SI unit: Pa).
- For **Density** enter a **Density** ρ_0 (SI unit: kg/m³).
- For **Number density** enter a **Number density** n_0 (SI unit: 1/m³).

Outlet

The **Outlet** node includes a set of boundary conditions describing the fluid flow condition at an outlet, in the Navier–Stokes limit.



All of the formulations for the **Outlet** type are also available, possibly slightly modified, in **Inlet** type as well. This means that there is nothing in the mathematical formulations to prevent a fluid from entering the domain through boundaries where the **Outlet** boundary type is specified.



See [Inlet](#) for all settings.

High Vacuum Pump

Use the **High Vacuum Pump** node to represent a high vacuum pump such as a turbomolecular pump or a diffusion pump in the molecular flow regime (Kn>10).

If the pump flux is specified directly then J_{pump} is user specified. Otherwise, the fraction of incident molecules that are pumped, f , is specified either directly or through a pump speed, S , such that $f = \sqrt{(2\pi M_n/RT)} S/A$ (here M_n is the molar mass of the species, R is the universal gas constant, T is the temperature, and A is the area of the pump). This condition applies pointwise across the surface of the pump.



Care should be taken with the specification of J_{pump} to ensure that the value is never negative.

HIGH VACUUM PUMP

Select an option from the **Specify pump flux** list: **Directly** (the default), **Fraction of incident molecules** (the default), or **Pump speed**. The emitted flux J is given by

$$J = G - J_{\text{pump}}$$

- For **Directly** enter a value or expression for the **Molecular flux** J_{pump} (SI unit: $1/(\text{m}^2 \cdot \text{s})$). The default is 0 ($1/(\text{m}^2 \cdot \text{s})$).
- For **Fraction of incident molecules** enter a value or expression for the **Fraction of incident molecules pumped** f (dimensionless). The default is 0.
- For **Pump speed** enter a value or expression for the **Pump speed** S (SI unit: m^3/s). The default is 0 m^3/s .

Diffuse Flux

The **Diffuse Flux** boundary condition sets the molecular flux emitted from the boundary to a user-specified value. This boundary condition applies only in the molecular flow limit (Knudsen number, $\text{Kn} > 10$). The flux is emitted according to Knudsen's law (the emitted flux in a given direction is proportional to the cosine of its angle to the surface normal). This can be set directly as a flux, as a total incoming mass flow, or as number of SCCMs. The flux is assumed to be emitted from a stationary boundary.

DIFFUSE FLUX

Select the **Boundary condition**: **Directly** (the default), **Number of SCCM units**, **Mass flux**, or **Total mass flow**.

Directly

For **Directly** enter a value or expression for the **Molecular flux** J_o (SI unit: $1/(\text{m}^2 \cdot \text{s})$). The default is 0 ($1/(\text{m}^2 \cdot \text{s})$).

Number of SCCM (standard cubic centimeters per minute) Units

For **Number of SCCM units** enter the **Number of SCCM units** Q_{sccm} (dimensionless), and this quantity is automatically converted to a standardized volumetric flow rate in SI units. Select an option from the **Standard flow rate defined by** list — **Standard density** (the default) or **Standard pressure and temperature**.

- For **Standard density** enter a **Standard molar volume** V_m (SI unit: m^3/mol). The default is 0.0224136 m^3/mol .
- For **Standard pressure and temperature** enter a **Standard pressure** P_{st} (SI unit: Pa) and **Standard temperature** T_{st} (SI unit: K). The defaults are 1 atm and 273.15 K, respectively.



For 2D components, also enter a **Channel thickness** d_{bc} (SI unit: m).

Mass Flux

For **Mass flux** enter a value or expression for the **Mass flux** M_f (SI unit: $\text{kg}/(\text{m}^2 \cdot \text{s})$).

Total Mass Flow

For **Total mass flow** enter a value or expression for the **Total mass flow** Q_m (SI unit: kg/s).

Total Vacuum

In the molecular flow limit (Knudsen number, $\text{Kn} > 10$), use the **Total Vacuum** boundary condition to set the flux emitted from the wall, J , equal to zero. This is equivalent to an opening to a large vessel or extended region with a much lower pressure than that in the system.

Reservoir

The **Reservoir** boundary condition is appropriate in the molecular flow limit (Knudsen number, $\text{Kn} > 10$) when a boundary represents a stationary opening to a large adjacent vessel or region filled with gas at a specified number density or pressure. In this case the effusing flux obeys Knudsen's law.

RESERVOIR

Select the **Boundary condition: Reservoir pressure** (the default) or **Reservoir number density**.

- For **Reservoir pressure** enter a value or expression for the **Reservoir pressure** p_0 (SI unit: Pa).

- For **Reservoir number density** enter the **Reservoir number density** n_0 (SI unit: $1/(m^3)$).



The **Reservoir** boundary condition represents an opening to an adjacent chamber whose extent is much larger than the molecular mean free path. The boundary condition assumes that the molecules in the reservoir have a Maxwellian distribution of velocities and that the size of the opening to the reservoir is much less than the mean free path. Molecules travel through the opening by a process known as molecular effusion. Effusion is a process that selects molecules with higher molecular speeds than average because they are traveling toward the hole faster than the other molecules. Effusing molecules therefore have a larger average velocity than the molecules in the adjacent reservoir. As a result the pressure or number density at the reservoir boundary can differ from the value set in the boundary condition — which determines the pressure that would be measured within the reservoir away from the opening.

Outgassing Wall

In the molecular flow limit (Knudsen number, $Kn > 10$), use the **Outgassing Wall** boundary condition to represent a wall or surface that is outgassing at a specified rate in addition to reemitting incident molecules diffusely. In this case the emitted molecular flux, J , is given by $J = G + J_{out}$, where J_{out} is the user-specified outgassing flux. J_{out} can be specified as a molecular flux, a mass flux, a number of SCCM units, or a total mass flow. The wall is assumed to be stationary.

OUTGASSING WALL

Select the **Boundary condition: Directly** (the default), **Number of SCCM units**, **Mass flux**, or **Total mass flow**.



The rest of the settings are the same as for the [Diffuse Flux](#) node.

Theory for the Transitional Flow Interface

The [Transitional Flow Interface](#) theory is described in this section:

- [Overview of the Lattice Boltzmann Method](#)
- [Relaxation Time and Mean Free Path](#)
- [About the Boundary Conditions for Transitional Flow](#)
- [Velocity Quadratures](#)
- [References for the Transitional Flow Interface](#)

Overview of the Lattice Boltzmann Method

The [Transitional Flow Interface](#) uses the lattice Boltzmann method to solve the Boltzmann BGK equation, which is a simplified form of the more general Boltzmann equation. The BGK equation (in the absence of volume forces) is given below:

$$\frac{\partial}{\partial t} f(\xi, \mathbf{x}, t) + \xi \cdot \nabla f(\xi, \mathbf{x}, t) = -\frac{f(\xi, \mathbf{x}, t) - f^{\text{eq}}(\xi, \mathbf{u}, n)}{\tau}$$

where ξ is the particle velocity, \mathbf{x} is the position, t is the time, $f(\xi, \mathbf{x}, t)$ is the distribution function (which represents the total number of particles in the phase space $d\xi d\mathbf{x}$), f^{eq} is the local equilibrium distribution, n is the number density, \mathbf{u} is the bulk velocity, and τ is the relaxation time. This equation is similar to the Boltzmann equation, except the complex collision term has been replaced by a much simpler term that implies a constant relaxation time for all velocities to return to equilibrium. The equilibrium function in N dimensions is given by

$$f^{\text{eq}}(\xi, \mathbf{u}, n) = n \left(\frac{m}{2\pi k_B T} \right)^{N/2} \exp \left(-\frac{m |\xi - \mathbf{u}|^2}{k_B T} \right) \quad (3-1)$$

The equation set is still nonlinear as $f^{\text{eq}}(\xi, \mathbf{u}, \rho)$ is a nonlinear function of $f(\xi, \mathbf{x}, t)$ (see definitions of n and \mathbf{u}). As in the Boltzmann equation, the macroscopic thermodynamic variables are defined as moment integrals of the distribution function. For example, the number density and fluid velocity are given by

$$n(\mathbf{x}, t) = \int_{\mathbf{v}} f(\xi, \mathbf{x}, t) d\xi$$

$$\mathbf{u}(\mathbf{x}, t) = \frac{1}{n(\mathbf{x}, t)} \int_{\mathbf{v}} \xi f(\xi, \mathbf{x}, t) d\xi$$

where m is the molecular mass.

In the lattice Boltzmann approach, the BGK equation is projected onto a finite velocity quadrature. In this case each velocity direction in the lattice has an associated distribution function, which can be interpreted as the local number density of the gas moving with the velocity specified. Weights associated with each of the velocities are used when calculating the moments of the distribution functions, such as the macroscopic density and velocity. When the lattice used is larger than that required to solve the Navier–Stokes equations, the lattice Boltzmann method is effectively an optimized discrete velocity method for solving the BGK equation

The Boltzmann Equations become

$$\frac{\partial}{\partial t} f_i(\mathbf{x}, t) + \xi_i \cdot \nabla f_i(\mathbf{x}, t) = -\frac{f_i(\mathbf{x}, t) - f_i^{\text{eq}}(\xi, \mathbf{u}, \rho)}{\tau} \quad (3-2)$$

where there are now i distribution functions, f_i , associated with each of the discrete velocity directions, ξ_i , and each having an associated equilibrium function, f_i^{eq} . The number density and velocity become:

$$n(\mathbf{x}, t) = \sum_i f_i(\mathbf{x}, t)$$

$$\mathbf{u}(\mathbf{x}, t) = \frac{1}{n(\mathbf{x}, t)} \sum_i \xi_i f_i(\mathbf{x}, t)$$

The velocity lattice and equilibrium functions used are critical in determining the accuracy of the final solution. The approach used to generate the equilibrium functions and velocity lattice used by COMSOL Multiphysics is described in detail in [Ref. 1](#). The distribution function $f(\xi, \mathbf{x}, t)$ is expanded in the Hermite polynomials. The velocity space is approximated by a Gauss–Hermite quadrature. This approach is closely related to Grad’s moment scheme for solving the Boltzmann equation ([Ref. 2](#)). The Hermite polynomials are chosen because the lower-order velocity moments of the distribution

functions depend only on the lower-order coefficients of the Hermite polynomials, so that truncating the expansion has a minimal effect on the moments themselves.

The f_i^{eq} functions are derived by expanding the drifting Maxwellian distribution in the Hermite polynomials to a given order. Using Equation 3-1 and the orthogonality and completeness of the Hermite basis this equation results:

$$f_i^{\text{eq}} = \sum_{n=0}^{\infty} \frac{1}{n!} a_n^{\text{eq}}(\mathbf{x}, t) H^{(n)}(\xi)$$

$$a_n^{\text{eq}}(\mathbf{x}, t) = \int_{\xi} n_{\text{eq}} \left(\frac{m}{2\pi k T} \right)^{N/2} \exp \left(-\frac{m |\xi - \mathbf{u}_{\text{eq}}|^2}{k_B T} \right) H^{(n)}(\xi) d\xi$$

where n_{eq} is the equilibrium number density (usually the local number density), \mathbf{u}_{eq} is the equilibrium velocity (usually the local velocity), and $H^{(n)}(\xi)$ is the n^{th} N -dimensional Hermite polynomial, given by

$$H^{(n)}(\xi) = \frac{(-1)^n}{w(\xi)} \nabla^n w(\xi)$$

with

$$w(\xi) = \frac{1}{(2\pi)^{N/2}} \exp \left(-\frac{\xi \cdot \xi}{2c_s^2} \right)$$

The expansion for the isothermal case, evaluated in the Gauss–Hermite quadrature, is given below, with terms according to progressively higher orders of approximation indicated (Ref. 1):

$$f_i^{\text{eq}}(\xi, \mathbf{u}, \rho) = n_{eq} w_i \left(\overbrace{1 + \frac{\xi \cdot \mathbf{u}_{eq}}{c_s^2}}^{\text{First order}} + \overbrace{\frac{\xi \cdot \mathbf{u}_{eq}}{2c_s^4} - \frac{u_{eq}^2}{2c_s^2}}^{\text{Second order}} \right. \\ \left. + \underbrace{\frac{\xi \cdot \mathbf{u}_{eq}}{6c_s^2} \left(\frac{(\xi \cdot \mathbf{u}_{eq})^2}{c_s^4} - \frac{3u_{eq}^2}{2c_s^2} \right)}_{\text{Third order}} \right) \quad (3-3)$$

here w_i is the weight associated with the velocity quadrature and c_s is the characteristic speed, which in the BGK model is given by

$$c_s = \sqrt{\frac{k_B T}{m}}$$

The higher-order terms are typically significant only at higher Mach numbers. While rarefied flows can be simulated with second-order or even first-order equilibrium functions, a set of distribution functions up to third order are provided for completeness.



