



# INTRODUCTION TO Particle Tracing Module

# Introduction to the Particle Tracing Module

© 1998–2023 COMSOL

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

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

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

Version: COMSOL 6.2

## Contact Information

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

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

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

Part number: CM022702

# Contents

---

Contents . . . . .	3
Introduction . . . . .	5
Main Subject Areas . . . . .	6
. . . . .	7
Charged Particle Tracing . . . . .	8
Monte Carlo Collision Modeling . . . . .	9
Multiphysics Interfaces for Particle Field Interaction . . . . .	10
Releasing Nonlaminar Ion and Electron Beams . . . . .	13
Particle Tracing for Fluid Flow . . . . .	14
Treatment of Particle Inertia . . . . .	14
Basic Requirements . . . . .	14
Flow Regimes . . . . .	14
Particle Diffusion . . . . .	16
Rarefaction Effects . . . . .	19
Wall Effects . . . . .	19
Other Built-in Forces . . . . .	19
Computing Particle Mass and Temperature . . . . .	20
Nozzles and Sprays . . . . .	20
Mathematical Particle Tracing . . . . .	21
Boundary Conditions . . . . .	22
Conditional Particle-Wall Interactions . . . . .	23
Releasing Secondary Particles . . . . .	23
Particle Release Mechanisms . . . . .	24
Advanced Modeling Tools . . . . .	27

Particle Statistics .....	27
Auxiliary Dependent Variables and Residence Time .....	28
Particle Counters.....	28
Velocity Reinitialization in Domains .....	28
Computing Sum, Average, Maximum, and Minimum .....	28
Results Processing Tools.....	29
Particle Trajectories Plots.....	29
Filters .....	29
Poincaré Maps and Phase Portraits.....	30
Particle Evaluation.....	31
Operations on Particle Data .....	31
Histograms.....	31
Tracing Particles Through a Laminar Static Mixer .....	33

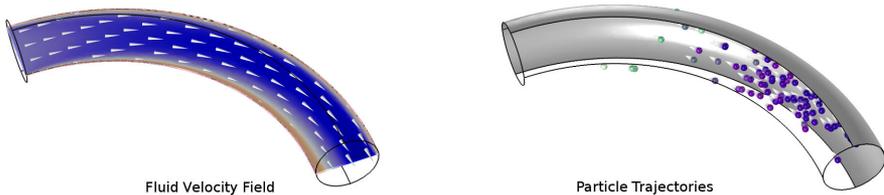
# Introduction

---

The Particle Tracing Module is a general-purpose tool for computing the paths of particles as they move through a geometry and are subjected to various forces. The simulated particles could represent ions, electrons, cells, grains of sand, projectiles, planets, stars, and much more.

Most of the physics interfaces in COMSOL Multiphysics use the Finite Element Method (FEM) to compute fields such as temperature, fluid velocity, electric potential, concentration, or displacement. In contrast, particle tracing provides a Lagrangian description of a problem, in which the particles are treated as discrete entities that can interact with external fields, with boundaries in the surrounding geometry, and with each other. Their trajectories are computed in the time domain by solving a set of equations based on their laws of motion. The most common such system of equations are the second-order ordinary differential equations based on Newton's second law of motion, to which the forces based on external fields can easily be added.

As particles travel through the geometry of a model, they may also come into contact with boundaries in their surroundings. When this happens, each surface can affect the particle behavior in a different way. The built-in particle-boundary interactions include different types of absorption and combinations of diffuse and specular reflection. The Particle Tracing Module also offers a wide variety of release mechanisms to control how the particles are inserted into the geometry. After the simulation is complete, many dedicated tools for visualizing particle motion and reporting particle statistics are available.



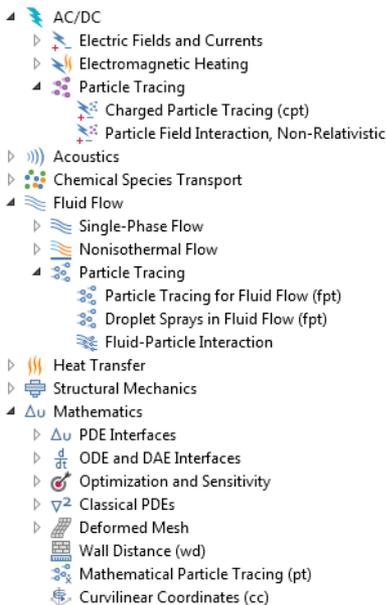
*A basic particle tracing model usually involves first solving for some fields, such as a fluid velocity field (left), then tracking particles in the field, using forces such as the built-in Drag Force (right).*

## Main Subject Areas

Particle Tracing Module applications can be loosely categorized into three subject areas:

- Charged particle tracing, the calculation of ion or electron paths under electric, magnetic, and collisional forces;
- Particle tracing in fluids, the tracking of micron-scale particles such as cells or airborne contaminants under the influence of drag, gravity, and other field forces; and
- Mathematical particle tracing, the setup of customized equations of motion, including the Hamiltonian and Lagrangian formulations from classical mechanics.

These three subject areas correspond to the three main physics interfaces included with the Particle Tracing Module: the Charged Particle Tracing interface () , the Particle Tracing for Fluid Flow interface () , and the Mathematical Particle Tracing interface () , respectively. In addition, several multiphysics interfaces are available, each with predefined couplings to other physics interfaces such as Electrostatics () and Laminar Flow () .



*Locations of Particle Tracing physics interfaces in the Model Wizard. The options shown are available with licenses for COMSOL Multiphysics and the Particle Tracing Module. Depending on other licensed products, the Model Wizard may include options other than those that are shown here.*

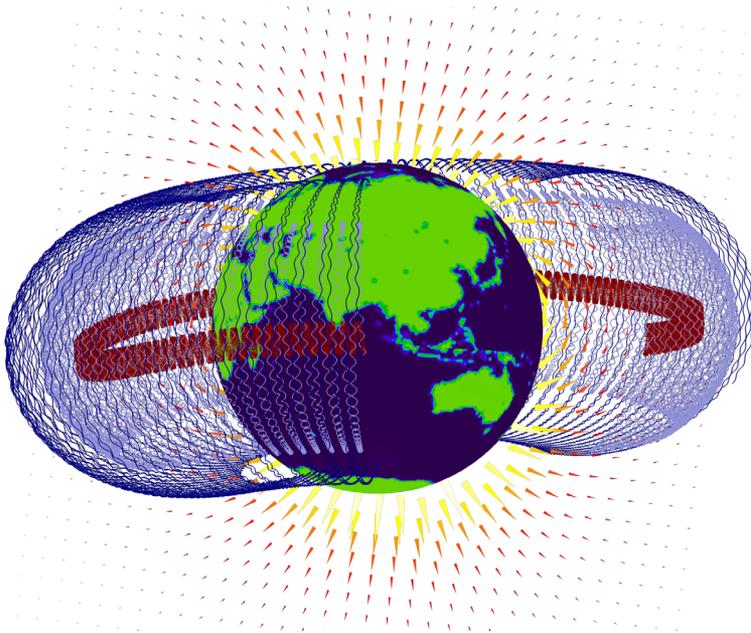
## Physics Interface Guide by Space Dimension and Preset Study Type

PHYSICS INTERFACE	ICON	TAG	SPACE DIMENSION	AVAILABLE STUDY TYPE
 <b>AC/DC</b>				
 <b>Particle Tracing</b>				
Particle Field Interaction, Nonrelativistic		—	3D, 2D, 2D axisymmetric	bidirectionally coupled particle tracing; time dependent
Particle Field Interaction, Relativistic <sup>1</sup>		—	3D, 2D, 2D axisymmetric	bidirectionally coupled particle tracing; time dependent
Charged Particle Tracing		cpt	3D, 2D, 2D axisymmetric	bidirectionally coupled particle tracing; time dependent
 <b>Fluid Flow</b>				
 <b>Particle Tracing</b>				
Particle Tracing for Fluid Flow		fpt	3D, 2D, 2D axisymmetric	bidirectionally coupled particle tracing; time dependent
Fluid-Particle Interaction		—	3D, 2D, 2D axisymmetric	bidirectionally coupled particle tracing; time dependent
Droplet Sprays in Fluid Flow		—	3D, 2D, 2D axisymmetric	bidirectionally coupled particle tracing; time dependent
 <b>Mathematics</b>				
Mathematical Particle Tracing		pt	3D, 2D, 2D axisymmetric	bidirectionally coupled particle tracing; time dependent
<sup>1</sup> Requires the addition of the AC/DC Module.				

## Charged Particle Tracing

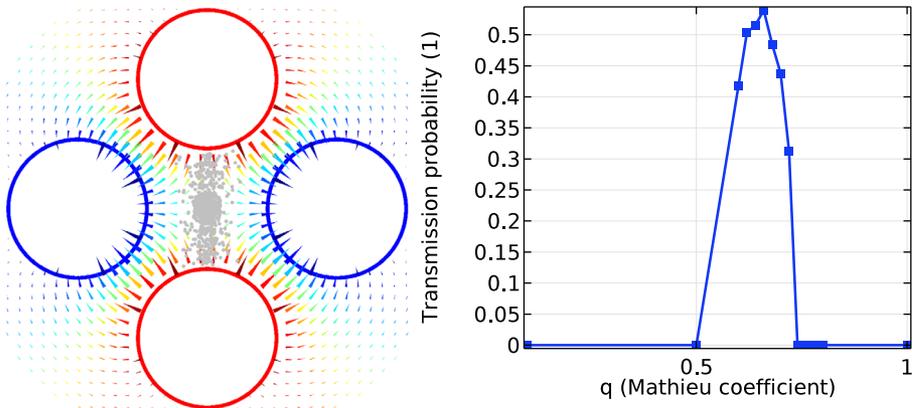
---

The Charged Particle Tracing interface (  ) is found under the AC/DC branch (  ) in the Model Wizard. It is primarily used to trace ions and electrons under the influence of electric and magnetic forces. These electric and magnetic forces can either be entered explicitly as user-defined expressions or computed using another physics interface. The magnetic force can also be taken from built-in data for Earth's magnetic field.



*Motion of trapped protons in Earth's magnetic field. The protons follow helical paths that oscillate between the northern and southern hemispheres.*

Included in the Electric Force and Magnetic Force features are built-in options to modulate AC fields. In this way, it is easy to include combinations of stationary and time-harmonic forces in a single model.