The equations used in the Transitional Flow interface include terms that are temperature dependent. However, the Boltzmann BGK model restricts the Prandtl number of the gas to be a fixed (unphysical) value and therefore cannot be used to treat nonisothermal flows correctly. This physics interface is therefore restricted to the treatment of isothermal flows, although the temperature must be entered in the Flow Properties node.

It is important to realize that highly accurate solutions of the Boltzmann equation are difficult to obtain in the transitional region. The accuracy of the Transitional Flow interface is strongly dependent on the number of lattice directions in the quadrature, and in general it is best to use the largest number possible within hardware and time constraints. You can assess the level of accuracy of the simulation by comparing the results obtained with solutions produced by the Free Molecular Flow interface in the molecular flow limit. The Transitional Flow interface also solves flows in the slip flow and molecular flow limits, which can be used for benchmarking.

Relaxation Time and Mean Free Path

The relation between the scattering time, τ , and the more familiar mean free path, λ , is nontrivial in general. For molecules traveling at a specific lattice velocity, ξ_i , the mean free path can be defined as

$$\lambda_i = |\xi_i| \tau$$

The Transitional Flow interface includes a mean free path variable defined as the velocity space average of the mean free path. This is given by:

$$\lambda = \frac{\sum_i |\xi_i| \tau f_i}{\sum_i f_i}$$

This variable can differ from kinetic theory definitions of the mean free path based on the viscosity and number density, due to the approximations inherent in representing the velocity space by the quadrature.

About the Boundary Conditions for Transitional Flow

Boundary conditions in the Transitional Flow interface are kinetically motivated and assume the diffuse reflection or emission of molecules from surfaces. Molecules can be emitted from surfaces with a drifting or nondrifting velocity distribution, which enables a range of boundary conditions to be formulated. In the Navier–Stokes limit, the emission of molecules from a surface with a drifting Maxwellian enables pressure and velocity boundary conditions to be formulated. In the molecular flow limit a range of boundary conditions based on the diffuse emission of molecules from stationary walls are available. The [Wall](#) boundary condition itself is valid across the full range of Knudsen numbers. When constructing models the validity of the boundary conditions must be carefully considered.

The diffuse reflection boundary condition was first derived for the lattice Boltzmann method (which is used by the Transitional Flow interface) by Ansumali and Karlin ([Ref. 3](#)). COMSOL Multiphysics uses a slightly modified version of this boundary condition, for consistency with the Free Molecular Flow interface. In [Ref. 3](#) a detailed formal derivation of the boundary condition is given, based on general kinetic theory arguments. Here, the boundary condition is derived using the assumption of Knudsen’s law for the emission of molecules from the boundary.

To account for the motion of the wall (at velocity \mathbf{u}_w), consider the distribution functions in a moving frame, such that the wall is stationary. In this case the continuum velocity distribution function, $f(\xi, \mathbf{x}, t)$, can be written as $g(\chi, \mathbf{x}, t)$, where $\chi = \xi - \mathbf{u}_w$. Knudsen’s law states that the emitted flux of molecules, $J(\chi, \mathbf{x}, t)$, is related to the equilibrium distribution by ([Ref. 5](#))

$$J(\chi, \mathbf{x}, t) = \alpha(-\chi \cdot \mathbf{n}) g^{\text{eq}}(\chi, 0, \rho) \quad (3-4)$$

where α is a normalization constant, and $g^{\text{eq}}(\chi, 0, \rho)$ is given by [Equation 3-1](#). α is determined by setting the total outgoing flux, J , to a fixed value, giving:

$$\alpha = \frac{J}{\int_{\chi \cdot \mathbf{n} < 0} (-\chi \cdot \mathbf{n}) g^{eq}(\chi, 0, \rho) d\chi} \quad (3-5)$$

For a wall boundary condition, the total outgoing flux is usually equated to the total incoming flux, G , given by

$$G = \int_{\chi' \cdot \mathbf{n} > 0} G(\chi', \mathbf{x}, t) d\chi' \quad (3-6)$$

The symbol χ' is used to represent velocity vectors in the incoming half of velocity space. The fluxes $J(\chi, \mathbf{x}, t)$ and $G(\chi', \mathbf{x}, t)$ can be written in terms of the distribution function by considering that the rate of molecules arriving at the surface is related to the normal velocity component of the molecules traveling at velocity χ , as well as the number density of these molecules:

$$\begin{aligned} J(\chi, \mathbf{x}, t) &= (-\chi \cdot \mathbf{n}) g(\chi, \mathbf{x}, t) & (\chi \cdot \mathbf{n} < 0) \\ G(\chi', \mathbf{x}, t) &= (\chi' \cdot \mathbf{n}) g(\chi', \mathbf{x}, t) & (\chi' \cdot \mathbf{n} > 0) \end{aligned} \quad (3-7)$$

Assuming $J = G$ and substituting Equations 3-5, 3-6, and 3-7 into Equation 3-4 gives the result

$$g^*(\chi, \mathbf{x}, t) = \frac{\int_{\chi' \cdot \mathbf{n} > 0} (\chi' \cdot \mathbf{n}) g(\chi', \mathbf{x}, t) d\chi'}{\int_{\chi \cdot \mathbf{n} < 0} (-\chi \cdot \mathbf{n}) g^{eq}(\chi, 0, \rho) d\chi} g^{eq}(\chi, 0, \rho) \quad \chi \cdot \mathbf{n} < 0 \quad (3-8)$$

The asterisk indicates that the variable g^* refers to only the outgoing velocity distribution function. Considering the form of $g^{eq}(\chi, 0, \rho)$ from Equation 3-1 it is clear that the equilibrium function is completely symmetric about the origin, so the following result applies:

$$\int_{\chi \cdot \mathbf{n} < 0} (-\chi \cdot \mathbf{n}) g^{eq}(\chi, 0, \rho) d\chi = \int_{\chi' \cdot \mathbf{n} > 0} |\chi' \cdot \mathbf{n}| g^{eq}(\chi', 0, \rho) d\chi'$$

Equation 3-8 is therefore equivalent to the result derived in Ref. 3 (note that the COMSOL Multiphysics convention of \mathbf{n} represents the outward normal of the domain, whilst in Ref. 3 \mathbf{n} is defined as the inward normal).

Considering Equation 3-1 again, it is clear that $g^{\text{eq}}(\chi, 0, \rho) = f^{\text{eq}}(\xi, \mathbf{u}, \rho)$ so Equation 3-5 can be transformed back to the stationary reference frame in the following manner:

$$f^*(\xi, \mathbf{x}, t) = \frac{J}{\int_{(\xi - \mathbf{u}_w) \cdot \mathbf{n} < 0} (-(\xi - \mathbf{u}_w) \cdot \mathbf{n}) f^{\text{eq}}(\xi, \mathbf{u}_w, \rho, T_w) d\xi} f^{\text{eq}}(\xi, \mathbf{u}_w, \rho) \quad (\xi - \mathbf{u}_w) \cdot \mathbf{n} < 0$$

Similarly Equation 3-6 becomes

$$G = \int_{(\xi' - \mathbf{u}_w) \cdot \mathbf{n} > 0} ((\xi' - \mathbf{u}_w) \cdot \mathbf{n}) f(\xi', \mathbf{x}, t) d\xi'$$

The boundary condition can be straightforwardly converted to the discretized velocity set used in the Transitional Flow interface:

$$f_i^* = \frac{J}{\sum_{(\xi - \mathbf{u}_w) \cdot \mathbf{n} < 0} (-(\xi - \mathbf{u}_w) \cdot \mathbf{n}) f_i^{\text{eq}}} f_i^{\text{eq}} \quad (\xi - \mathbf{u}_w) \cdot \mathbf{n} < 0 \quad (3-9)$$

$$G = \sum_{(\xi_i - \mathbf{u}_w) \cdot \mathbf{n} > 0} ((\xi_i - \mathbf{u}_w) \cdot \mathbf{n}) f_i \quad (3-10)$$

The validity of the discretization form was proved by Ansumali and Karlin (Ref. 3).



The outgoing number densities in Equation 3-9 contain f_i^{eq} terms in both the numerator and the denominator. Inspection of Equation 3-3 shows that all these terms are premultiplied by the equilibrium number density, n_{eq} . In COMSOL Multiphysics, these number density factors are left out of the equilibrium functions, except where they are specifically required, which results in considerable performance improvements for the physics interface.

Equation 3-9 and Equation 3-10 form the basis of all the boundary conditions used in the Transitional Flow interface. For wall boundary conditions at all Knudsen numbers the relation $J = G$ is used to define the boundary condition. In the molecular flow limit J is usually defined on the basis of kinetic theory expressions for the effusive molecular flux, with $\mathbf{u}_w = \mathbf{0}$. In the Navier–Stokes limit a fluid velocity boundary condition can

be implemented by setting $\mathbf{u}_w = \mathbf{u}_{\text{fluid}}$ and $J = G$ ($\mathbf{u}_{\text{fluid}}$ is the user-defined fluid velocity). The pressure boundary condition in the Navier–Stokes limit is more complex, and is described next.

PRESSURE BOUNDARY CONDITION FOR NAVIER–STOKES LIMIT

Specifying a pressure in the fluid in the Navier–Stokes limit is in general difficult with a kinetic equation. In the molecular flow limit the kinetic theory expressions for the flux due to molecular effusion from a reservoir at constant pressure can be employed. In this case the flux enters the domain with a stationary Maxwellian. In the Navier–Stokes limit the incoming flux is harder to specify because the molecules entering the domain do so with a Maxwellian that is drifting at the velocity of the fluid. Pressure boundary conditions based on solutions of the Boltzmann equation within Grad’s 13 moment system have been developed (Ref. 2 and Ref. 6), but these are complex. A simpler approach is used in the Transitional Flow interface, which produces good results when compared with Navier–Stokes fluid flow benchmarks.

In Equation 3-9 there are two unknowns: the emitted flux and the velocity of the Maxwellian. To make the incoming molecules enter with a velocity distribution determined by the fluid flow in the Navier–Stokes limit, $\mathbf{u}_w = \mathbf{u}$ is set, where \mathbf{u} is the local fluid velocity (a function of both the outgoing and incoming distribution functions; note that at this stage the outgoing distribution functions are not determined). The amount of gas entering the fluid flow domain is not straightforward to determine, but it is possible to adjust the flux entering the domain continuously to ensure that the pressure remains constant. In COMSOL Multiphysics this is done by means of a weak form equation for the emitted flux, defined on the boundary. This approach ensures that the pressure is fixed on the boundary whilst allowing the velocity to be determined by the overall fluid flow. This is the desired behavior for a pressure boundary condition in the Navier–Stokes limit. Note that for converged solutions it is observed that the computed emitted flux is equal to the incoming flux, which justifies the use of the moving wall boundary condition in this manner.

Velocity Quadratures

The velocity quadrature used in a simulation is a critical factor to determine the accuracy of the results obtained. With the form of the equilibrium function defined above, it is necessary to use a Gauss–Hermite quadrature for the velocity lattice, with the appropriate basis function. This guarantees that summing over the velocity basis, to compute quantities such as the density, results in the most accurate value for the

quantity of interest when compared to an integral over the continuum velocity space (which is what the lattice Boltzmann method approximates).

COMSOL Multiphysics uses velocity quadratures based on the formalism described in [Ref. 1](#). Consider first the generation of a one-dimensional quadrature. The p th order Hermite polynomial is generated from the function $w(\xi)=\exp(-\xi^2/2)/\sqrt{(2\pi)}$ by the equation

$$H^{(p)}(\xi) = \frac{(-1)^p}{w(\xi)} \frac{d^p}{d\xi^p} (w(\xi))$$

To generate a one-dimensional quadrature of order p , the roots of the equation $H^{(p)}(\xi)=0$ need to be evaluated. These roots represent the velocity coordinates of the quadrature. The p th order polynomial, $H^{(p)}(\xi)$, has p roots: ξ_q , where $q = 1, \dots, p$. To compute the weights associated with the quadrature the following formula is used:

$$w_q = \frac{p!}{(pH^{(p-1)}(\xi_q))^2}$$

Two-dimensional and three-dimensional quadratures can be generated from a one-dimensional quadrature through the product rule. The product rule constructs a two-dimensional quadrature by placing the one-dimensional quadrature along both the x - and y -axes and generating a grid that passes through all the points on the axes. The points at the intersection of the grid lines represent the velocity coordinates of the two-dimensional quadrature. The weights corresponding to each velocity coordinate are also multiplied together. Three-dimensional quadratures are generated in an analogous manner.

The following quadratures have been generated as described above using the product rule: D2Q9, D2Q16, D2Q25, D2Q36, D2Q64, D2Q100, D3Q27, D3Q64, D3Q216, and D3Q512. COMSOL Multiphysics also includes some lower-order quadratures that are not generated directly from the product rule. These are listed in [Table 3-1](#).



A correct transitional flow solution should be independent of the quadrature when the number of velocity directions is increased. It is therefore necessary to demonstrate that solutions are both quadrature and mesh independent as part of the validation procedure.

TABLE 3-1: TWO-DIMENSIONAL VELOCITY LATTICES AVAILABLE IN THE TRANSITIONAL FLOW INTERFACE.

QUADRATURE	ξ_i/c_s	w	NOTES
D2Q12	$\sqrt{6}(\cos(\frac{n\pi}{2}), \sin(\frac{n\pi}{2}))$	1/36	$n = 0, 1, 2, 3$
D2Q12 (continued)	$r(\cos(\frac{m\pi}{4}), \sin(\frac{m\pi}{4}))$	$\frac{5+2\sqrt{5}}{45}$	$m = 1, 3, 5, 7$ $r^2 = \frac{9-3\sqrt{5}}{2}$
D2Q12 (continued)	$s(\cos(\frac{m\pi}{4}), \sin(\frac{m\pi}{4}))$	$\frac{5-2\sqrt{5}}{45}$	$m = 1, 3, 5, 7$ $s^2 = \frac{9+3\sqrt{5}}{2}$



Note that ξ_g is the velocity associated with the given lattice point (c_s is the characteristic speed) whilst w is the corresponding weight.

References for the Transitional Flow Interface

1. X. Shan, X.-F. Yuan, and H. Chen, “Kinetic Theory Representation of Hydrodynamics: A Way Beyond the Navier–Stokes Equations,” *J. Fluid Mech.*, vol. 550, pp. 413–441, 2006.
2. H. Grad, “On the Kinetic Theory of Rarefied Gases”, *Communications on Pure and Applied Mathematics*, vol. 2, pp. 331–407, 1949.
3. S. Ansumali and I.V. Karlin, “Kinetic Boundary Conditions in the Lattice Boltzmann Method,” *Phys. Rev. E*, vol. 66, p. 026311, 2002.
4. J.C. Maxwell, “On Stresses in Rarefied Gases Arising from Inequalities of Temperature”, *Philosophical Transactions of the Royal Society of London*, vol. 170, pp. 231–256, 1879.
5. E.H. Kennard, *Kinetic Theory of Gases*, McGraw Hill, New York, 1938.
6. S.S. Chikatamarla, S Ansumali, and I V Karlin. “Grad’s Approximation for Missing Data in Lattice Boltzmann Simulations,” *Europhys. Lett.*, vol. 74, no. 2, pp. 215–221, 2006.

The Free Molecular Flow Interface

In the free molecular flow regime, the mean free path of a gas is larger than the dimensions of the object being considered. Equivalently the Knudsen number is much greater than 1. Gas molecules therefore interact with surfaces more frequently than they interact with one another and the flow of gas is determined by collisions with the surfaces in the system.

The **Free Molecular Flow (fmf)** interface () found under the **Rarefied Flow** branch () is intended for molecular flows interacting with objects that are moving slowly with respect to the speed of the molecules, such as vacuum systems. Diffuse reflection from all surfaces is assumed (this is reasonable in the majority of practical situations) with molecules from all directions effectively adsorbed onto the surface and subsequently reemitted according to Knudsen's law (that is, with an intensity that varies as the cosine of the angle of emission to the normal to the surface).

When this physics interface is added, these default nodes are also added to the **Model Builder: Molecular Flow, Wall**, and **Initial Values**. Then, from the **Physics** toolbar, add other nodes that implement, for example, boundary conditions and volume forces. You can also right-click **Free Molecular Flow** to add physics from the context menu.



For this physics interface the transient solver computes a quasistatic flow. The assumption is made that all fluxes in the model change much more slowly than the transit time of the molecules across the geometry. At every time step the solver assumes that flux emitted from a source boundary arrives instantaneously at a destination boundary.

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 `fmf`.

INTEGRATION SETTINGS

Select an **Integration method** — **Hemicube** (the default) or **Direct area integration**. These methods define how the incoming molecular flux, G , is computed at each element.

Hemicube

Hemicube is the default. The more sophisticated and general hemicube method uses a z -buffered projection on the sides of a hemicube (with generalizations to 2D and 1D) to account for shadowing effects. Think of it as rendering digital images of the geometry in five different directions (in 3D; in 2D only three directions are needed), and counting the pixels in each mesh element to evaluate its view factor.

Its accuracy can be influenced by setting the integration resolution of the virtual snapshots. The number of z -buffer pixels on each side of the 3D hemicube equals the specified resolution squared. Thus the time required to evaluate the flux increases quadratically with resolution. In 2D, the number of z -buffer pixels is proportional to the resolution property, and thus the time is, as well.

For an axisymmetric geometry, G_m and F_{amb} must be evaluated in a corresponding 3D geometry obtained by revolving the 2D boundaries about the axis. COMSOL creates this virtual 3D geometry by revolving the 2D boundary mesh into a 3D mesh. You can control the resolution in the azimuthal direction by setting the number of azimuthal sectors, which is the same as the number of elements to a full revolution. Try to balance this number against the mesh resolution in the rz -plane.

Direct Area Integration



Direct area integration is fast and accurate for simple geometries with no shadowing, but such circumstances rarely apply for free molecular flows. This setting is not recommended.

COMSOL Multiphysics evaluates the integrals directly, without considering which face elements are obstructed by others. This means that shadowing effects (that is, surface elements being obstructed in nonconvex cases) are not taken into account. Elements facing away from each other are, however, excluded from the integrals.

You can control the accuracy by specifying an integration order. Sharp angles and small gaps between surfaces can require a higher integration order for accuracy but also more time to evaluate the flux.

- For **Hemicube** select an **Integration resolution** — **256** is the default.
- For **Direct area integration** select an **Integration order** — **4** is the default.



For 2D axisymmetric components, also select the **Integration sectors**. To compute the integrals in an axisymmetric component COMSOL Multiphysics must construct a virtual geometry in 3 dimensions. The number of sectors in this geometry is determined by this setting.



Integrating the flux over geometries that involve sharp angles or small gaps is difficult and requires a higher integration resolution and more refined elements than similar geometries without these features. In these situations large areas of the surface are present in regions that correspond to a small solid angle, effectively exaggerating errors caused by the pixelation of the geometry. If these physics are not critical to the model it is best practice to remove them by, for example, adding a chamfer to an acute edge.

COMPUTE

In order to solve a free molecular flow it is always necessary to solve for the incident molecular flux, G for each species. The **Pressure**, **Number density**, and **Heat flux** on the surfaces can be optionally computed by performing additional integrations. The **Pressure** and **Number density** are computed by default.

DEPENDENT VARIABLES

Enter the **Number of species**. The default is 1, but it is possible to add more. It is often convenient to rename the species to something more descriptive for cases when multiple species exist, for example, H_2 or SiH_4 .

The **Incident molecular fluxes** G (SI unit: $1/(m^2s)$) dependent variable is always solved for. It represents the incoming flux at the surface (the outgoing flux is available as $f_{mf}.J$). Other dependent variables might not be solved for, depending on the settings in the **Compute** section (see above).




- The **Pressure** p (SI unit: Pa) represents the total normal force acting on the surface per unit area (including contributions from both incoming and outgoing molecules).

- The **Number density** n (SI unit: $1/(m^3)$), gives the total number density in the vicinity of the surface.
- The **Outward heat flux** Q (SI unit: W/m^2) is the heat flux out of the Free Molecular Flow domain as a result of heat adsorbed by the wall as it interacts with the gas.

For each of the dependent variables, the name can be changed in the corresponding field, but the name of fields and dependent variables must be unique within a model.

DISCRETIZATION

Select the element order for the dependent variable. The default is constant, meaning the molecular flux, pressure, and number density will be constant within each mesh element.

	<p>In the <i>COMSOL Multiphysics Reference Manual</i> 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.</p>
	<ul style="list-style-type: none"> • Domain, Boundary, Edge, Point, Pair, and Global Nodes for the Free Molecular Flow Interface • Theory for the Free Molecular Flow Interface
	<p>The following models are found at this Application Library path: Molecular_Flow_Module/Industrial_Applications/</p> <ul style="list-style-type: none"> • <i>Adsorption and Desorption of Water in a Load Lock Vacuum System</i>: application name: water_adsorption_desorption • <i>Differential Pumping</i>: application name: differential_pumping • <i>Evaporator</i>: application name: evaporator • <i>Molecular Flow in an Ion-Implant Vacuum System</i>: application name: ion_implanter • <i>Ultrahigh Vacuum, Chemical Vapor Deposition</i>: application name: uhv_cvd



The following models are found at this Application Library path:

Molecular_Flow_Module/Benchmarks/

- *Molecular Flow Through an S-Bend*: application name: **s_bend_benchmark**
- *Molecular Flow Through a Microcapillary*: application name: **vacuum_capillary**
- *Molecular Flow Through an RF Coupler*: application name: **rf_coupler**
- *Outgassing Pipes*: application name: **outgassing_pipes**
- *Rotating Plate in a Unidirectional Molecular Flow*: application name: **rotating_plate**

Domain, Boundary, Edge, Point, Pair, and Global Nodes for the Free Molecular Flow Interface

The **Free Molecular Flow Interface** has these domain, boundary, edge, point, pair, and 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.