*Quadrupole mass filter: Argon ions propagate through a quadrupole arrangement of electrodes (left). The stability of the design is determined by the magnitude of the DC and AC potentials used to guide the ions through the filter and reported as a function of the dimensionless Mathieu coefficient (right).*

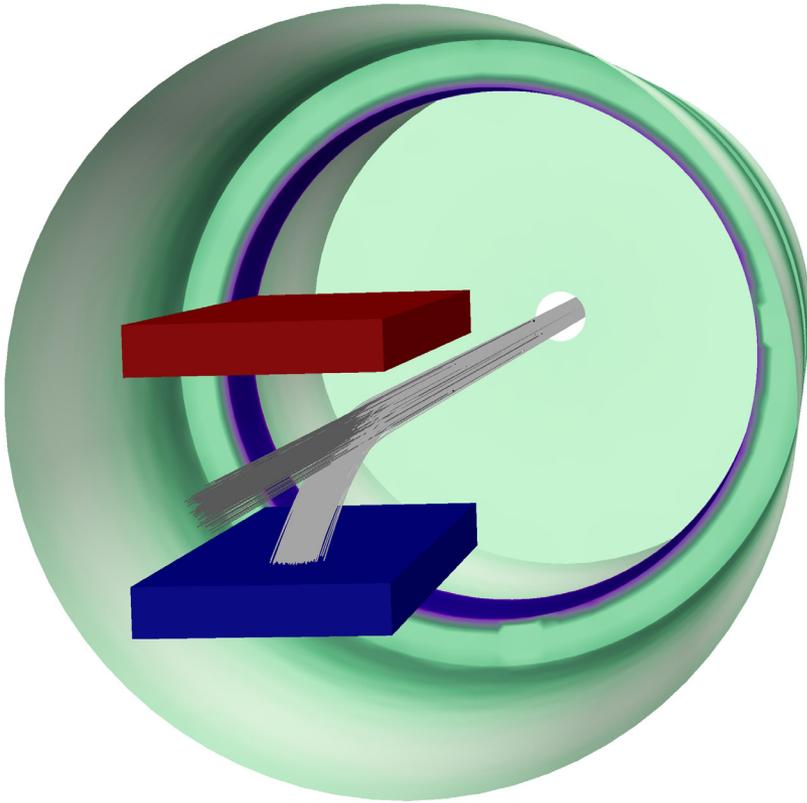
### Monte Carlo Collision Modeling

In addition to electric and magnetic forces, energetic ions and electrons are often subjected to collisions with the molecules in a rarefied background gas. At sufficiently low pressure, these collisions can be modeled stochastically using a Monte Carlo approach. In each time step taken by a time-dependent solver, each model particle has a chance to collide with a background molecule and be scattered in a random direction.

The collision probability and the probability distribution function of scattered particle velocities are based on the particle kinetic energy, type of reaction, background gas density, and collision cross section data. Because the collision cross section is usually a nontrivial function of energy, it is common to load cross section data from a separate file and use it to define an interpolation function.

Some of the built-in collision types can release secondary particles of the same or a different species. For example, ionization reactions can yield secondary electrons, and charge exchange reactions can yield new combinations of ionized and neutralized species.

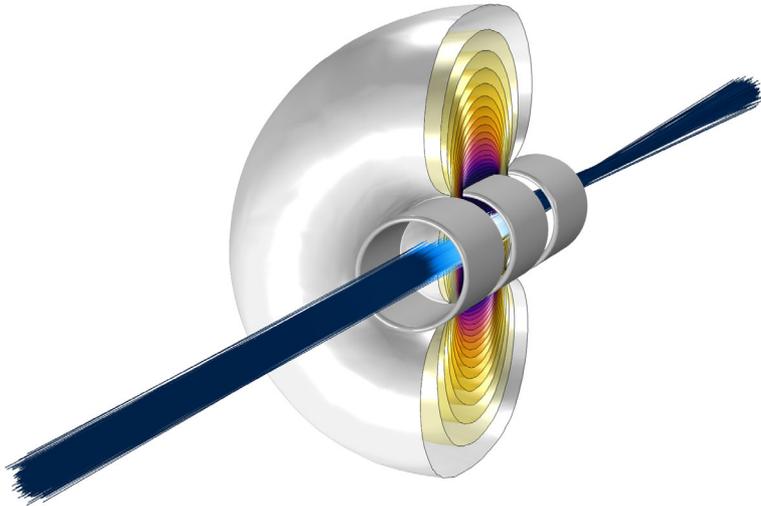
Because of its Monte Carlo collision modeling capability, the Charged Particle Tracing interface is sometimes extremely useful in modeling the motion of neutralized species, not just ions or electrons, in spite of the interface's name.



*Charge exchange cell: A fast neutral beam is produced by accelerating protons (light gray) and passing them through a vacuum chamber containing a rarefied argon gas. Collisions with the background gas produce energetic neutral particles (dark gray). The remaining protons are deflected by an electrostatic field between two oppositely charged plates (red and blue).*

### ***Multiphysics Interfaces for Particle Field Interaction***

If the number density of charged particles is sufficiently low, the Coulomb interaction between ions or electrons can sometimes be neglected, since it is likely to be dwarfed in magnitude by the external electric and magnetic fields. Such models use a unidirectional coupling from the electric and magnetic fields to the particle trajectories. In other words, first the fields are computed, and then these fields are used to exert forces on the charged particles.



*Einzel lens: The electrostatic force from the electrodes, illustrated by isosurfaces of electric potential, is much greater than the Coulomb interaction between the electrons, making this a unidirectionally coupled model.*

## COULOMB INTERACTION

If the density of charged particles is suitably high, then it may be necessary to include the Coulomb force that acts between the particles. The built-in Coulomb force model for the Particle-Particle Interaction node computes the instantaneous force on each charged particle based on the position and charge number of all other particles in the model.

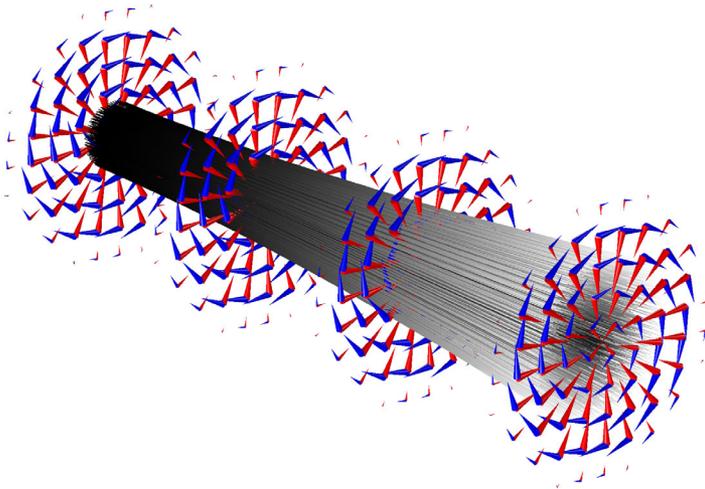
When particle-particle interactions are included in a model, the computational requirements to evaluate the force increase quadratically with the total number of particles because each particle's motion is affected by the position of every other particle in the model. When including the Coulomb force, it is often best to start with a small number of particles, solve the model, and then assess whether or not the effect is important.

The Coulomb force is well suited to models in which the interaction between particles may be significant but the total population of particles in the model remains constant over time. For beam simulations with space charge effects, however, ions or electrons are not just affected by particles that are released simultaneously, but also particles that are released earlier or later in time. This can lead to a very large number of model particles being required to accurately capture the Coulomb interaction. A two-way coupled particle-field interaction, described in the next section, might be a better choice.

## BIDIRECTIONAL PARTICLE-FIELD INTERACTION

An alternative to directly applying the Coulomb force is to allow particles to contribute to the space charge density in the domain for the purpose of computing the electric potential, then to use this modified electric potential to compute a more accurate electric force on the particles. This is called a bidirectional coupling between the particles and fields. The bidirectional coupling is well suited to simulations in which ions or electrons are released continuously over time such that the electric potential at any point in space remains approximately constant. The dedicated Particle Field Interaction, Non-Relativistic interface (  ) combines the Charged Particle Tracing (  ) interface with the Electrostatics (  ) interface to quickly set up the two-way interaction in which the particles perturb the electric potential in their surroundings.

A beam of relativistic ions or electrons can generate sufficient current to produce a significant magnetic force on the beam. This self-induced magnetic force can be implemented using the Particle Field Interaction, Relativistic interface (  ), which automatically combines the Charged Particle Tracing (  ), Electrostatics (  ), and Magnetic Fields (  ) physics interfaces. This multiphysics coupling requires both the AC/DC Module and the Particle Tracing Module.

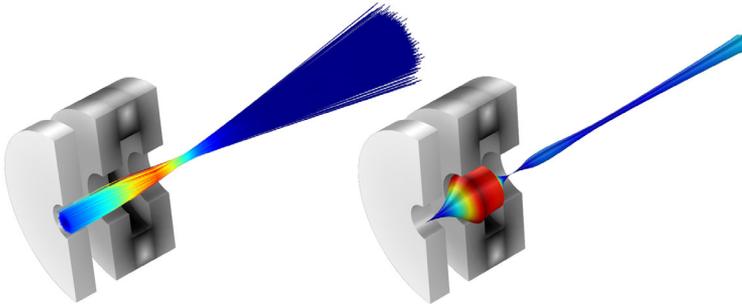


*Relativistic electron beam: despite being released at the beam waist, the beam electrons (gray) diverge because the outward force exerted by the electric field (red) is greater in magnitude than the inward force exerted by the magnetic field (blue).*

## *Releasing Nonlaminar Ion and Electron Beams*

The Charged Particle Tracing interface includes a dedicated feature for releasing ion and electron beams. You can specify quantities such as the beam emittance and Twiss parameters to characterize the particle position and velocity distributions.

When one or more charged particle beams are released in a model, quantities such as beam emittance are automatically computed along the nominal beam trajectory and are available for postprocessing.



*A beam of electrons is focused by a magnetic lens (left). The beam emittance can then be plotted as the thickness and color along the nominal beam path (right).*

## Particle Tracing for Fluid Flow

---

The Particle Tracing for Fluid Flow interface (  ), found under the Fluid Flow branch (  ) in the Model Wizard, computes the motion of particles in a background fluid. Particle motion can be driven by built-in forces including drag, gravitational, electric, magnetic, and acoustophoretic radiation forces. You can also apply custom user-defined forces, model particle-particle interactions, and solve for particle mass and temperature.

### *Treatment of Particle Inertia*

The Particle Tracing for Fluid Flow interface includes dedicated formulations for tracking inertial particles or neglecting the inertial terms. A full treatment of inertial particles means that the acceleration of the particle in the surrounding fluid is fully resolved in time. For very small particles (usually around a micron or smaller), this acceleration takes place over such a short time that it is more computationally efficient to simply assume that the particles are always in an equilibrium state, with the drag force perfectly counterbalancing all other applied forces such as gravity. This allows you to take significantly larger time steps when tracking small particles in a fluid.

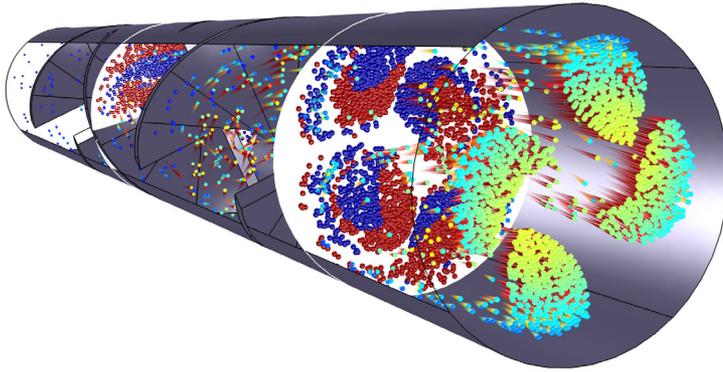
### *Basic Requirements*

In order for the particle tracing approach to be valid, the system should be a *dilute* or *dispersed* flow. This means that the particles should occupy a very small fraction of the volume of the surrounding fluid, generally less than 1%. When the volume fraction of the particles is not small, the fluid system is categorized as a *dense* flow and a different modeling approach is required.

It is important to realize that with the particle tracing approach, particles do not displace the fluid they occupy. Furthermore, the finite radius of the particle is not usually taken into account when detecting and applying particle-wall interactions.

### *Flow Regimes*

Motion of microscopic and macroscopic sized particles is typically dominated by the drag force acting on particles immersed in a fluid. There are two phases in the system: a *discrete phase* consisting of bubbles, particles, or droplets, and a *continuous phase* in which the particles are immersed. The velocity of the continuous phase is sometimes entered as an expression but is most often computed using a fluid flow physics interface such as Laminar Flow (  ), Turbulent Flow, k- $\epsilon$  (  ), or Creeping Flow (  ).



Comet tail plot of particle trajectories through a laminar static mixer (colored). In addition, Poincaré maps show the deviation of particle trajectories from their initial position (blue & red).