- Diffuse Flux
- Evaporation
- Initial Values
- Molecular Flow
- Number Density Reconstruction
- Plane Symmetry
- Reservoir
- Surface Temperature
- Symmetry
- Total Vacuum
- Vacuum Pump
- Wall



For 2D axisymmetric components, COMSOL Multiphysics takes the axial symmetry boundaries (at $r = 0$) into account and automatically adds an **Axial Symmetry** node to boundaries that represent the symmetry axis.



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.

Number Density Reconstruction

Use the **Number Density Reconstruction** node to compute the number density on interior domains, boundaries, edges, or points at the time of solving the model.



In addition to the Number Density Reconstruction features, the Free Molecular Flow interface also includes an operator that can be used in postprocessing or in the absence of geometry elements to select. The number density operator is called in the following manner:

```
comp1.fmf.nop_name(x,y,z) (3D)
comp1.fmf.nop_name(x,y)   (2D)
comp1.fmf.nop_name(r,z)   (2D axisymmetric)
```

where name is the dependent variable name for a given species. The default is G, so `comp1.fmf.nop_G(x,y,z)` would be the default operator to compute the number density in 3D. If there are multiple components or Free Molecular Flow interfaces in the model, change the component name `comp1` and physics interface name `fmf` accordingly.

Molecular Flow

Use the **Molecular Flow** node to define the mean molar mass. This is required to compute the pressure, number density, and heat flux at the boundaries.

MOLECULAR FLOW

By default the **Mean molar mass** $M_{n,G}$ (SI unit: kg/mol) is **User defined**. For each species, enter a value or expression in the field. The default is 0.028 kg/mol.

Surface Temperature

Use the **Surface Temperature** node to define the surface temperature at the boundary. For nonisothermal models, additional Surface Temperature features can be added to the model to define different temperatures on different surfaces. The temperature can

also be computed by a second coupled physics interface (for example, a Heat Transfer in Solids interface).

SURFACE TEMPERATURE

By default the **Temperature** T (SI unit: K) is **User defined**. The default is 293.15 K. If additional physics interfaces (such as a Heat Transfer in Solids interface) are available which compute the temperature, the output of these physics interfaces can be selected.

Evaporation

Use the **Evaporation** node to define an evaporative source. This feature is usually used in combination with a Surface Temperature node, which sets the temperature of the source.

EVAPORATION

For each species, enter a **Vapor pressure** $p_{\text{vap}, G}$ (SI unit: Pa) for the evaporating material. The default is 0 Pa.

For each species, enter an **Evaporation coefficient** $a_{v, G}$ (dimensionless) for the evaporating material. The default is 1.

Initial Values

The **Initial Values** node adds initial values for the incident molecular flux, number density, and pressure. For the transient solver, these values define the state of the problem at the initial time step; for the stationary solver, they serve as a starting point for the nonlinear solver.



For transient problems it is often difficult to specify consistent initial conditions. It is possible to add a stationary step in the study to serve as the initial condition for the transient problem, or to start the system at total vacuum (all values equal to zero) and to ramp up the pressure or flux at inlets or reservoirs prior to adding time-dependent effects.

INCIDENT MOLECULAR FLUX

For each species, enter the **Incident molecular flux**, G_G (SI unit: $1/(\text{m}^2 \cdot \text{s})$).

PRESSURE

For each species, enter the **Pressure**, p_G (SI unit: Pa).

NUMBER DENSITY

For each species, enter the **Number density**, n_G (SI unit: $1/\text{m}^3$).

Wall

The **Wall** feature provides boundary conditions for modeling different kinds of solid surfaces within vacuum systems.

WALL TYPE

Select **Wall type**: **Wall** (the default), **Outgassing wall**, **Adsorption/desorption**, or **Deposition**. For **Wall** it sets the incoming molecular flux for each species, G , equal to the flux emitted from the wall, J . Therefore all the incident molecules are diffusely reflected.

FLUX

For **Outgassing wall** it allows an additional outgassing flux $J_{0,G}$ to be added to the incident flux, so the total emitted flux for each species is given by $J = G + J_{0,G}$. Select an **Outgoing flux**: **Specify directly** (the default), **Number of SCCM units**, **Mass flux**, **Total mass flow**, or **Thermal desorption rate**.

Specify Directly

For **Specify directly**, and for each species, enter a value or expression for $J_{0,G}$ (SI unit: $1/(\text{m}^2 \cdot \text{s})$).

Number of SCCM (Standard Cubic Centimeters Per Minute) Units

For **Number of SCCM units**, and for each species, enter $Q_{\text{sccm},G}$ (dimensionless). This quantity is automatically converted to a standardized volumetric flow rate in SI units. Select an option from the **Standard flow rate defined by** list — **Standard density** (the default) or **Standard pressure and temperature**.

- For **Standard density** enter a **Standard molar volume** V_m (SI unit: m^3/mol). The default is $0.0224136 \text{ m}^3/\text{mol}$ (in accordance with SEMI standard E12-0303).
- For **Standard pressure and temperature** enter a **Standard pressure** P_{st} (SI unit: Pa) and **Standard temperature** T_{st} (SI unit: K). For 2D components, also enter a **Channel thickness** d_{bc} (SI unit: m).

Mass Flux

For **Mass flux**, and for each species, enter a value or expression for $M_{f,G}$ (SI unit: $\text{kg}/(\text{m}^2 \cdot \text{s})$). The default is $0 \text{ kg}/(\text{m}^2 \cdot \text{s})$.

Total Mass Flow

For **Total mass flow**, and for each species, enter a value or expression for $Q_{m,G}$ (SI unit: kg/s). The default is 0 kg/s. For 2D components, also enter a **Channel thickness** d_{bc} (SI unit: m). This is used to specify the depth of the boundary in the out-of-plane direction.

Thermal Desorption Rate

For **Thermal desorption rate**, and for each species, enter a value or expression for $D_{r,G}$ (SI unit: W/m²). The default is $3e-12$ [torr*1 / (cm²*s)]. The outgassing flux, $J_{0,G}$ is related to the thermal desorption rate via $J_{0,G} = N_A D_{r,G} / RT$, where N_A is Avogadro's number, R is the universal gas constant, and T is the temperature.

ADSORPTION/DESORPTION

For **Adsorption/desorption** it is used in transient models to model adsorption and desorption of gas molecules on the surface. For each of the species enter the following information.

- **Sticking coefficient** S_G (dimensionless). This defines the probability that molecules incident on the surface are adsorbed.
- **Desorption rate** D_G (SI unit: mol/(m²*s)). This defines the rate of desorption. The boundary condition tracks the number of moles of gas adsorbed per unit area of the surface (fmf.n_ads) as a function of time.
- **Initial adsorbent concentration** $n_{ads,0,G}$ (SI unit: mol/m²).
- **Additional molar flux** Γ_G (SI unit: mol/(m²*s)) can be added to the surface, enabling, for example, the modeling of molecules arriving at the surface by diffusion through the walls of the chamber from the outside (these molecules are added to the adsorbed molecules on the surface). Note that molar units are used for the adsorbed molecules, but the total number of molecules per unit area is available as the variable fmf.N_ads.



Solving for the adsorbent concentration can cause convergence issues if the concentration is close to zero, as its value can sometimes go negative during the solution process. If this happens when using this boundary condition, tighten the relative solver tolerance (see the settings for the **Study 1>Time-Dependent Solver** node) to $1e-3$. Then in the settings for the **Study 1>Solver Configurations>Time-Dependent Solver** node change the absolute tolerance to $1e-4$ (if this node is not visible right-click the **Study 1** node and select **Show Default Solver**).



See also [Adsorption and Desorption](#) in the section [Theory for the Free Molecular Flow Interface](#).

DEPOSITION

Deposition is used for simulations in which the molecular flow of a deposited species is tracked. It is assumed that all the molecules arriving at the surface are deposited.

For each species, enter a **Film density** $\rho_{\text{film,G}}$ (SI unit: kg/m^3) and **Initial film thickness** $h_{\text{film,0,G}}$ (SI unit: m) to compute the film thickness (`fmf.h_film`) as a function of time.

Diffuse Flux

The **Diffuse Flux** boundary condition sets the molecular flux emitted from the boundary to a user-specified value. The flux is emitted according to Knudsen's law (the emitted flux in a given direction is proportional to the cosine of its angle to the surface normal). This can be set directly as a flux, as a total incoming mass flow, or number of SCCMs.

DIFFUSE FLUX

Select the **Outgoing flux: Specify directly** (the default), **Number of SCCM units**, **Mass flux**, or **Total mass flow**. The rest of the settings are the same as for [Wall](#), described in the [Flux](#) section.

Total Vacuum

Use the **Total Vacuum** boundary condition to set the flux emitted from the wall, J , equal to zero. This is equivalent to an opening to a large vessel or extended region with a much lower pressure than that in the system.

Reservoir

The **Reservoir** boundary condition computes the incoming molecular flux from a large adjacent vessel or region filled with gas at a specified number density or pressure. In this case the effusing flux obeys Knudsen's law.

The reservoir boundary condition represents an opening to an adjacent chamber whose extent is much larger than the molecular mean free path. The boundary condition assumes that the molecules in the reservoir have a Maxwellian distribution of velocities and that the size of the opening to the reservoir is much less than the mean

free path. Molecules travel through the opening by a process known as molecular effusion. Effusion is a process that selects molecules with higher molecular speeds than average, because they are traveling toward the hole faster than the other molecules. Effusing molecules therefore have a larger average velocity than the molecules in the adjacent reservoir. As a result the pressure or number density at the reservoir boundary can differ from the value set in the boundary condition — which determines the pressure that would be measured within the reservoir away from the opening.

RESERVOIR

Select the **Boundary condition: Reservoir pressure** (the default) or **Reservoir number density**.

- For **Reservoir pressure**, and for each species, enter a value or expression for the **Reservoir pressure** $p_{0,G}$ (SI unit: Pa). The default is 0 Pa.
- For **Reservoir number density**, and for each species, enter the **Reservoir number density** $n_{0,G}$ (SI unit: $1/\text{m}^3$).

Vacuum Pump

Use the **Vacuum Pump** node to represent a vacuum pump such as a turbomolecular pump or a diffusion pump.

VACUUM PUMP

Select how to **Specify pump flux: Fraction of incident molecules** (the default), **Directly**, or **Pump speed**. The emitted flux J is given by $J = G - J_{\text{pump}}$.

The **Fraction of incident molecules** f_G , is specified either directly, or through a **Pump speed**, S_G , such that $f = \sqrt{(2\pi M_n/RT)S/A}$ (here M_n is the molar mass of the species, R is the universal gas constant, T is the temperature, and A is the area of the pump). This condition applies pointwise across the surface of the pump.



Care should be taken with the specification of $J_{\text{pump},G}$ to ensure that the value is never negative.

- For **Directly**, and for each species, enter a value for $J_{\text{pump},G}$ (SI unit: $\text{mol}/(\text{m}^2 \cdot \text{s})$).
- For **Fraction of incident molecules**, and for each species, enter a value for f_G (dimensionless). The default is 0.8.
- For **Pump speed**, and for each species, enter a value or expression for S_G (SI unit: m^3/s). The default is 500 l/s.

For 2D components, also enter a **Channel thickness** d_{bc} (SI unit: m). This specifies the out-of-plane thickness of the pump, which is required in order to compute its area.

Plane Symmetry



The **Plane Symmetry** node is obsolete and will be removed in future versions. Instead, consider using the [Symmetry](#) node, which is capable of handling sector symmetry as well as plane symmetry with multiple perpendicular reflection planes. The **Plane Symmetry** and **Symmetry** features should not be used together in the same model.

Use the **Plane Symmetry** node to specify boundaries which correspond to symmetry; that is, boundaries where the incoming and outgoing fluxes are equal. Note that these boundaries must be planar and cannot be curved.

Symmetry

Use the **Symmetry** node to specify that some exterior boundaries of the model geometry either represent planes of reflection symmetry, where incoming and outgoing fluxes are equal; or the start and end planes for sector symmetry.

Only one **Symmetry** node may be added to each instance of the Free Molecular Flow physics interface. It is not recommended to use the **Symmetry** node together with the obsolete [Plane Symmetry](#) node in the same interface.

When assigning selections for the symmetry planes, it is important that each selection only includes surfaces that are coplanar. For example, you should not add two surfaces to the selection list for **First Reflection Plane** if they are oriented at an angle with each other; even if they are parallel, they should not both be selected if there is some normal displacement between them.

However, it is also important not to miss any surfaces that do lie in the same symmetry plane, since adding boundaries to a selection list will override other boundary conditions, such as the default [Wall](#), on those same boundaries.

For all of the supported symmetry types, the selected boundaries must be planar and cannot be curved. Only boundaries on the exterior of the physics interface domain selection may be selected.

SYMMETRY

Select an option from the **Type of symmetry** list. The default is **One symmetry plane**. The option selected from this list changes which selection inputs are displayed in the settings window. For 2D axisymmetric models, only the **One symmetry plane** option is supported, and the selected boundaries should always be perpendicular to the z-axis.

- For **One symmetry plane**, only the selection input for the **First Reflection Plane** is shown.
- For **Two perpendicular symmetry planes**, the selection inputs for the **First Reflection Plane** and **Second Reflection Plane** are shown. The boundaries selected for both reflection planes should be perpendicular to each other.
- For **Three perpendicular symmetry planes** (3D only), the selection inputs for the **First Reflection Plane**, **Second Reflection Plane**, and **Third Reflection Plane** are shown. The boundaries selected for all three reflection planes should be mutually perpendicular.
- For **Sector symmetry**, the selection inputs for the **Sector Start Plane** and **Sector End Plane** are shown. The line where the two sector planes meet is the axis of rotation for n -fold rotation symmetry, where n is the **Number of sectors** specified in the **Sector Symmetry** section.
- For **Sector symmetry**, the selection inputs for the **First Reflection Plane** and **Sector Start Plane** are shown. The sector end plane is assumed to be the mirror image of the sector start plane across the reflection plane; that is, the sector itself has plane symmetry. The line where the planes meet is the axis of rotation for n -fold rotation symmetry, where n is the **Number of sectors** specified in the **Sector Symmetry** section.

SECTOR SYMMETRY

This section is shown if **Sector symmetry** or **Sector symmetry with reflection** is selected from the **Type of symmetry** list in the **Symmetry** section. Enter a value for the **Number of sectors** (dimensionless). The default is 4. This should be an integer representing the number of sectors that constitute a full revolution around the symmetry axis. For **Sector symmetry** with reflection this should be the number of full sectors after applying the reflection, not the number of half sectors.

SKETCH

This section includes some illustrations of the different types of symmetry that are supported. The sketch will change depending on space dimension and on the option selected from the **Type of symmetry** list in the **Symmetry** section.

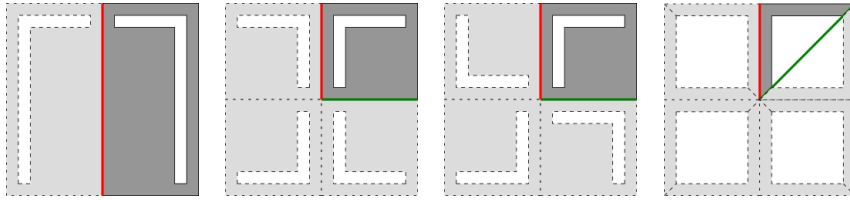


Figure 3-1: Sketches for symmetry in 2D. From left to right: single reflection plane, two reflection planes, sector symmetry, and sector symmetry with a reflection.

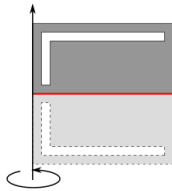


Figure 3-2: Sketch for a single symmetry plane in a 2D axisymmetric geometry.

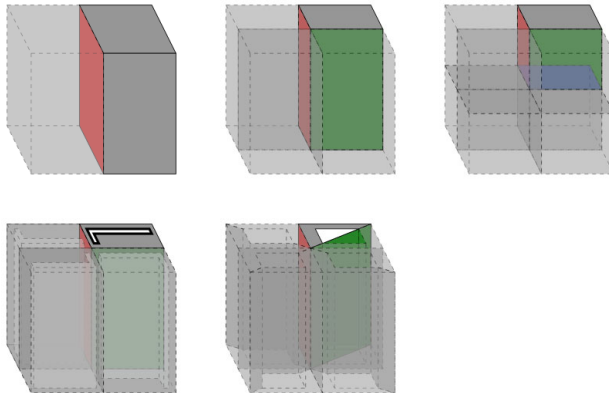


Figure 3-3: Sketches for symmetry in 3D. Top row: one, two, and three reflection planes. Bottom row: sector symmetry and sector symmetry with reflection.

Theory for the Free Molecular Flow Interface

The [Free Molecular Flow Interface](#) theory is described in this section:

- [Solving the Flow](#)
- [Calculating the Particle Flux](#)
- [Calculating the Pressure](#)
- [Calculating the Number Density](#)
- [Calculating the Heat Flux](#)
- [About Boundary Conditions for Pressure and Number Density](#)
- [Adsorption and Desorption](#)
- [Evaporation](#)
- [References for the Free Molecular Flow Interface](#)

Solving the Flow

For the Free Molecular Flow interface when the Knudsen number is much greater than 1, collisions between gas molecules as they traverse the interior of a system can be ignored. There are two common approaches to solving the flow in this case: the *Monte Carlo method* (which computes the trajectories of large numbers of randomized particles through the system) and the *angular coefficient method*. COMSOL Multiphysics uses the angular coefficient method, which computes the molecular flow by summing the flux arriving at a surface from all other surfaces in its line of sight. The macroscopic variables in the vicinity of the surface can be derived from kinetic theory.



Although COMSOL Multiphysics currently does not offer a physics interface for Monte Carlo modeling of Molecular Flows, it is possible to use the Particle Tracing interface for limited Monte-Carlo computations (this requires the Particle Tracing Module). The *Molecular Flow Through an S-Bend* shows how to do this.

In many cases it is reasonable to assume that molecules are adsorbed and subsequently diffusely emitted from the surface (this is often referred to as total accommodation).

In this instance, the appropriate probability density functions for emission from a surface $\rho(\theta, c)$ (SI unit: m^{-1}s), where θ is the angle to the normal (SI unit: rad) and c is the particle speed (SI unit: ms^{-1}), have been derived from the laws of physical gas dynamics and verified by molecular dynamics simulations (Ref. 1). A complete derivation of the probability distribution function in 2D is given in this section.

THE COSINE LAW

The distribution of angles of reflection follows Knudsen's cosine law,

$$f(\theta)d\theta = \frac{1}{2}\cos\theta d\theta \quad f(\sin\theta)d\sin\theta = \frac{1}{2}d\sin\theta \quad (3-11)$$

for θ in the range $(-\pi/2, \pi/2)$ for both cylindrical (2D) and polar (3D) coordinates. In 3D the azimuthal angle ϕ is in the range $(0, \pi)$.

THE MAXWELL-BOLTZMANN DISTRIBUTION

In free space, the gas molecule velocity components follow the Maxwell-Boltzmann distribution,

$$\rho(c_i) = \sqrt{\frac{m}{2\pi k_B T}} \exp\left(-\frac{mc_i^2}{2k_B T}\right)$$

where

- c_i is the velocity component in the i direction ($-\infty < c_i < \infty$),
- $i \in \{x, y\}$ in 2D or $i \in \{x, y, z\}$ in 3D,
- $k_B = 1.3806488 \times 10^{-23}$ J/K is the Boltzmann constant, and
- T is the temperature (SI unit: K).

PROBABILITY DISTRIBUTION FUNCTION IN 2D

Let c_x and c_y be the velocity components in the parallel and wall normal directions, respectively. The relationships between the velocity components in Cartesian and polar coordinates are

$$\begin{aligned} c_x &= c \sin\theta \\ c_y &= c \cos\theta \end{aligned}$$

Due to symmetry considerations, the component of the reflected molecule velocity parallel to the wall follows a Gaussian distribution,

$$\rho_x(c_x) = \sqrt{\frac{m}{2\pi k_B T}} \exp\left(-\frac{mc_x^2}{2k_B T}\right) \quad (3-12)$$

However, as shown in the following derivation, the interactions with the surface in the normal direction are not symmetric and do not follow a Gaussian distribution, so the probability distribution function of the molecule speed c differs from the usual 2D Maxwell distribution.

The Jacobian of transformation between Cartesian and polar coordinates in 2D is

$$\left| \frac{\partial(c_x, c_y)}{\partial(c, \sin\theta)} \right| = \frac{c}{\cos\theta} \quad (3-13)$$

and therefore

$$\rho(c, \sin\theta) = \frac{c}{\cos\theta} \rho(c_x, c_y)$$

The two random variables for the velocity components are assumed to be independent, so that the probability distribution functions can be expressed in the form

$$\begin{aligned} \rho(c_x, c_y) &= \rho_x(c_x) \rho_y(c_y) \\ \rho(c, \sin\theta) &= \rho_c(c) \rho_\theta(\sin\theta) \end{aligned}$$

The normalization conditions on the probability distribution functions are thus

$$\int_{-1}^1 \int_0^\infty \rho_c(c) \rho_\theta(\sin\theta) dc d(\sin\theta) = 1$$

Applying [Equation 3-13](#) to the normalization condition yields

$$\int_{-1}^1 \int_0^\infty \frac{c}{\cos\theta} \rho_x(c \sin\theta) \rho_y(c \cos\theta) dc d(\sin\theta) = 1$$

Substituting [Equation 3-12](#) into the modified normalization condition then yields

$$\int_{-1}^1 \int_0^\infty \frac{c}{\cos\theta} \rho(c \cos\theta) \sqrt{\frac{m}{2\pi k_B T}} \exp\left(-\frac{mc^2 \sin^2\theta}{2k_B T}\right) dc d(\sin\theta) = 1$$

The functional form of $\rho(c \cos \theta)$ that satisfies both the normalization requirement and the cosine law (Equation 3-11) is

$$\rho(c \cos \theta) = \frac{mc \cos \theta}{k_B T} \exp\left(-\frac{mc^2 \cos^2 \theta}{2k_B T}\right)$$

or, in Cartesian coordinates,

$$\rho_y(c_y) = \frac{mc_y}{k_B T} \exp\left(-\frac{mc_y^2}{2k_B T}\right) \quad (3-14)$$

Combining Equation 3-12 and Equation 3-14 yields the probability distribution function of the speed c ,

$$\rho_{2D}(c) = \sqrt{\frac{2}{\pi}} \left(\frac{m}{k_B T}\right)^{3/2} c^2 \exp\left(-\frac{mc^2}{2k_B T}\right)$$

which is noticeably different from the Maxwell distribution in 2D, as shown in Figure 3-4.