## SPARSE FLOW

In a *sparse flow*, the continuous phase affects the motion of the particles via the drag force, but the particles do not have enough inertia to significantly perturb the continuous phase. This is usually true when the particles in the discrete phase are very small or have relatively low number density. This is often referred to as unidirectional coupling. When modeling such a system, it is usually most efficient to solve for the continuous phase first, then compute the trajectories of the discrete particles in a separate study. For example, in the following plot of a static mixer, the stationary velocity field and pressure were first solved using a stationary study, and then the particle trajectories were computed in a time-dependent study.

## DILUTE FLOW

In a *dilute flow* the continuous phase affects the motion of the particles, and the particle motion in turn disrupts the continuous phase. This is often referred to as a bidirectional coupling. Compared to a sparse flow, the particles in a dilute flow typically have more inertia, due to the particles being bigger, denser, or more numerous.

The Fluid-Particle Interaction (☰☪) interface automatically sets up a bidirectional coupling between the Particle Tracing for Fluid Flow interface (☪☰) and the Laminar Flow interface (☰☰). The particles exert a volume force on the fluid at

their location, equal in magnitude and opposite in direction to the total drag force exerted on the particles. The computational demand is significantly higher when modeling dilute flows than sparse flows. Before setting up a bidirectional coupling, it is often beneficial to first set up the model with a unidirectional coupling and then determine whether the effect of the particles on the surrounding fluid is significant.

## DISPERSED FLOW

In a *dispersed flow*, the density of particles is greater than in the dilute flow described in the previous section, but their volume fraction is still lower than in a dense flow. The dispersed flow is the upper limit of the applicability of a Lagrangian particle tracking approach. In addition to the bidirectional fluid-particle interaction mentioned in the previous section, particle-particle interactions may also need to be taken into account. This is sometimes referred to as four-way coupling. Particle-particle interactions can be included in models but the following limitations apply:

- Hard sphere collisions are not supported. Forces must vary continuously with respect to the distance between particles as in, for example, the Coulomb force between charged particles.
- The computation time scales as the number of particles squared. This is because every particle interacts with every other particle over all distances. It is possible to apply a cutoff distance for particle-particle interactions, but the interaction force must still be evaluated for every pair of particles.
- If the particle-particle interaction law is highly nonlinear, it may be necessary to use a very small time step. This is particularly true if the Lennard-Jones option is selected.

## Particle Diffusion

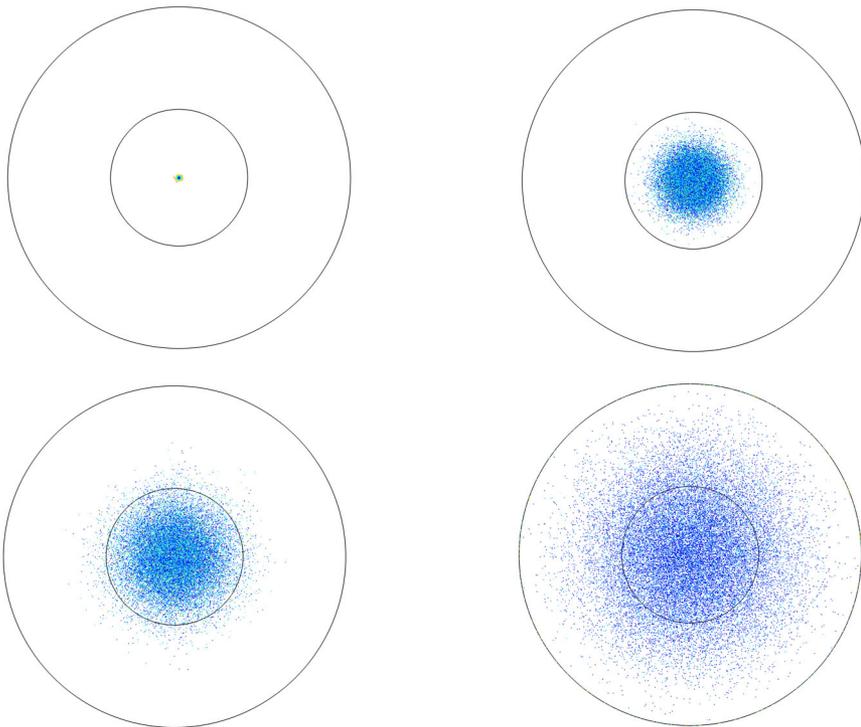
Particle motion in a fluid is a combination of advective and diffusive transport. Advection is the bulk transport of particles by the mean fluid velocity via the drag force. Advection is typically deterministic; given a particle as a specified location in a laminar flow field, the motion of that particle at future times is completely predictable and reproducible (allowing, of course, some small differences due to the discrete time stepping used by the time dependent study). If 100 identical particles pass through the same point in a stationary flow field, each having the same velocity at that point, then all 100 particles will follow the same path.

Diffusion, on the other hand, is implemented as a stochastic term that can give different results for each particle in the simulation, even if the initial conditions are the same. There are two main mechanisms for diffusive transport in the Particle Tracing for Fluid Flow interface: molecular diffusion and turbulent dispersion.

## MOLECULAR DIFFUSION: THE BROWNIAN FORCE

Molecular diffusion is most significant for extremely small particles, on the order of  $0.1\ \mu\text{m}$  or smaller. When such small particles are in a fluid of nonzero absolute temperature, the thermal motion of individual atoms or molecules in that fluid causes them to collide randomly with the particles in directions other than that of the mean flow velocity. These random collisions can cause particles to spread out over time, a phenomenon known as Brownian motion.

The COMSOL implementation of the Brownian force involves seeding unique random numbers for each particle at each time step taken by the solver. Therefore, a sufficiently large number of time steps should be taken so that the results show statistical convergence. In a similar fashion, the number of model particles should be sufficiently large so that the average behavior of the particles can be clearly seen.



*Brownian motion of particles that are released from a single point. They diffuse outward. Upper left  $t = 0\ \text{s}$ , upper right  $t = 10\ \text{s}$ , lower left  $t = 30\ \text{s}$ , and lower right  $t = 100\ \text{s}$ .*

## TURBULENT DISPERSION

Turbulence is the dominant mechanism for diffusive transport of macroscopic particles in a high Reynolds number flow. Unlike Brownian motion, turbulent dispersion can remain significant even for particles at the micron scale or larger. For laminar flow, the drag force only contributes to the advective transport of the particles,

$$\mathbf{F}_D = \frac{1}{\tau_p} m_p (\mathbf{u} - \mathbf{v})$$

However, if the flow is turbulent, then the fluid velocity  $\mathbf{u}$  at the particle's position consists of a mean flow term and an instantaneous fluid velocity perturbation,

$$\mathbf{u} = \bar{\mathbf{u}} + \mathbf{u}'$$

This is because most of the turbulence models in COMSOL Multiphysics are variants of the Reynolds Averaged Navier Stokes (RANS) model. In a RANS model, eddies in a turbulent flow are not modeled explicitly; instead, information about the magnitude and lifetime of eddies is only retained in a statistical sense, often through the use of transport equations for additional dependent variables. Several different expressions for the perturbation term exist, the simplest being

$$\mathbf{u}' = \zeta \sqrt{\frac{2k}{3}}$$

where  $\zeta$  is a random unit vector and  $k$  is the turbulent kinetic energy. Some turbulence models, such as the  $k$ - $\epsilon$  model, solve for  $k$  as an additional degree of freedom; it represents the amount of energy associated with eddies in the flow.

Unlike molecular diffusion, turbulent diffusion has a finite resolution in time. The turbulent velocity perturbation experienced by a certain particle is assumed to have the same magnitude and direction for a finite time interval, based on the lesser of two time scales: the eddy lifetime (over which eddies or vortices in the flow die out and form again) or the eddy crossing time (over which particles have sufficient inertia to be thrown out of one eddy and into another one). For extremely small particles, the eddy crossing time may grow so large as to become irrelevant.

If the time step taken by the solver is made extremely small, then eventually the turbulent perturbation term between successive iterations no longer consists of uncorrelated random vectors. The built-in discrete random walk and continuous random walk models compensate for this effect by only seeding the random perturbation term at discrete times based on the turbulent dissipation rate  $\epsilon$ , if it is available. Therefore, it is important to take a sufficiently small time step so that the changes in this random number seed can be resolved.

## Rarefaction Effects

Some forces, such as the drag and thermophoretic forces, include high Knudsen number corrections that improve accuracy when the particles are extremely small or the surrounding fluid is a highly rarefied gas. These rarefaction effects begin to become significant when the Knudsen number, defined as the ratio of the mean free path of surrounding molecules to the particle diameter, is greater than 0.1.

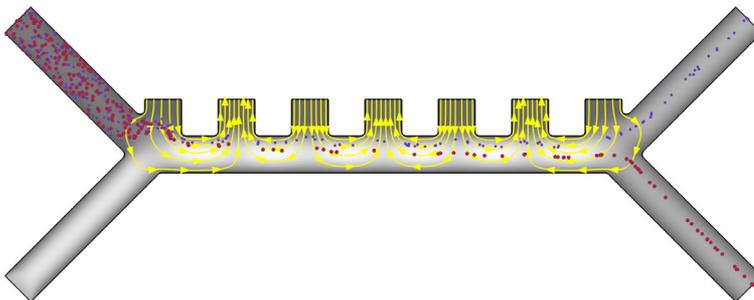
## Wall Effects

The drag force and lift force include correction terms to account for the presence of nearby walls, improving their accuracy for channel flows. For turbulent wall-bounded flows, an anisotropic turbulent dispersion model is available, taking different fluctuations in the streamwise, spanwise, and wall normal directions.

## Other Built-in Forces

In addition to the drag force and Brownian force, the Particle Tracing for Fluid Flow interface includes a wide variety of other built-in forces that can be used to create high-fidelity multiphysics models, including the following:

- Electric and magnetic forces can be exerted on charged particles.
- The virtual mass and pressure gradient forces can affect particle motion in a liquid, based on the spatial and time derivatives of the liquid velocity.
- Dielectrophoretic and magnetophoretic forces can affect particles in nonuniform electric and magnetic fields.
- The acoustophoretic radiation force affects particles in pressure waves.
- The thermophoretic force causes particles to migrate in a temperature gradient. This force is most significant in rarefied gases.
- Lift forces are exerted on particles in a shear flow.



Dielectrophoretic separation: of red blood cells (red) and platelets (blue) in a flow channel. The fluid velocity is shown in grayscale. The electric field is indicated by yellow streamlines.

## Computing Particle Mass and Temperature

You can solve additional equations for the particle mass and temperature along each particle trajectory. When solving for temperature, built-in features for radiative and convective heating and cooling become available. You can also add user-defined heat sources and sinks.

If the particles are heated or cooled due to radiative or convective heat exchange with their surroundings, then optionally you can compute the dissipated particle heat as a volumetric source or sink term. For example, if hot particles are cooled by being submerged in cold water, you can add the Dissipated Particle Heat node to compute the corresponding heat source term in the fluid, potentially enabling two-way coupled simulation of particle tracing in a nonisothermal flow.

When solving for particle mass, you can make particles grow or shrink as they propagate. When releasing particles, you can enter the initial particle mass directly or sample it from a distribution, such as a lognormal distribution with a specified Sauter mean diameter.

As an alternative to specifying particle growth rate directly, if the particles represent liquid droplets, you can use a dedicated feature to compute the evaporation rate for droplets of a given size and temperature. When solving for both particle mass and temperature, the built-in Stefan-Fuchs evaporation model can optionally include an evaporative cooling term.