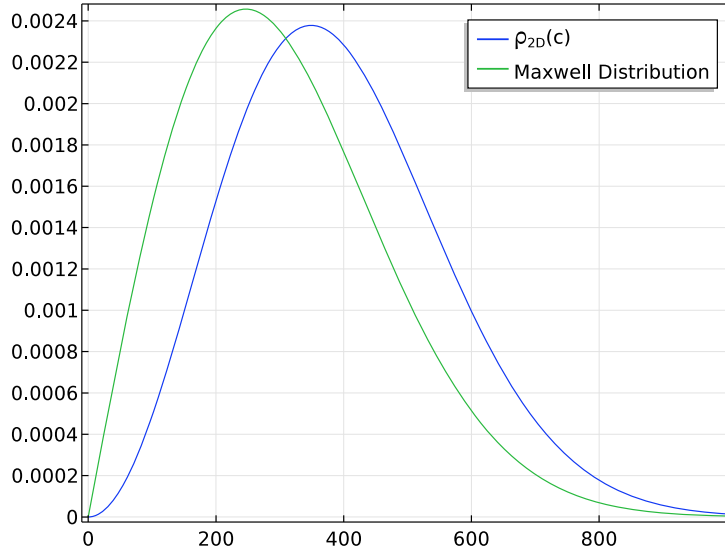


Figure 3-4: Comparison of the speed distribution of diffusely emitted particles in 2D to the standard Maxwell distribution.

SUMMARY OF PROBABILITY DISTRIBUTION FUNCTIONS

Repeating the derivation of the previous section in 3D yields an analogous probability distribution function; as in 2D, the result is found to differ from the distribution function of c that would be obtained naively from the Maxwell–Boltzmann distribution. The results in both 2D and 3D are summarized below. The polar angle distribution always follows the cosine law,

$$\begin{aligned}\rho_{2D}(\theta) &= \frac{\cos\theta}{2} \\ \rho_{3D}(\theta) &= \frac{\cos\theta}{\pi}\end{aligned}$$

The molecule speed follows one of the distributions

$$\begin{aligned}\rho_{2D}(c) &= \sqrt{\frac{2}{\pi}} \left(\frac{m}{k_B T}\right)^{3/2} c^2 \exp\left(-\frac{mc^2}{2k_B T}\right) \\ \rho_{3D}(c) &= \left(\frac{m}{k_B T}\right)^2 \frac{c^3}{2} \exp\left(-\frac{mc^2}{2k_B T}\right)\end{aligned}\tag{3-15}$$

based on space dimension. In 3D, the probability distribution function of the azimuthal angle ϕ is uniform over the interval $(0, \pi)$.

Calculating the Particle Flux

The flux of molecules from x' arriving at point x as shown in [Figure 3-5](#) is derived.

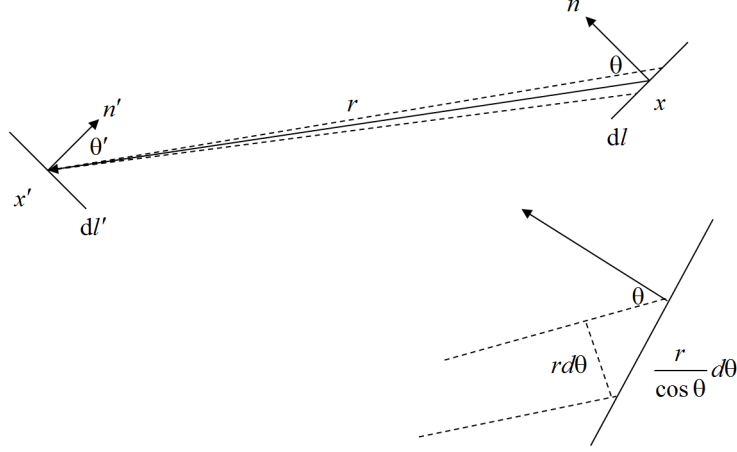


Figure 3-5: Diagram showing the contribution of the flux arriving at point x from point x' in 2 dimensions.

Let the total rate of emission of molecules per unit area at x' be J' (SI unit: $1/(\text{m}^2 \cdot \text{s})$). In the steady state J' is independent of time. Note that in practice, the molecules take a finite time to travel from x' to x , but this is not accounted for in the calculations (which are consequently quasistatic).

In 2D the flux of molecules with speeds between c' and $c' + dc'$ leaving x' with angles between θ' and $\theta' + d\theta'$ is

$$J' dl' \rho_{2D}(\theta') d\theta' \cdot \rho_{2D}(c') dc'$$

In 3D the flux of molecules with speeds between c' and $c' + dc'$ leaving x' with angles between θ' and $\theta' + d\theta'$ and ϕ' and $\phi' + d\phi'$ is

$$J' dS' \rho_{3D}(\theta') \sin|\theta'| d\theta' d\phi' \cdot \rho_{3D}(c') dc'$$

In 2D, molecules leaving x' at angles between θ' and $\theta' + d\theta'$ is spread over an area of $r d\theta'$ in the plane with normal parallel to r at x . This plane is in turn projected onto an area

$$(r d\theta') / (\cos \theta)$$

on the surface at x (see Figure 3-5). In 3D, molecules leaving x' at angles between θ' and $\theta' + d\theta'$ and ϕ' and $\phi' + d\phi'$ is spread over an area of

$$r^2 \sin|\theta'| d\theta' d\phi'$$

in the plane with normal parallel to r at x giving a projected area of

$$(r^2 \sin|\theta'| d\theta' d\phi') / (\cos \theta)$$

The arriving flux per unit area at x from x' , dG ($\text{m}^{-2} \cdot \text{s}^{-1}$), is therefore:

$$\begin{aligned} dG_{2D} &= \frac{J dl' \cos \theta \rho_{2D}(\theta') \rho_{2D}(c') dc'}{r} \\ dG_{3D} &= \frac{J dS' \cos \theta \rho_{3D}(\theta') \rho_{3D}(c') dc'}{r^2} \end{aligned} \quad (3-16)$$

The total flux is therefore given by:

$$\begin{aligned} G_{2D} &= \int_{l'} \frac{J \cos \theta \rho_{2D}(\theta')}{r} dl' \int_0^\infty \rho_{2D}(c') dc' \\ G_{3D} &= \int_{S'} \frac{J \cos \theta \rho_{3D}(\theta')}{r^2} dS' \int_0^\infty \rho_{3D}(c') dc' \end{aligned}$$

where the integral is over the path l' , or the surface S' that includes all the surfaces in the line of sight of x .

By the definition of the probability distribution, both the integrals over c' evaluate to 1. Substituting for $\rho(\theta')$ the following equations are left:

$$\begin{aligned} G_{2D} &= \int_{l'} \frac{J \cos \theta \cos \theta'}{2r} dl' = - \int_{l'} \frac{J'(\mathbf{n} \cdot \mathbf{r})(\mathbf{n}' \cdot \mathbf{r})}{2r^3} dl' \\ G_{3D} &= \int_{S'} \frac{J \cos \theta \cos \theta'}{\pi r^2} dS' = - \int_{S'} \frac{J'(\mathbf{n} \cdot \mathbf{r})(\mathbf{n}' \cdot \mathbf{r})}{\pi r^4} dS' \end{aligned} \quad (3-17)$$

In the Free Molecular Flow interface, COMSOL Multiphysics computes the incident flux on the surfaces G as the dependent variable. The outgoing flux from a surface (usually specified in terms of G by the boundary condition) is also available as `fmf.J`.

Calculating the Pressure

The pressure acting on the surface at x can be calculated from the rate of change of momentum of molecules normal to the surface. For incoming molecules the molecules from x' give the following contribution to the pressure on the surface:

$$dp_{2D}^{\text{in}} = \frac{J' dl' \rho_{2D}(\theta') m \cos^2 \theta' \rho_{2D}(c') dc'}{r}$$

$$dp_{3D}^{\text{in}} = \frac{J' dS' \rho_{3D}(\theta') m \cos^2 \theta' \rho_{3D}(c') dc'}{r^2}$$

Substituting for ρ (θ) and integrating gives the total contribution of incoming molecules to the pressure, p^{in} (SI unit: Pa), at x :

$$p_{2D}^{\text{in}} = m \int_{l'} \frac{J' \cos \theta' \cos^2 \theta}{2r} \int_0^\infty c \rho_{2D}(c') dc' dl' = - \int_{l'} \frac{J' (\mathbf{n} \cdot \mathbf{r})^2 (\mathbf{n}' \cdot \mathbf{r})}{2r^4} m \langle c'_{2D} \rangle dl'$$

$$p_{3D}^{\text{in}} = m \int_{S'} \frac{J' \cos \theta' \cos^2 \theta}{\pi r^2} \int_0^\infty c \rho_{2D}(c') dc' dS' = - \int_{S'} \frac{J' (\mathbf{n} \cdot \mathbf{r})^2 (\mathbf{n}' \cdot \mathbf{r})}{\pi r^5} m \langle c'_{3D} \rangle dS'$$

From [Equation 3-15](#) it is shown that:

$$\langle c'_{2D} \rangle = \sqrt{\frac{8k_B T}{\pi m}}$$

$$\langle c'_{3D} \rangle = \sqrt{\frac{9\pi k_B T}{8m}}$$
(3-18)

The computed incoming pressure contribution is available on surfaces as `fmf.Pin`.

The surface is also emitting molecules at a rate J (in most cases $J = G$, but in general this does not need to be the case — for example, the surface can be outgassing in addition to emitting the adsorbed molecules so that $J = G + J_{\text{out}}$). The emitted molecules also have a normal momentum change that contributes to the pressure in the following manner:

$$dp_{2D}^{\text{out}} = J m c \cos \theta \rho_{2D}(\theta) \rho_{2D}(c) d\theta dc$$

$$dp_{3D}^{\text{out}} = J m c \cos \theta \rho_{3D}(\theta) \rho_{3D}(c) \sin |\theta| d\theta dc$$

Thus the total pressure contribution from emitted molecules, p^{out} (SI unit: Pa), is:

$$p_{2D}^{\text{out}} = Jm \int_{-\pi/2}^{\pi/2} \frac{\cos^2 \theta}{2} d\theta \int_0^{\infty} c \rho_{2D}(c) dc = \frac{\pi}{4} Jm \langle c_{2D} \rangle$$

$$p_{3D}^{\text{out}} = Jm \int_{-\pi/2}^{\pi/2} \frac{\cos^2 \theta \sin |\theta|}{\pi} d\theta \int_0^{\pi} 1 d\phi \left(\int_0^{\infty} c \rho_{3D}(c) dc \right) = \frac{2}{3} Jm \langle c_{3D} \rangle$$

Substituting for the mean velocity (using [Equation 3-18](#) with $T' \rightarrow T$) the equation obtained:

$$p^{\text{out}} = \left(\frac{\pi k_B T m}{2} \right)^{1/2} J \quad (3-19)$$

where the same result holds in 2D and 3D. The outgoing pressure contribution is available in the Free Molecular Flow interface as `fmf.Pout`.

Finally the total pressure acting on the surface, p (SI unit: Pa), is given by:

$$p = p_{\text{in}} + p_{\text{out}}$$

This is computed in the physics interface as the dependent variable p .

Calculating the Number Density

The number density at the surface can also be calculated from a similar argument. Molecules arriving at x from x' with speeds between c and $c + dc$ make a contribution, dn , to the number density, n (SI unit: $1/\text{m}^3$), at the surface that is related to their flux at the surface, dG , by:

$$dn = \frac{dG}{c' \cos \theta}$$

Following the same argument, the total incoming number density at x is given by:

$$n_{2D}^{\text{in}} = \int_{l'} \frac{J' \cos \theta'}{2r} \int_0^{\infty} \frac{\rho_{2D}(c')}{c} dc' dl' = - \int_{l'} \frac{J'(\mathbf{n}' \cdot \mathbf{r})}{2r^2} \left\langle \frac{1}{c'_{2D}} \right\rangle dl' \quad (3-20)$$

$$n_{3D}^{\text{in}} = \int_{S'} \frac{J' \cos \theta'}{\pi r^2} \int_0^{\infty} \frac{\rho_{3D}(c')}{c} dc' dS' = - \int_{S'} \frac{J'(\mathbf{n}' \cdot \mathbf{r})}{\pi r^3} \left\langle \frac{1}{c'_{3D}} \right\rangle dS'$$

The means of inverse speed (calculated using [Equation 3-15](#)) are given by:

$$\begin{aligned}\left\langle \frac{1}{c'_{2D}} \right\rangle &= \sqrt{\frac{2m}{\pi k_B T}} \\ \left\langle \frac{1}{c'_{3D}} \right\rangle &= \sqrt{\frac{\pi m}{8k_B T}}\end{aligned}\tag{3-21}$$



The incoming number density is available as `fmf.Nin`.