## Nozzles and Sprays

The Particle Tracing for Fluid Flow interface is not just limited to solid particles in liquids or gases, but can also be used to model sprays of liquid droplets.

A dedicated Nozzle feature is available to release a spray of droplets from points in the geometry. In addition, the Droplet Breakup feature can be used to split the model particles into smaller particles over time. Typically the breakup occurs due to instability on the droplet surface created by external drag forces, or due to sudden acceleration of the droplet.

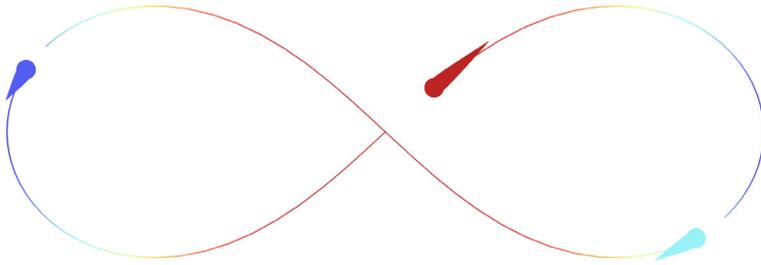
The Droplet Sprays in Fluid Flow interface () , found under the Fluid Flow branch () in the Model Wizard, is a special case of the Particle Tracing for Fluid Flow interface () that uses default settings and features that are frequently used to model sprays of liquid droplets. This includes options to specify particle viscosity and surface tension as well as an instance of the Droplet Breakup feature.

# Mathematical Particle Tracing

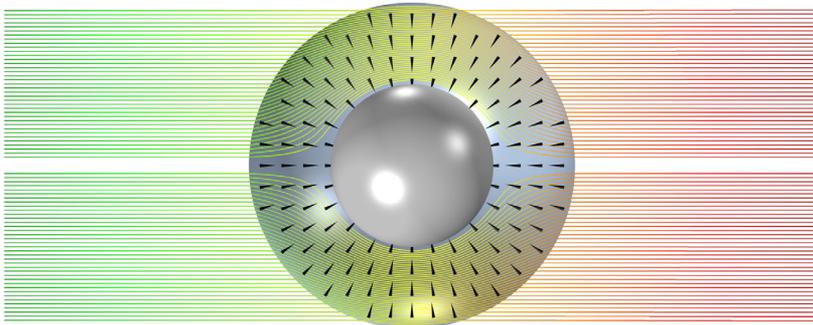
---

The Mathematical Particle Tracing interface (  ) gives access to the underlying mathematical formalism on which the Charged Particle Tracing (  ) and Particle Tracing for Fluid Flow (  ) interfaces are built. The Mathematical Particle Tracing interface also allows for specification of particle motion in terms of either a Lagrangian or Hamiltonian. Often it is easier to write down an expression for the Lagrangian or Hamiltonian for particles rather than deriving the equations of motion.

The Hamiltonian formulation solves for both the particle position and the particle momentum, so the number of degrees of freedom is doubled when the Hamiltonian formulation is activated, but the equations become first order instead of second order.



*Solution to the gravitational three-body problem, in which the gravitational attraction is modeled as a user defined Particle-Particle Interaction force. Shown above is the stable figure-eight configuration.*

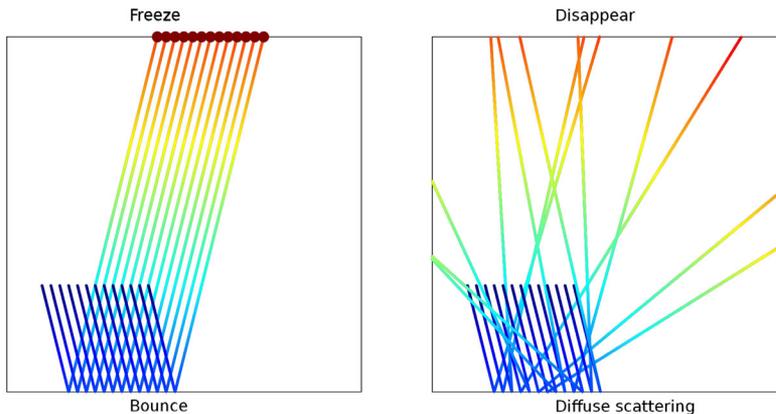


*A user-defined Hamiltonian can be used to model omnidirectional optical cloaking of an object by a rotationally invariant, anisotropic medium. The arrows indicate that the optic axis points in the radial direction.*

## Boundary Conditions

---

The following options are available for the boundary conditions for particles when they make contact with a wall: Bounce, Freeze, Stick, Disappear, Pass through, Diffuse scattering, Isotropic scattering, Mixed diffuse and specular reflection, and General reflection. Some of the available options are illustrated in the figure below.



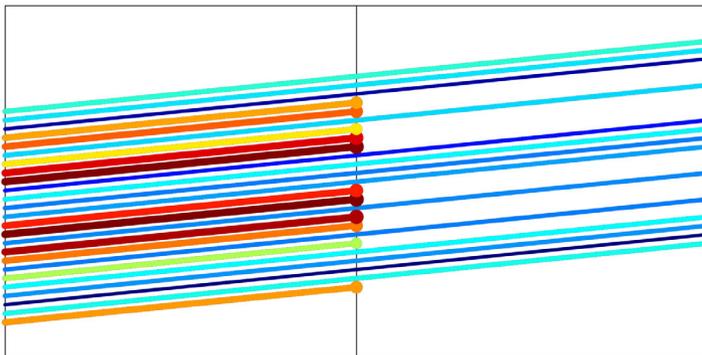
*Demonstration of the Bounce, Freeze, Diffuse scattering, and Disappear Wall conditions.*

- The Freeze, Stick, and Disappear options all stop a particle when it hits a boundary. The Stick option allows the particle to move if the boundary moves, as in the Moving Mesh interface (). The Disappear option completely removes particles from the simulation when they hit a surface.
- The Bounce option specularly reflects from the wall such that the particle kinetic energy is conserved.
- The Diffuse scattering option bounces particles off a wall according to the cosine law, sometimes called Knudsen's cosine law or Lambertian reflection depending on the application area. The probability of a particle bouncing off the surface in a given direction within a differential solid angle  $d\omega$  is given by  $\cos(\theta)d\omega$ , where  $\theta$  is the angle between the direction of the reflected particle and the wall normal. The kinetic energy of the particle is conserved.
- The Isotropic scattering option bounces particles off a wall in a random direction. In contrast with the Diffuse scattering option, the particle has an equal probability of being reflected in any direction. The kinetic energy of the particle is conserved.

- The Mixed diffuse and specular reflection condition is a combination of the Bounce and Diffuse scattering conditions, each with a given probability.
- The Pass through option allows particles to cross boundaries unimpeded. By combining this with other Wall conditions, it is possible to filter particles based on their properties or with a given probability.
- The General reflection option allows an arbitrary post-collision velocity to be specified. This can either be done in Cartesian coordinates or in the tangent-normal coordinate system. The velocity components can be functions of the incident particle velocity, energy, or any other quantity.

### ***Conditional Particle-Wall Interactions***

The above boundary conditions for particle-wall interaction can all be conditional, either assigned a probability or contingent on a logical expression being satisfied.



*A conditional Pass through Wall condition can be used to filter particles based on properties such as particle mass or diameter. Here the interior boundary acts as a screen that blocks large particles (marked mostly in shades of red and orange) but allows smaller particles to pass through unimpeded.*

### ***Releasing Secondary Particles***

Secondary particles can be introduced into the modeling domain when a primary particle strikes the wall. Any number of secondary particles can be emitted during a single wall collision. The released particles can be assigned initial velocity based on an expression or sampled from a Lambertian or isotropic distribution.

The degrees of freedom for secondary particles are allocated when the study begins, not when the secondary particles are emitted. Therefore, enabling the release of secondary particles increases the number of degrees of freedom in the solution at all time steps.

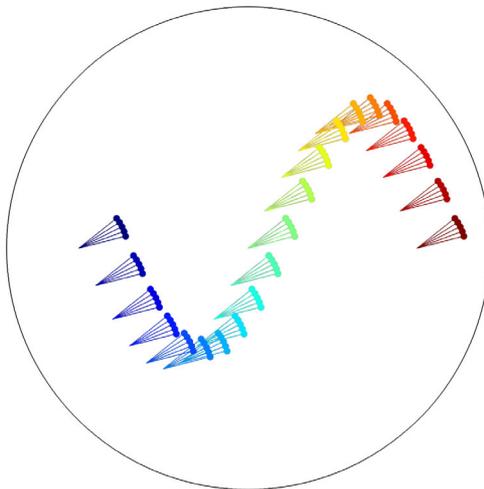
# Particle Release Mechanisms

---

There are several ways to release particles: either from a grid, by importing a data file, or by selecting geometric entities that mark the initial particle positions. In each case the number of particles and the release times can be specified. There are also specific options available for each release feature as described below.

## RELEASE FROM A GRID

Particles can be released from a grid that can either be regular or graded. In addition to the initial coordinates of the particles, the initial velocity and times to release the particles can be specified. There are also predefined initial velocity distributions available to release particles with a Maxwellian, Isotropic spherical, hemispherical, conical, or Lambertian distribution. Particles can either be placed at all combinations of the initial coordinates or only at specified combinations of them.

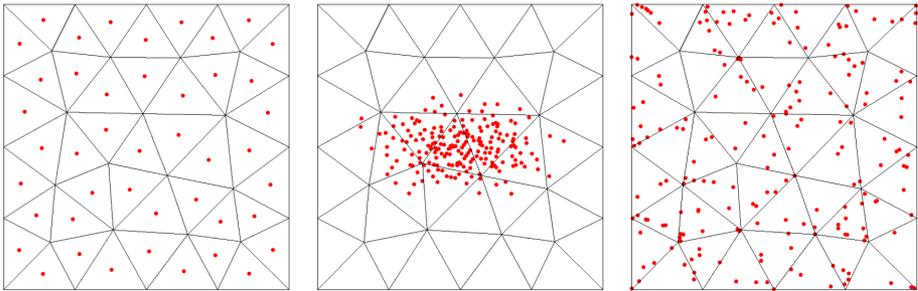


*The initial particle coordinates can be a list of values or expressions, like the sine curve shown above. At each release point, a single particle can be released or particle velocities can be sampled from a distribution. Here a cone of particles is released at each point.*

## RELEASE FROM A DOMAIN

There are several different mechanisms for choosing the release points from within a selected domain.

- When the particle release is Mesh based, there is an option to specify the Refinement factor. The higher this factor, the more particles are released from within each mesh element.
- When the particle release is Density based, the release of particles is weighted according to a user-defined expression. This expression for number density can be an analytic expression, a random function, or even a field variable computed by a previous study.
- The Random option allows the initial position of the particles to be chosen at random. New random positions are chosen for each release time.



*Comparison of the mesh-based release (left), density-based release using a bi-Gaussian distribution (center), and random release in the selected domain (right).*

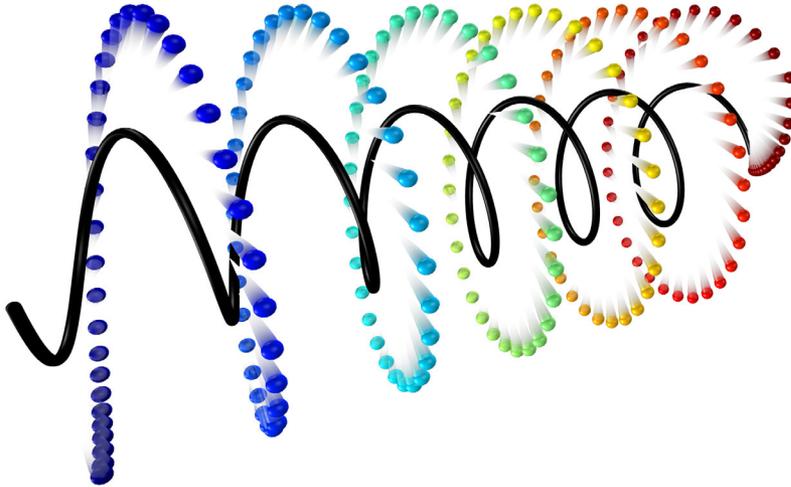
## RELEASE FROM A BOUNDARY (INLET)

The Inlet feature releases particles from a set of boundaries. The options for the domain-based Release are also available here, as well as an additional option to release particles uniformly from a selected boundary in 2D or from a projected plane grid in 3D.