Note that the integrand in [Equation 3-20](#) depends only on the emitted flux, the normal, and the molecular velocity at the emitting surface (and not on the orientation or properties of the wall where the flux is arriving). [Equation 3-20](#) can therefore be used to compute the flux arriving at any point within the flow domain, in addition to the flux arriving at surfaces. This is the basis of the number density reconstruction features and the number density calculation operator, which can be used to calculate the number density within the domain. See [Number Density Reconstruction](#) in the previous section for further details on how to use these features.

Considering again the case of a surface in the model, the contribution to the outgoing number density at a given angle and speed the from the emitted molecules is:

$$\begin{aligned}dn_{2D}^{\text{out}} &= \frac{J\rho_{2D}(\theta)\rho_{2D}(c)}{c\cos\theta}d\theta dc \\ dn_{3D}^{\text{out}} &= \frac{J\rho_{3D}(\theta)\rho_{3D}(c)\sin|\theta|}{c\cos\theta}d\phi d\theta dc\end{aligned}$$

so the total outgoing number density is given by:

$$\begin{aligned}n_{2D}^{\text{out}} &= J \int_{-\pi/2}^{\pi/2} \frac{1}{2} d\theta \int_0^{\infty} \frac{\rho_{2D}(c)}{c} dc = \frac{\pi}{2} J \left\langle \frac{1}{c_{2D}} \right\rangle \\ n_{3D}^{\text{out}} &= J \int_{-\pi/2}^{\pi/2} \frac{\sin|\theta|}{\pi} d\theta \int_0^{\pi} 1 d\phi \int_0^{\infty} \frac{\rho_{3D}(c)}{c} dc = 2J \left\langle \frac{1}{c_{3D}} \right\rangle\end{aligned}$$

Substituting for the mean inverse velocities (using [Equation 3-21](#) with $T' \rightarrow T$) the equation obtained:

$$n^{\text{out}} = \left(\frac{\pi m}{2k_B T} \right)^{1/2} J$$

where the same result holds in 2D and 3D. The outgoing number density is available in the physics interface as `fmf.Nout`.

The total number density of the molecules near the surface is therefore:

$$n = n^{\text{in}} + n^{\text{out}}$$

This is computed in the physics interface as the dependent variable n .

Calculating the Heat Flux

To compute the incident heat flux on a surface it is necessary to sum the kinetic energy of the incident molecules. On average each molecule carries kinetic energy, q_m , given by:

$$q_m = \frac{1}{2}m \langle c'^2 \rangle$$

Equation 3-15 can be used to shown that:

$$\begin{aligned} \langle c'^2_{2D} \rangle &= \frac{3k_B T}{m} \\ \langle c'^2_{3D} \rangle &= \frac{4k_B T}{m} \end{aligned} \tag{3-22}$$

The total kinetic energy arriving at the surface is calculated by multiplying the total flux arriving from a given direction, multiplying this quantity by Equation 3-22, and then integrating over all directions. From Equation 3-17 the appropriate integral is given by:

$$\begin{aligned} Q_{2D}^{\text{in}} &= - \int_{l'} \frac{3k_B T}{2m} \frac{J'(\mathbf{n} \cdot \mathbf{r})(\mathbf{n}' \cdot \mathbf{r})}{2r^3} dl' \\ Q_{3D}^{\text{in}} &= - \int_{S'} \frac{2k_B T}{m} \frac{J'(\mathbf{n} \cdot \mathbf{r})(\mathbf{n}' \cdot \mathbf{r})}{\pi r^4} dS' \end{aligned}$$



The incoming heat flux is available as `fmf.Qin`.

The outgoing heat flux is simply:

$$Q_{2D}^{\text{out}} = \frac{1}{2} m \langle c_{2D}^2 \rangle J = \frac{3}{2} k_B T J$$

$$Q_{3D}^{\text{out}} = \frac{1}{2} m \langle c_{3D}^2 \rangle J = 2 k_B T J$$

The net heat flux removed from the gas is given by:

$$Q^{\text{tot}} = Q^{\text{int}} - Q^{\text{out}}$$

This is the heat flux that flows *into* the solid walls of the vacuum chamber.



The net heat flux is available as `fmf.Qtot`.

About Boundary Conditions for Pressure and Number Density

Consider the molecules effused from an infinite half space filled with a gas of number density n through a small hole in the half space with diameter much less than the mean free path of the gas.

Within the gas the molecules are traveling in all directions on a unit sphere with equal probability. The probability of molecules striking the hole at angles between θ and $\theta + d\theta$ to the normal of the bounding surface of the half space is proportional to the fraction of solid angle (or angle in 2D) that this angular range occupies. In 3D, as shown in [Figure 3-6](#), the total solid angle corresponding to angles between θ and $\theta + d\theta$ and between ϕ and $\phi + d\phi$ is $\sin \theta d\theta d\phi$. In 2D the angle is simply $d\theta$.

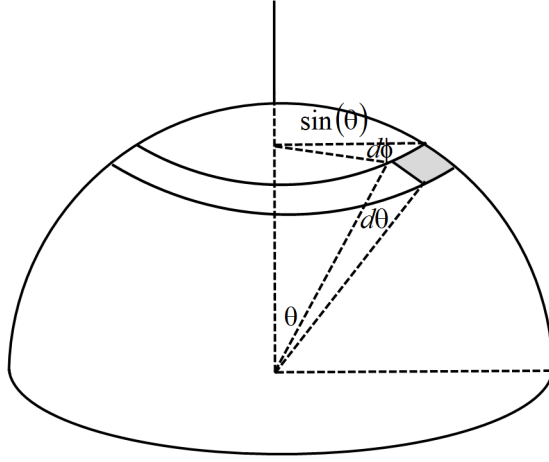


Figure 3-6: Diagram illustrating the portion of solid angles occupied by particles approaching a surface at angles between θ and $\theta + d\theta$, and between ϕ and $\phi + d\phi$.

The probability of particles effusing at angles between θ and $\theta + d\theta$ is therefore:

$$\rho_{2D}^{\text{eff}}(\theta)d\theta = \frac{1}{2\pi}d\theta$$

$$\rho_{3D}^{\text{eff}}(\theta)d\theta d\phi = \frac{1}{4\pi}\sin\theta d\theta d\phi$$

It is assumed that the molecules are in equilibrium, so their speeds are given by the corresponding Maxwell–Boltzmann distribution in 2D and 3D. Therefore:

$$\rho_{2D}^{\text{eff}}(c) = \frac{m}{k_B T} c \exp\left(-\frac{mc^2}{2k_B T}\right)$$

$$\rho_{3D}^{\text{eff}}(c) = \sqrt{\frac{2}{\pi}} \left(\frac{m}{k_B T}\right)^{3/2} c^2 \exp\left(-\frac{mc^2}{2k_B T}\right)$$

If molecules are traveling toward the surface at velocity c , then a volume of gas $A c \cos\theta$ strikes area A in unit time. The effusing flux contribution is then $n c \cos\theta$. The contribution to the total flux from molecules at angles between θ and $\theta + d\theta$ with speeds c and $c + dc$ is therefore:

$$\begin{aligned}
dJ_{2D}^{\text{eff}} &= \frac{1}{2\pi} \frac{m}{k_B T} \cos\theta d\theta \cdot c^2 \exp\left(-\frac{mc^2}{2k_B T}\right) dc \\
dJ_{3D}^{\text{eff}} &= \frac{1}{4\pi} \sqrt{\frac{2}{\pi}} \left(\frac{m}{k_B T}\right)^{3/2} \cos\theta \sin\theta d\theta d\phi \cdot c^3 \exp\left(-\frac{mc^2}{2k_B T}\right) dc
\end{aligned} \tag{3-23}$$

Compare this expression to the outgoing flux from a wall, noting that the form of the velocity distribution functions is similar to those in [Equation 3-15](#):

$$\begin{aligned}
dJ_{2D}^{\text{out}} &= \frac{1}{2} \sqrt{\frac{2}{\pi}} \left(\frac{m}{k_B T}\right)^{3/2} J \cos\theta d\theta \cdot c^2 \exp\left(-\frac{mc^2}{2k_B T}\right) dc \\
dJ_{3D}^{\text{out}} &= \frac{1}{2\pi} \left(\frac{m}{k_B T}\right)^2 J \cos\theta \sin\theta d\theta d\phi \cdot c^3 \exp\left(-\frac{mc^2}{2k_B T}\right) dc
\end{aligned}$$

Therefore, treat the flux effusing into the domain from the half space as a boundary with a flux J^{eff} given by:

$$J^{\text{eff}} = \left(\frac{k_B T}{2\pi m}\right)^{1/2} n$$

where the same result holds in 2D and 3D. Because the half space is infinite the pressure in this region can still be determined by the ideal gas law giving the following result for the flux at an external pressure, p :

$$J^{\text{eff}} = \left(\frac{1}{2\pi m k_B T}\right)^{1/2} p$$

Inverting this expression gives

$$p = (2\pi m k_B T)^{1/2} J^{\text{eff}}$$

which can be compared with [Equation 3-19](#)

$$p^{\text{out}} = (\pi m k_B T/2)^{1/2} J^{\text{eff}}$$

The factor of two difference occurs because the incoming pressure contribution is calculated separately in the case of [Equation 3-19](#).

There is considerable complexity in the detailed physics of gas adsorption and desorption (see [Ref. 2](#)). Some of this complexity is summarized in [Figure 3-7a](#), which shows some of the processes involved in the adsorption and desorption of an imaginary diatomic molecule. Molecules incident on a wall of the vacuum system are either reflected or adsorbed. The *sticking coefficient* defines the probability that an incident molecule is adsorbed. The molecule is usually *physically adsorbed* initially, being bound to the surface by weak van der Waals interactions between the adsorbate and the atoms in the solid surface, or in the case of multiple physically adsorbed layers, between the adsorbate and the underlying layer of adsorbed molecules. Molecules adjacent to the surface can then dissociate and *chemically adsorb* on the metal surface of the chamber. The reverse processes: chemical desorption, followed by reassociation can increase the number of physically adsorbed molecules on the surface. These physically adsorbed molecules can dissociate from the surface and consequently contribute to the desorbed flux. In addition to these processes, gases can pass through the walls of a vacuum system by the process of diffusion. Depending on the method of manufacturing, a newly constructed vacuum system contains a certain amount of dissolved gas in its walls. Initially the concentration profile of the gases is probably uniform through the thickness of the wall. When the chamber is pumped out, over a long period of time this uniform concentration changes to a roughly linear profile with a high concentration on the outside and a low concentration on the inside. It takes some time for this concentration profile to develop and the process of reaching this point is frequently referred to as *diffusion* in the vacuum literature. Once an equilibrium profile is reached then a constant mass transport occurs along the concentration gradient. This is known as *permeation* (although the process itself is, of course, diffusive). Permeation needs to account for the adsorption of the molecules onto the exterior surfaces of the vacuum chamber and the desorption of the molecules from the interior of the chamber.

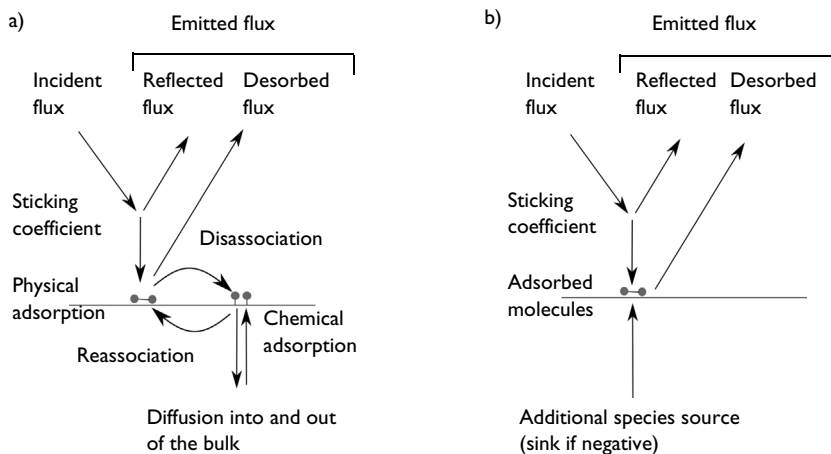


Figure 3-7: (a) Processes involved in the adsorption of an imaginary diatomic molecule. It is difficult to obtain the details of the individual reaction processes involved in this process for particular molecules. (b) The Molecular Flow Module's simplified adsorption/desorption scheme. This scheme assumes that the molecule is adsorbed as a whole. Although this might not be physically what is happening, the effect of the dissociation can still be added to the model by appropriate modification of the sticking coefficient and the desorption rate.

Modeling the adsorption and desorption of molecules on the surface of a vacuum chamber at this level of detail is possible if the Free Molecular Flow interface is coupled with the Chemical Reaction Engineering Module's Surface Reactions interface. However, in practice there is usually insufficient data available to fully characterize the reactions occurring on the surfaces of the chamber. Furthermore, the barriers to desorption of physically adsorbed species are small at room temperature, so physical adsorption has little effect on molecular transport in the vacuum system (Ref. 3). It is therefore usually appropriate to consider only chemical adsorption. Given these considerations, the Adsorption/Desorption feature considers only the chemical adsorption of the molecule as a whole (see Figure 3-7b). If the molecule dissociates on adsorption then it is still possible to use this mechanism for practical modeling, but allowance must be made for the fact that the boundary condition tracks the total number of moles of the original gas molecule that have been adsorbed (in the example of the diatomic molecule the number of moles of the adsorbed species is double this number). An additional source term for the number of adsorbed molecules can be used to include the effects of diffusion and permeation through the chamber walls. Very complex and realistic situations can be modeled with this approach.

The adsorption/desorption boundary condition specifies outgoing flux and the number of adsorbed molecules in the following manner:

$$\begin{aligned} J &= (1 - S)G + DN_A \\ \frac{dn_{\text{ads}}}{dt} &= S \frac{G}{N_A} - D + \Gamma \end{aligned}$$

where J is the emitted molecular flux (SI unit: molecules/m²/s), S is the *sticking coefficient*, G is the incident molecular flux (SI unit: molecules/m²/s), N_A is Avogadro's number, n_{ads} is the number of adsorbed moles (SI unit: mol/m²), D is the desorption rate (SI unit: mol/m²/s), and Γ is the additional source term for the adsorbed molecules (SI unit: mol/m²/s). S , D , and Γ , along with an initial value for the number of adsorbed molecules ($n_{\text{ads},0}$), are specified in the boundary condition. These values can be specified as arbitrary functions of the concentration of adsorbed species (`fmf.n_ads`) allowing for highly flexible modeling of the adsorption and desorption process.

Evaporation

The [Evaporation](#) boundary condition specifies the outgoing flux using the following expression:

$$J = \alpha_v \left(\frac{N_A^2}{2\pi M_n R T} \right)^{1/2} p_{\text{vap}}$$

where α_v is the evaporation coefficient (SI unit: dimensionless), p_{vap} is the vapor pressure (SI unit: Pa), N_A is Avogadro's number, M_n is the molecular weight (SI unit: kg/mol), R is the universal gas constant (SI unit: J/(mol K)), and T is the surface temperature (SI unit: K).

References for the Free Molecular Flow Interface

1. F. Celestini and F. Mortessagne, "Cosine Law at the Atomic Scale: Toward Realistic Simulations of Knudsen Diffusion," *Phys. Rev. E*, vol. 77, p. 021202, 2008.
2. J.M. Lafferty, *Foundations of Vacuum Science and Technology*, Wiley Interscience, 1998.
3. J.F. O'Hanlon, *A User's Guide to Vacuum Technology*, Wiley Interscience, 2003.

Glossary

This [Glossary of Terms](#) contains finite element modeling terms specific to the Molecular Flow Module and its applications. For mathematical terms as well as geometry and CAD terms specific to the COMSOL Multiphysics® software and its documentation, see the glossary in the *COMSOL Multiphysics Reference Manual*. To find references in the documentation set where you can find more information about a given term, see the index.

Glossary of Terms

adsorption Attachment of a molecule or atom to a solid surface. Adsorption involves a chemical bond between the adsorbed species and the surface.

angular coefficient method A method for solving molecular flows that computes the total flux arriving at a point by integrating the flux arriving from all other surface elements visible from that point.

Boltzmann equation A partial differential equation that describes the evolution of the density and velocity distribution function for a gas in both space and time.

Boltzmann BGK equation A form of the Boltzmann equation with a simplified collision term involving a single relaxation time. The equation was introduced in a paper by Bhatnagar, Gross, and Krook.

continuum flow Fluid flow that is well described by approximating the liquid as a continuum with the Navier–Stokes equations.

desorption The release of a molecule or atom from a solid surface to which it was previously attached. Desorption involves breaking the chemical bond between the adsorbed species and the surface.

diffusion Transport of material resulting from the random motion of molecules in the presence of a concentration gradient. In the context of vacuum systems, this term can refer to the transient diffusive process that occurs when gas molecules within the walls of the system diffuse into the chamber after it is initially pumped down. Once a constant concentration gradient is present in the chamber walls, the diffusive transport through the walls is often referred to in the context of permeation.

discrete velocity method A kinetic method for solving the Boltzmann equation, usually with a simplified collision term (for example, the Boltzmann BGK equation). The method removes the velocity dependence of the distribution function by discretizing the velocity space and solving separate convection equations for each of the discrete velocities. These equations are coupled together by the scattering term.

DSMC method A kinetic method for solving the Boltzmann equation, which computes the trajectories of large numbers of randomized particles.

effusion The process by which gas is transported through a hole that is small compared to the mean free path.

equilibrium function A function that describes the equilibrium concentration of molecules belonging to one of the discrete velocities used in the lattice Boltzmann method.

free molecular flow The flow of gas molecules through a geometry that is much smaller than the mean free path (Knudsen number, $Kn > 10$). In the free molecular flow regime, the gas molecules collide with the walls of the geometry much more frequently than they collide with themselves.

Knudsen number A dimensionless number that provides a measure of how rarefied a gas flow is; in other words, it is the mean free path of the gas molecules compared to the length scale of the flow. The following equation defines the Knudsen number Kn where λ is the mean free path of the molecules and L is a length scale characteristic to the flow.

$$Kn = \frac{\lambda}{L}$$

Knudsen layer A layer of rarefied fluid flow that occurs within a few mean free paths of the walls in a gas flow. The continuum Navier–Stokes equations break down in this layer.

lattice Boltzmann method A discrete velocity method which is optimized for near continuum flows.

permeation A three-step process in which gases, especially hydrogen, can enter a vacuum system from the outside. The gas first adsorbs on the outer surface of a vacuum system, then it diffuses through the walls of the vessel, and finally it desorbs from the interior wall of the system.

quadrature A term describing the set of discrete velocities employed in the lattice Boltzmann method.

SCCM Standard cubic centimeters per minute. A unit of flow in which the quantity of gas flowing is converted into a volumetric flow rate under standardized conditions.

slip flow Fluid flow that occurs when the *Knudsen number*, Kn , is in the range $0.01 < Kn < 0.1$. As a result of rarefaction effects in the *Knudsen layer* the no slip boundary condition fails. The flow outside the Knudsen layer can be represented by the continuum Navier–Stokes equations provided that an appropriate slip boundary condition is used for the fluid flow and the correct temperature jump boundary condition is applied at the interface.

sticking coefficient The probability that molecules incident on a surface are adsorbed.

transitional flow Fluid flow that occurs when the *Knudsen number*, Kn , is in the range $0.1 < Kn < 10$. In this regime, the flow is so rarefied that continuum equations break down completely. However, collisions between the molecules are still important so free molecular flow is not applicable.

I n d e x

- A**
 - angular coefficient method 64
 - Application Libraries window 11
 - application library examples
 - free molecular flow 53
 - Monte Carlo modeling 21, 27
- B**
 - Boltzmann BGK equation 18, 30, 40
 - Boltzmann equation 18, 40
 - boundary conditions 19
 - boundary conditions, transitional flow 44
 - boundary nodes
 - free molecular flow 54
 - transitional flow 32
- C**
 - chemical adsorption 78
 - common settings 9
 - continuity on interior boundary (node)
 - 35
 - continuum flow 16
 - cosine law 65
- D**
 - diffuse flux (node)
 - free molecular flow 59
 - transitional flow 37
 - direct area integration method 51
 - direct gauges 26
 - direct simulation Monte Carlo method
 - 18, 21
 - discrete velocity method 18
 - documentation 10
 - domain nodes
 - free molecular flow 54
 - transitional flow 32
- E**
 - edge nodes, free molecular flow 54
 - effusion 39, 60
 - emailing COMSOL 12
 - evaporation (node) 56
- F**
 - flow properties (node) 34
 - flow rate in SCCMs 37
 - flow regimes, for rarefied flows 16
 - free molecular flow 21
 - free molecular flow interface 50
 - theory 64
- G**
 - gauges, for pressure 26
 - Gaus-Hermite quadrature 47
 - global nodes, free molecular flow 54
 - Grad's moment scheme 41
- H**
 - hemicube integration method 51
 - hemicube method 24
 - high vacuum pump (node) 36
- I**
 - ideal gas law 25
 - indirect gauges 26
 - initial values (node)
 - free molecular flow 56
 - transitional flow 35
 - inlet (node) 35
 - internet resources 10
- K**
 - knowledge base, COMSOL 12
 - Knudsen number 16, 21
 - Knudsen's law 44
- L**
 - lattice Boltzmann method 18, 30, 40
 - lattice directions 43
- M**
 - Maxwell-Boltzmann distribution 65
 - mean free path 43
 - molecular effusion 39, 60
 - molecular flow (node) 55
 - Monte Carlo method 64
 - MPH-files 11
- N**
 - Navier–Stokes limit 47
 - nodes, common settings 9
 - number density reconstruction (node)

- 55
- number density, vs. pressure 26
- O** outgassing wall (node) 39
- outlet (node) 36
- P** pair nodes, free molecular flow 54
- permeation 78
- physical adsorption 78
- physics interfaces, common settings 9
- point nodes, free molecular flow 54
- pressure, vs. number density 26
- pumps 60
- Q** quadrature 18
- quadrature, for velocity 47
- R** rarefied gas flow 16
- reservoir (node)
 - free molecular flow 59
 - transitional flow 38
- S** scattering time 43
- SCCM, flow rate in 37
- standard cubic centimeters per minute
 - 37
- standard settings 9
- sticking coefficient 78
- surface temperature (node) 55
- T** technical support, COMSOL 12
- theory
 - free molecular flow 64
 - transitional flow 40
- total vacuum (node)
 - free molecular flow 59
 - transitional flow 38
- transitional flow 18
- transitional flow interface 30
 - theory 40
- V** vacuum pump (node) 60
- velocity quadrature 47
- W** wall (node)
 - free molecular flow 57
 - transitional flow 35
- websites, COMSOL 12