## RELEASE FROM EDGES AND POINTS

The Release from Edge feature can be used to release particles along edges in a 3D geometry. Similarly, the Release from Point feature can be used to release particles from selected points in the geometry.

When releasing particles along edges, particle positions can be mesh-based, weighted by a user-defined density, or uniformly distributed along the edges.



*Using the Release from Edge feature, particles can be released from positions along arbitrary curves, such as the helix shown above.*

## RELEASE FROM DATA FILE

The Release from Data File feature allows the initial positions of particles to be specified by selecting columns from an imported text file. The initial velocity can also be read from the text file. If any Auxiliary Dependent Variable nodes are present, the initial values of auxiliary dependent variables can also be imported from the file.

When loading the initial particle positions from a file, you can also prescribe combinations of translation, rotation, and dilation. If a rotation is applied to the positions, and the particle velocity is also loaded from the same file, then optionally you can apply the same rotation to the loaded velocity vectors.

## Advanced Modeling Tools

---

The Particle Tracing Module provides a wide range of specialized modeling tools to assist in extracting specific quantities of interest.

### *Particle Statistics*

The particle tracing interfaces define a number of special variables, some of which can only be used during results processing. These variables include information about the overall population of particles in the model.

These variables are categorized as particle statistics and release statistics. Some are described below. The variable names are given assuming that the physics interface is an instance of the Mathematical Particle Tracing interface with tag `pt`. When using the Charged Particle Tracing interface, for example, replace `pt` with `cpt`.

The following variables are defined for each particle:

- Particle index `pt.pidx`. Each particle is assigned a unique index starting from 1 up to the total number of particles. This expression can be used as an input argument to a function. For example, `randomnormal(pt.pidx)` would sample pseudorandom numbers from a normal distribution with unit variance. The random numbers would be distinct for each particle.
- Particle release feature `pt.prf`. If there are multiple release features in a model, it is useful to be able to visualize how the particles mix together based on their initial release position. Particles from each release feature are assigned a unique integer corresponding to that feature, starting at 1.
- Release time `pt.rti`. This indicates when each particle first enters the modeling domain. It works for both primary and secondary particles.
- Stop time `pt.st`. Using the difference between the release time and stop time, it is easy to compute the residence time of each particle.

There are also variables that are only available during results processing and can only be evaluated using the Global Evaluation node under Derived Values.

- Total number of particles `pt.Nt`. This includes primary and secondary particles, those that have not been released and those that have disappeared.
- Total number of particles in selection `pt.Nsel`. If a selection has been applied to the Particle dataset, the number of particles in that selection can be evaluated.
- Transmission probability `pt.alpha`. Often the transmission probability is the main quantity of interest in a particle tracing model. As with the number of particles in selection `pt.Nsel`, the Particle dataset should be assigned a selection consisting of a set of domains or boundaries in the model.

## ***Auxiliary Dependent Variables and Residence Time***

Auxiliary dependent variables can be used to help keep track of quantities like residence time, particle trajectory length, and integrated shear rate along each particle's path. When an Auxiliary Dependent Variable feature is added to the physics interface a first-order additional ordinary differential equation (ODE) is solved for each particle. This equation can be integrated either over time or along the particle trajectories.

The residence time can also be computed in a different way, by setting the Store particle status data property to On. This creates variables for the particle release time and the particle stop time (the time when a particle left the modeling domain). The residence time is then simply the difference between the two.

## ***Particle Counters***

Particle counters can be added to domains or boundaries in order to provide information about particles arriving on the set of selected domains or surfaces. It is possible to analyze the transmission of all particles or only those particles that are released by a specific feature. The Particle Counter computes quantities such as the number of particles transmitted, transmission probability, transmitted current, mass flow rate, and so on. The feature provides convenient expressions that can be used in the Filter node of the Particle Trajectories Plot, which allows only the particles which reach the Particle Counter selection to be visualized.

## ***Velocity Reinitialization in Domains***

It is possible to stop a particle or change its velocity when a specified logical expression is satisfied, even if the particle is not in contact with a boundary. For example, particles can be made to disappear after a certain amount of time, or to change direction if their  $z$ -coordinate exceeds a specified value. These interactions can also cause the emission of secondary particles; for example, a particle can be made to split into two or more smaller particles if its mass exceeds a threshold.

## ***Computing Sum, Average, Maximum, and Minimum***

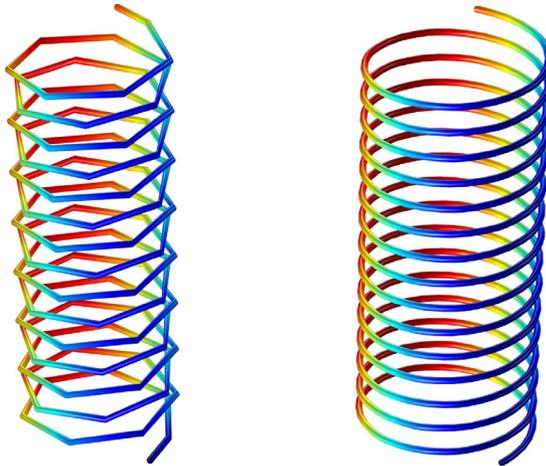
Each particle tracing physics interface defines nonlocal couplings to compute the sum, maximum, minimum, and average of an expression over all particles. For example, when using the Mathematical Particle Tracing interface with default name `pt`, the expression for the total sum of all particles would be `pt.sum(pt.mp)`, whereas the average particle speed would be `pt.ave(pt.V)`. The couplings can easily be evaluated using the Global Evaluation node, or plotted as a function of time using the Global plot in a 1D Plot Group.

## Results Processing Tools

---

### *Particle Trajectories Plots*

A Particle Trajectories plot is available in both 2D  and 3D . It is created automatically when a Study solving for a particle tracing interface is computed. There are options to set the particle trajectories to be rendered as lines, tubes, or ribbons. It is also possible to render the particles as points, comet tails, arrows, or with a number of ellipses along each trajectory. For models where the number of output times is small, an interpolation option is available that will create smoother looking particle trajectory plots.

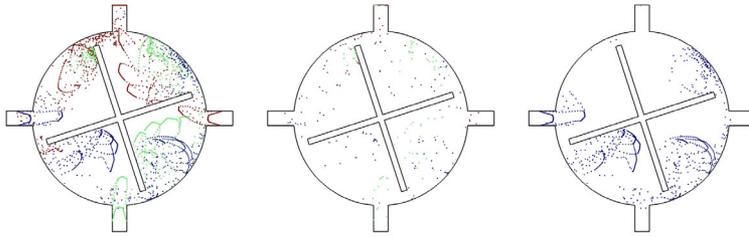


*The trajectory of an ion in a uniform magnetic field is plotted with no interpolation (left) and with interpolation (right).*

### *Filters*

Visualizing the trajectory of systems with a very large number of particles can consume a lot of computer resources and create cluttered-looking plots in which particles obscure each other from view. It is possible to filter the type of particle and the number of particles which should be rendered. To do this, right click the Particle Trajectories plot type and choose Filter.

The particles can be filtered so that only primary particles secondary particles, or those satisfying a specified logical expression are plotted. It is also possible to plot only a specified fraction or number of particles.

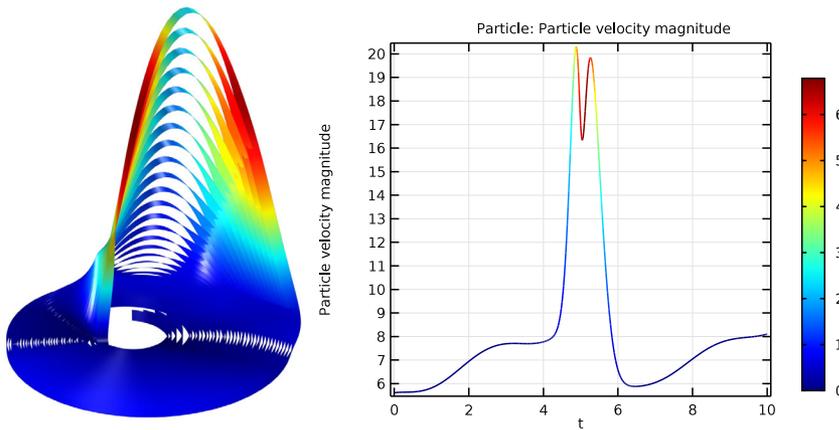


Particle trajectories in a micromixer (left). It is possible to render only a fraction of the particles (middle) and to plot only the particles that are released by a certain inlet using a logical expression (right).

## PARTICLE PLOTS

A Particle plot  can be used to observe the value of a particle property over time. Built-in data series operations can be used, for example, to compute the sum or maximum of an expression over all particles.

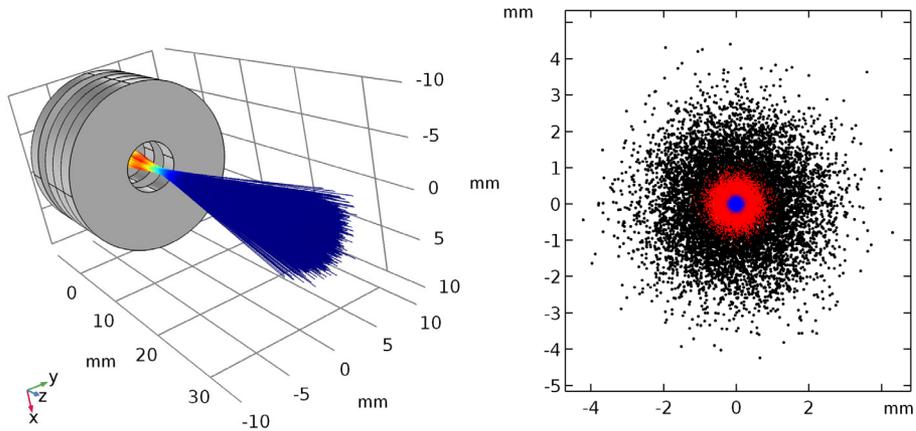
Alternatively, the Particle plot can be used to compare two particle properties against each other for all particles at a set of selected time steps.



Rosser attractor: 3D plot of particle trajectories (left) and 1D plot of the average speed over time (right).

## Poincaré Maps and Phase Portraits

Poincaré maps are available in both 2D  and 3D . This plot type is useful to visualize the particle trajectories in a plot that represents the position of the particles in a section that is usually transversal to the particle trajectories. The Poincaré map represents the particle trajectories in a space dimension that is one dimension lower than the original particle space.



Electrons in a magnetic lens (left). The Poincaré maps (right) show the electron distribution in beam cross-sections at three different axial positions.

Phase portraits  are available as a 2D plot type under More Surface Plots. Use a Phase Portrait plot to visualize large datasets of particle trajectories. The traditional use of a phase portrait is to plot the particle position on the  $x$ -axis and the particle velocity on the  $y$ -axis. Each dot in the  $xy$ -plane represents a particle.

### Particle Evaluation

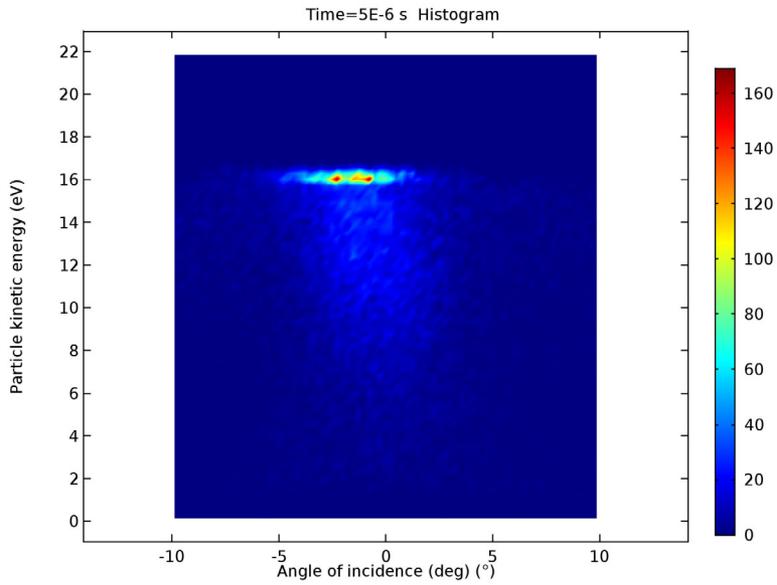
Information about expressions and variables along particle trajectories can be written to the Results Table using the Particle Evaluation  option under Derived Values. Once the data has been written to the results table it can be manipulated and plotted accordingly. There are options to only write data to the table for a fraction or a specific number of particles.

### Operations on Particle Data

It is possible to set the source dataset for the Integration , Average , Maximum , and Minimum  to be a Particle  dataset. This allows operations to be performed on the particle dataset, to compute average particle velocity, maximum energy, and so on.

### Histograms

Statistical information about the behavior of the particles is often best visualized with a histogram. The histogram sorts the value of a variable into a specified number of bins. The most obvious application of the histogram is that it allows for visualization of the velocity and energy distribution function of a set of particles.

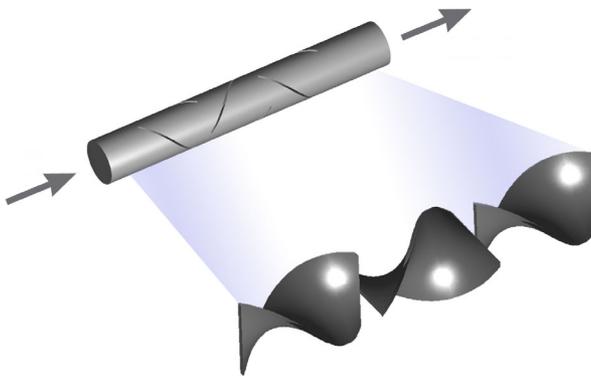


*Information about the ion energy distribution function in a plasma can be visualized using a 2D Histogram.*

## Tracing Particles Through a Laminar Static Mixer

---

This section takes you through the modeling stages of computing particle trajectories based on a computed fluid velocity field. In static mixers, also called motionless or in-line mixers, a fluid is pumped through a pipe containing stationary blades. This mixing technique is particularly well suited for laminar flow mixing because it generates only small pressure losses in this flow regime. This example studies the flow in a twisted-blade static mixer. It evaluates the mixing performance by calculating the trajectory of suspended particles and displaying their positions at several cross sections of the mixing blades.



### Model Wizard

---

**Note:** These instructions are for the user interface on Windows but apply, with minor differences, also to Linux and Mac.

- I To start the software, double-click the COMSOL icon on the desktop. When the software opens, you can choose to use the Model Wizard  to create a new COMSOL model or Blank Model  to create one manually. For this tutorial, click the Model Wizard  button.

If COMSOL is already open, you can start the Model Wizard by selecting New  from the File menu and then click Model Wizard .

The Model Wizard guides you through the first steps of setting up an application. The next window lets you select the dimension of the modeling space.

- 2 In the Select Space Dimension window click the 3D button .
  - 3 In the Select Physics tree under Fluid Flow  > Single-Phase Flow , click Laminar Flow .
- Even though this is a particle tracing model, the first step is to solve for the fluid velocity field, which will then be used to exert a drag force on the particles. The Particle Tracing for Fluid Flow interface  will be added later.
- 4 Click Add.
  - 5 Click the Study  button.
  - 6 In the tree under General Studies, click Stationary . Although the particles are moving through the channel over time, the fluid velocity at any fixed point does not change over time, making a stationary analysis possible.
  - 7 Click the Done  button.

## Geometry

---

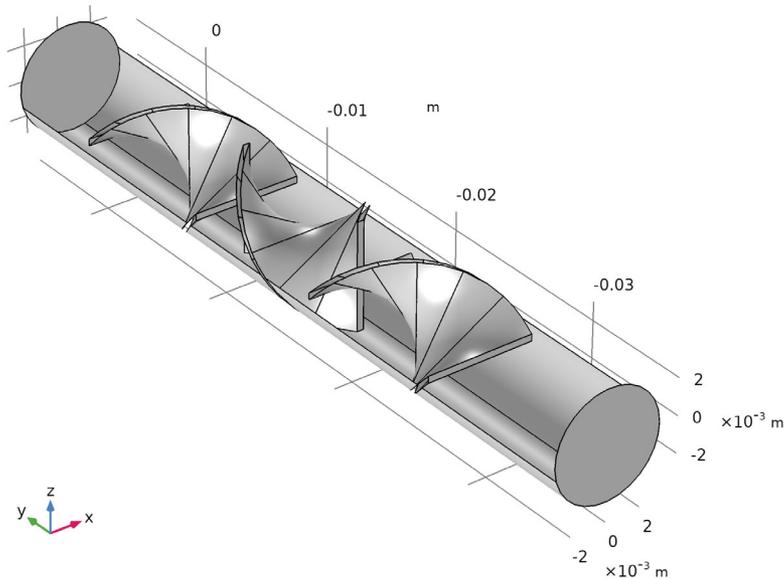
To save time, you can load the geometry sequence from a file. This sequence can then be edited at any time by expanding the Geometry  node.

- 1 In the Geometry toolbar click Insert Sequence .
- 2 Browse to the application library folder and double-click the file `laminar_mixer_particle_geom_sequence.mph`. The location of the files used in this exercise varies based on your installation. For example, if the installation is on your hard drive, the file path might be similar to `C:\Program Files\COMSOL\COMSOL62\Multiphysics\applications\`. The folder for this example is `Particle_Tracing_Module\Fluid_Flow`.
- 3 Click Build All  to see the geometry.

Alternatively, you can build the geometry, one step at a time. The modeling instructions for the *Particle Trajectories in a Laminar Static Mixer* example model in the Application Libraries include an appendix with a complete set of geometry setup instructions. Another alternative is to load a CAD geometry from another program by clicking the Import  button in the Geometry toolbar.

The surfaces of the mixing blades can be difficult to see because they are covered by the outer surfaces of the flow channel. There are several ways to view the inside of the geometry more easily. One option is to click the Wireframe Rendering button  in the Graphics toolbar, to hide all surfaces in the Graphics window.

An alternative is to click the Select Boundaries  and Click and Hide  buttons in the Graphics toolbar. Then you can hide the outer surfaces of the channel by clicking on them. The resulting view is shown below.



The mixer geometry consists of three twisted blades, each making a 90 degree angle with the adjacent blades. The middle blade is twisted in the opposite direction from the first and last blades.

## Global Definitions

- 1 In the Home toolbar click Parameters  $P_i$  and then select Parameters 1  $P_i$  .  
Alternatively, click the Parameters 1  $P_i$  node in the Model Builder.

**Note:** On Linux and Mac, the Home toolbar refers to the specific set of controls near the top of the Desktop.

- 2 In the Settings window locate the Parameters section.
- 3 The inserted geometry sequence uses a global parameter  $R_a$ , called the Tube radius, with a value of 3 millimeters. This parameter was automatically defined when the geometry sequence was inserted in the previous steps. The text in the Expression column, 3[mm] with square brackets, is a standard way to enter unit syntax in COMSOL models.
- 4 Add a second row to the table by typing  $u_{av}$  in the Name column, 1[cm/s] in the Expression column, and Mean velocity in the Description column. This global parameter will be used to define boundary conditions in the model.

For the parameter values entered above, and for water with an approximate density of  $1000 \text{ kg/m}^3$  and viscosity of  $0.001 \text{ Pa s}$ , the Reynolds number of the flow is approximately 60, meaning that the use of the Laminar flow physics interface is appropriate.

## Definitions

---

Next, under the Component 1  >Definitions  node, some named selections will be added. These will be used during results processing.

### OUTER SURFACES

- 1 In the Definitions toolbar, click  Explicit.
- 2 In the Settings window for Explicit, type Outer Surfaces in the Label text field.
- 3 Locate the Input Entities section. From the Geometric entity level list, choose Boundary.
- 4 Select boundaries 1, 8, 19-23, 48-50, 57, and 58. These are the boundary numbers for the outer walls of tube.

An easier way to apply this selection is to first select the Group by continuous tangent check box. Then, in the Graphics window, click anywhere on the outer surface of the cylinder. This will cause that boundary and all boundaries tangent to it to be selected. Then, you only need to click the two ends of the cylinder to add them to the selection.

### BLADE SURFACES

- 1 In the Definitions toolbar, click  Explicit.
- 2 In the Settings window for Explicit, type Blade Surfaces in the Label text field.
- 3 Locate the Input Entities section. From the Geometric entity level list, choose Boundary.
- 4 Select boundaries 2-7, 9-18, 24-47, and 51-56, the mixing blade surfaces.

A shortcut for this selection is to first click Go to XY View  in the Graphics toolbar, then click Select Box . Click and drag in the Graphics window until the box contains all three blades, then release the mouse pointer. You might need to deselect just one or two outer pipe segments, like boundary 21.

## Materials

---

- 1 In the Materials toolbar click Blank Material . To make sure the material appears under Component 1  in the Model Builder, not under Global Definitions , make sure the last node you clicked was under Component 1.
- 2 In the Settings window for Material locate the Material Contents section. In the table, enter the following settings:

Material Contents					
Property	Variable	Value	Unit	Property group	
<input checked="" type="checkbox"/> Density	rho	1000	kg/m <sup>3</sup>	Basic	
<input checked="" type="checkbox"/> Dynamic viscosity	mu	1e-3	Pa·s	Basic	

By default the first material you define in the model is applied to all domains, so it is not necessary to select the flow channel.

## Physics Interfaces

---

Now add an expression for the inflow velocity which is parabolic.

- 1 In the Physics toolbar click Boundaries  and choose Inlet .
- 2 Select Boundary 23 only. This is the end of the pipe located at  $y = 6$  mm. Alternatively, Click Paste Selection  and type 23 in the dialog box.
- 3 In the Settings window for Inlet locate the Boundary Condition section. In the list, the default is Velocity, meaning that the inflow velocity distribution is specified directly. Instead, select Fully developed flow from the list. Then locate the Fully Developed Flow section. In the text field for the Average velocity  $U_{av}$ , type the parameter  $u_{av}$ .

**Note:** The fully developed flow profile will have zero velocity at the walls that surround the inlet, where a no-slip boundary condition is enforced. Because the inlet cross section is circular, the velocity will be greatest at its center. The incoming velocity distribution given by the Fully developed flow option represents what you would get if the flow entered the geometry through a very long pipe of uniform cross section before encountering the mixing blades.

- 4 In the Physics toolbar click Boundaries  and choose Outlet .
- 5 Select Boundary 20 only. This is the surface opposite Boundary 23 which was selected for the Inlet boundary condition. It is located at  $y = -36$  mm.

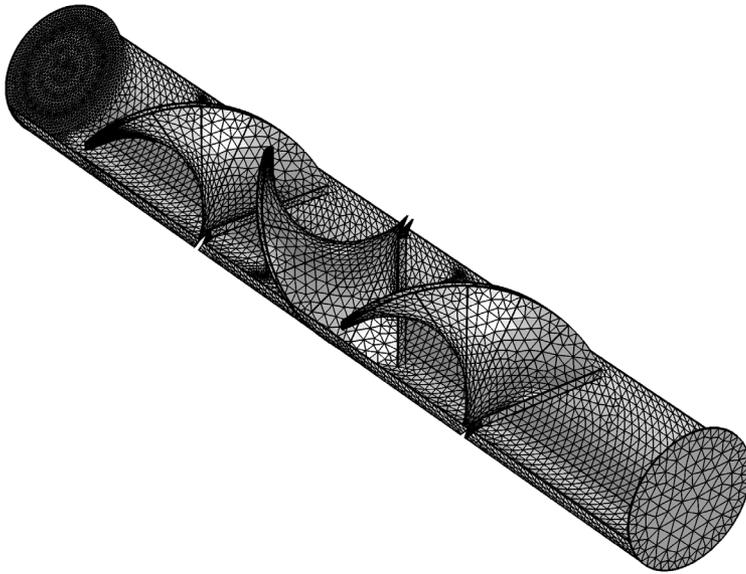
## Mesh

---

The mesh needs to be quite fine to ensure that the particle motion is accurate throughout the modeling domain. In this case, take care to ensure that the mesh is sufficiently fine close to the mixing blades.

- 1 In the Mesh toolbar, click Boundary  and choose Free Triangular .
- 2 Select Boundaries 5, 16–18, and 53–55 only. These boundaries include the leading edges of all three mixing blades.
- 3 Right-click Free Triangular 1  and choose Size .
- 4 In the Settings window for Size locate the Element Size section.
  - From the Calibrate for list, choose Fluid dynamics.
  - From the Predefined list, choose Extremely fine.
- 5 Click the Size  node above the Free Triangular  node.
- 6 In the Settings window for Size locate the Element Size section.
  - From the Predefined list, choose Extremely fine.
  - Click the Custom button. By first choosing the Extremely fine option and then switching to a Custom mesh, you can begin customizing the mesh with a reasonable set of default values.
  - Locate the Element Size Parameters section. In the Curvature factor text field, type 0.15. The curvature factor is used to refine the mesh on curved surfaces while allowing the mesh to be somewhat coarser on planar surfaces.
- 7 In the Mesh toolbar, click Boundary  and choose Free Triangular .
- 8 Select Boundary 23, the Inlet boundary, only.
- 9 Right-click Free Triangular 2  and choose Size .
- In the Settings window for Size locate the Element Size section. From the Calibrate for list, choose Fluid dynamics.
- From the Predefined list, choose Extra fine.
- 10 Right-click Mesh 1  and choose Free Tetrahedral . Alternatively, select Free Tetrahedral  from the Mesh toolbar.
- 11 Click Build All .

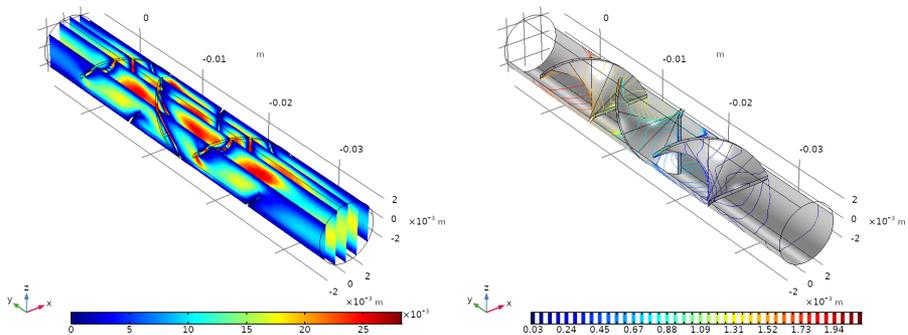
The mesh should look like the following image. As for the geometry, some outer surfaces have been hidden from view so that the mesh on the mixing blades can be seen easily.



## Study

---

- | In the Home toolbar click Compute . In this model, the default solver sequence for the Stationary study is sufficient to model the flow. The study creates default plots of the fluid velocity and pressure as shown below. The legends have been moved to the bottom of each plot to make the results easier to see.



Default plots of the fluid velocity (left) and pressure (right).

## Model Wizard

---

Now that the fluid velocity has been computed, add a physics interface and a new study to compute the particle trajectories.

- 1 In the Home toolbar click Add Physics .
- 2 Go to the Add Physics window. In the default configuration, the Add Physics window appears to the right of the Graphics window. In the Add Physics tree under Fluid Flow  expand Particle Tracing  and click Particle Tracing for Fluid Flow .
- 3 Find the Physics interfaces in study subsection. This is located beneath the list of physics interfaces. In the Solve column, deactivate Study 1 by clicking on the check box containing the blue check . The check box will be cleared  indicating that this physics interface is not solved in Study 1. This subsection then looks like the following:

Physics interfaces in study	
Studies	Solve
Study 1	<input type="checkbox"/>

The Particle Tracing for Fluid Flow interface is excluded from the Stationary study because the particles are only solved for in the time domain.

- 4 Click the Add to Component 1  button, which is located above the list of physics interfaces.
- 5 In the Home toolbar click Add Study .
- 6 Go to the Add Study window. By default, this will appear to the right of the Graphics window, covering the Add Physics window. Find the Studies subsection. In the tree under General Studies, click Time Dependent .
- 7 Find the Physics interface in study subsection. In the table, click on the blue check  next to the Laminar Flow interface so that this subsection looks like the following:

Physics interfaces in study	
Physics	Solve
 Laminar Flow	<input type="checkbox"/>
 Particle Tracing for Fluid Flow	<input checked="" type="checkbox"/>

It is not necessary to solve for the Laminar Flow physics in the new Time Dependent study step because it is a stationary field that was already solved for in a previous step.

- 8 Click the Add Study  button.
- 9 In the Home toolbar, again click the Add Physics  and Add Study  buttons to remove the corresponding windows from view.

## Physics Interfaces

---

In this section all of the relevant particle tracing physics features are set up. The default boundary condition, Wall, stops particles that touch a boundary. In the following steps, the drag force and the particle release mechanism are specified.

- 1 In the Model Builder window, under Component 1  click Particle Tracing for Fluid Flow . In the Settings window, locate the Particle Release and Propagation section. From the Formulation list, select Newtonian, ignore inertial terms.

The default Formulation is Newtonian, which is a full inertial treatment of the particles that solves second order differential equations to track them through the fluid. With an inertial treatment, particle trajectories can cross fluid velocity streamlines even if the drag force is the only force in the model, because the particles take a finite amount of time to accelerate in response to a change in the velocity of the surrounding fluid.

When inertial terms are ignored, the particle velocity is prescribed such that the drag force is always in dynamic equilibrium with all other applied forces. This reduces the second order equations for particle motion to first order equations. For the Stokes drag law, the relaxation time of a particle is

$$\tau_p = \frac{\rho_p d_p^2}{18\mu}$$

The Newtonian, ignore inertial terms formulation is appropriate when this time scale is much smaller than the range of solution times you are interested in. This is most likely to be the case for very small particles in a viscous fluid.

- 2 In the Physics toolbar click Domains  and choose Drag Force .
- 3 Select Domain 1 by clicking on any part of the channel in the Graphics window.
- 4 In the Settings window for Drag Force locate the Drag Force section.
  - From the **u** list, choose Velocity field (spf) to take the fluid velocity from the Laminar Flow physics interface solved for in the previous study.
  - From the **μ** list, choose Dynamic viscosity (spf/fp1).

Next, set up the release of particles from the Inlet boundary. The goal is to release particles with greater number density where the magnitude of the velocity field is higher. To do this you use the Density option for the particles' Initial Position.

- 5 In the Physics toolbar click Boundaries  and choose Inlet .
- 6 Select Boundary 23 only. This is the same surface as the Inlet for the Laminar Flow physics interface.

- 7 In the Settings window for Inlet locate the Initial Position section.
  - From the Initial position list, choose Density.
  - In the  $N$  text field, type 3000. This is the total number of released particles.
  - In the  $\rho$  text field, type  $\text{spf} \cdot U$ . The number density of particles at the Inlet will be proportional to the fluid velocity profile.

For the Newtonian, ignore inertial terms formulation, it is not necessary to enter the initial particle velocity, which is defined implicitly from the drag force.
- 8 In the Physics toolbar click Boundaries  and choose Particle Counter .
 

**Note:** The Particle Counter is used to compute the transmission probability of particles. Alternatively, it is possible to compute the transmission probability in postprocessing by assigning a Selection  to the Particle  dataset.
- 9 Select Boundary 20 only. This was the Outlet boundary for the Laminar Flow physics interface.
- 10 In the Settings window for Particle Counter locate the Particle Counter section.
- 11 From the Release feature list, choose Inlet 1. Because this model only includes one inlet, it is not strictly necessary to choose a specific release feature, although doing so can be useful in more complicated models.
- 12 In the Model Builder under Particle Tracing for Fluid Flow, click Particle Properties 1 .
- 13 To change the diameter of the particles, in the Settings window for Particle Properties locate the Particle Properties section. From the  $\rho_p$  list, choose User defined. In the  $d_p$  text field, type  $5E-7$  [m].

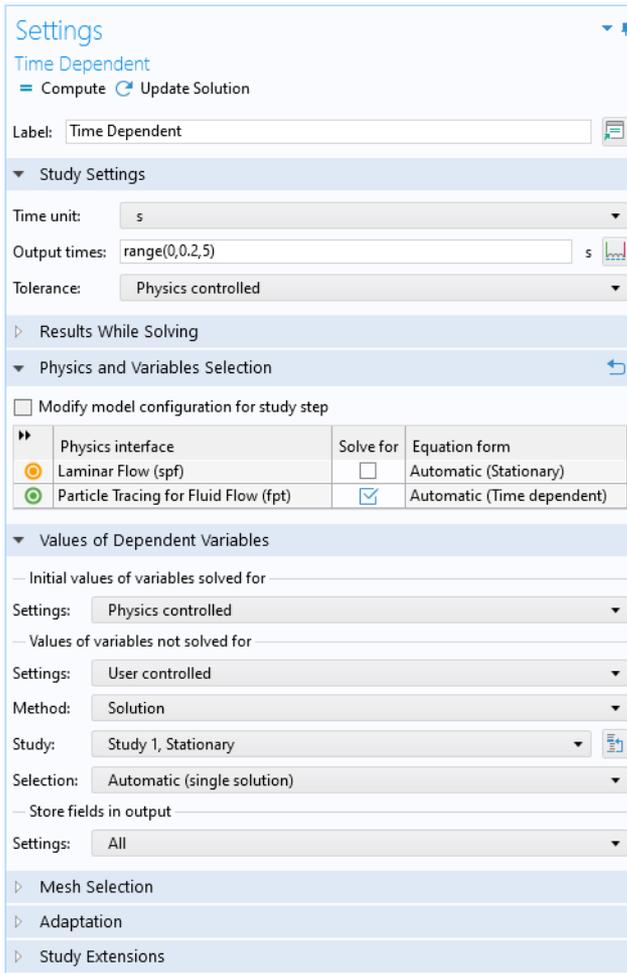
## Study 2

---

- 1 In the Model Builder expand the Study 2 node and click Step 1: Time Dependent .
- 2 Expand the Values of Dependent Variables section. Under Values of variables not solved for, from the Settings list, choose User controlled.
- 3 From the Method list, choose Solution.
- 4 From the Study list, choose Study 1, Stationary.

**Note:** The above settings are very important for coupling the fluid velocity computed in the previous study to the particle tracing analysis. If you forget to specify the Values of variables not solved for, then the initial particle velocity and background fluid velocity will both evaluate to zero everywhere. In this example, the particles would never move as a result.

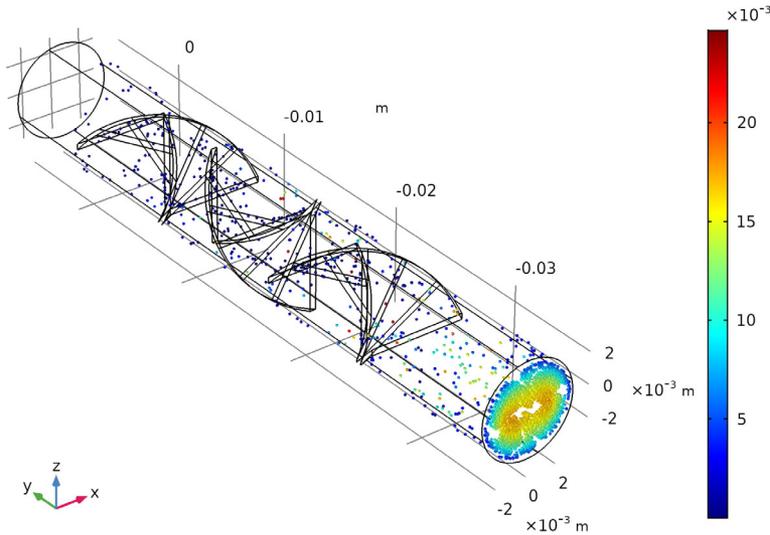
- 5 Locate the Study Settings section. Click the Range button .
- 6 Go to the Range dialog box.
  - In the Step text field, type 0.2.
  - In the Stop text field, type 5.
- 7 Click the Replace button. The Settings window should look like the following screenshot.



- 8 In the Home toolbar click Compute .

## Results

By default you end up in the settings window for the Particle Trajectories (fpt) plot group when the study is complete. Since the default wall condition at the outlet is the Freeze condition, particles are still displayed at the outlet. The color expression of these stopped particles indicates their speed at the time they reached the outlet.



The default Particle Trajectories plot. Particles are colored according to their speed at the final time.

### PARTICLE TRAJECTORIES (FPT)

- 1 In the Settings window for 3D Plot Group, locate the Color Legend section.
- 2 Clear the Show legends check box.
- 3 Click to expand the Title section. From the Title type list, choose None.

#### Particle Trajectories I

- 4 In the Model Builder expand the Results>Particle Trajectories (fpt) node and click Particle Trajectories I.
- 5 In the Settings window for Particle Trajectories, locate the Coloring and Style section.
- 6 Find the Point style subsection. From the Type list, choose Comet tail. The comet tails will show the magnitude and direction of each particle's velocity.

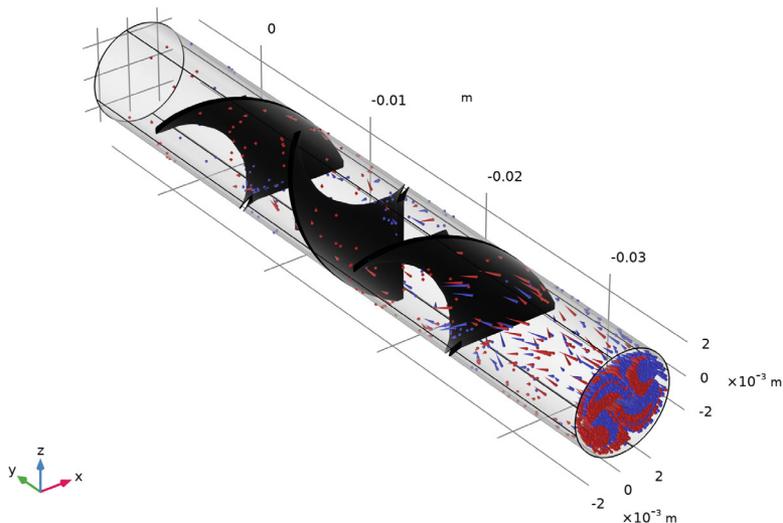
- 7 In the Model Builder window, expand the Particle Trajectories 1  node and click Color Expression 1 .
- 8 In the Settings window for Color Expression, locate the Expression section.
- 9 In the Expression text field, type  $\text{at}(0, \text{qx} < 0)$ .  
The  $\text{at}$  operator evaluates the second input argument at the solution time given by the first. The above expression will return 1 for particles that have a negative initial  $x$ -coordinate, or return 0 otherwise.
- 10 Locate the Coloring and Style section. From the Color table list, choose RainbowLight. This gives a bit more contrast compared to the default Rainbow.

### Surface 1

- 11 In the Particle Trajectories (fpt) toolbar, click  Surface.
- 12 In the Settings window for Surface, locate the Expression section.
- 13 In the Expression field, type 1.
- 14 Locate the Coloring and Style section. From the Coloring list, choose Uniform.
- 15 From the Color list, choose Black.
- 16 From the Particle Trajectories (fpt) toolbar, click  Selection.
- 17 In the Settings window for Selection, locate the Selection section.
- 18 From the Selection list, choose Blade surfaces.

### Surface 2

- 19 In the Particle Trajectories (fpt) toolbar, click  Surface.
- 20 In the Settings window for Surface, locate the Expression section.
- 21 In the Expression field, type 1.
- 22 Locate the Coloring and Style section. From the Coloring list, choose Uniform.
- 23 From the Color list, choose Gray.
- 24 From the Particle Trajectories (fpt) toolbar, click  Selection.
- 25 In the Settings window for Selection, locate the Selection section.
- 26 From the Selection list, choose Outer surfaces.
- 27 In the Model Builder, right-click Surface 2  and choose  Transparency.
- 28 In the Settings window for Transparency, locate the Transparency section.
- 29 In the Fresnel transmittance text field, type 1, or drag the slider to the right.
- 30 Click the Plot button .
- 31 Click Zoom Extents  in the Graphics toolbar. The plot should look like the following figure. Fresnel transmittance makes surfaces look more transparent if they are viewed normally, or more opaque when viewed at an angle.



You can also view the trajectories at different times by selecting other solution times in the Settings window for the Particle Trajectories (fpt)  node. You can also cycle through

In the next section, create a copy of this trajectory plot in which the trajectories are rendered as lines instead of points.

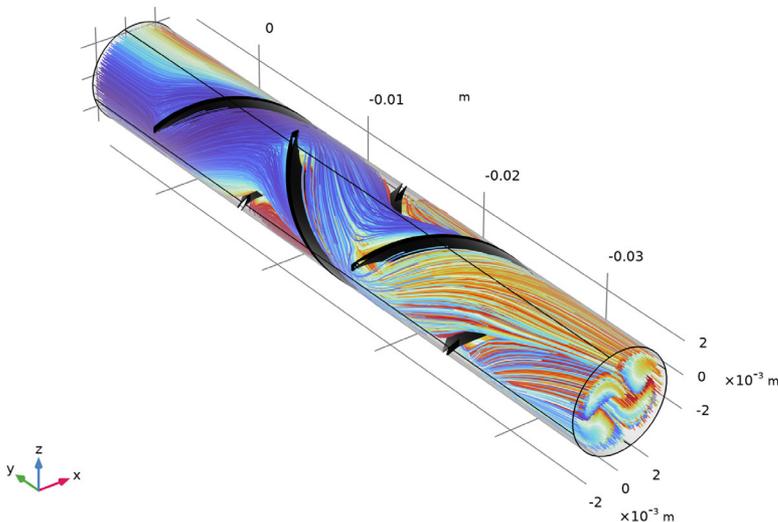
## PARTICLE TRAJECTORIES (FPT) I

- 1 To clean up the Model Builder window, you can collapse the Particle Trajectories (fpt)  node.
- 2 In the Model Builder window, right-click Particle Trajectories (fpt)  and choose Duplicate. This will create a copy called Particle Trajectories (fpt) I .

### *Particle Trajectories I*

- 3 Expand the Results>Particle Trajectories (fpt) I  node and click Particle Trajectories I .
- 4 In the Settings window for Particle Trajectories, locate the Coloring and Style section.
- 5 Find the Point style subsection. From the Type list, choose None.
- 6 Find the Line style subsection. From the Type list, choose Line.

- 7 In the Model Builder window, expand the Particle Trajectories 1  node and click Color Expression 1 .
- 8 In the Settings window for Color Expression, locate the Expression section.
- 9 In the Expression text field, type  $(0, q_x)$ .
- 10 Locate the Coloring and Style section. From the Color table list, choose Dipole.
- 11 Click the Plot button .
- 12 Click Zoom Extents  in the Graphics toolbar.



Now evaluate the transmission probability using the Particle Counter that was created earlier. For this purpose, transmission probability is defined as the fraction of the released particles to reach the outlet by the final time.

- 1 In the Results toolbar click Global Evaluation .
- 2 In the Settings window for Global Evaluation locate the Data section.
  - From the Dataset list, choose Study 2/Solution 2 (sol2).
  - From the Time selection list, choose Last.
- 3 Click Replace Expression  in the upper-right corner of the Expression section. From the menu, choose Model>Component 1>Particle Tracing for Fluid Flow>Particle Counter 1>fpt.pcnt1.alpha – Transmission probability.
- 4 Click the Evaluate button . The transmission probability should be about 0.8.

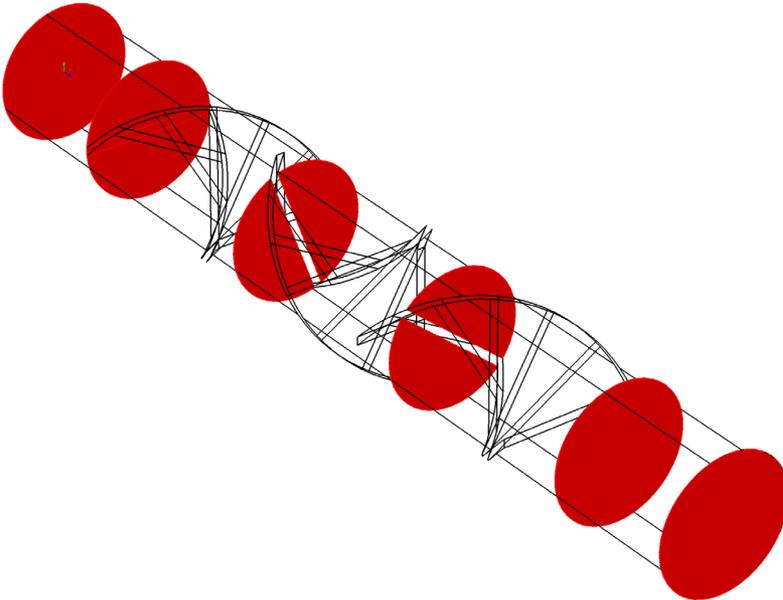
## Poincaré Map

One useful way to visualize how particles mix is to use a Poincaré Map plot. The Poincaré map places a colored dot for each particle at the location at which the particle passes through a cut plane (known as a Poincaré section).

The first step is to define the cut planes where the intersections with the particle trajectories will be taken. These cut planes are defined as datasets.

- 1 In the Results toolbar click Cut Plane .
- 2 In the Settings window for Cut Plane locate the Data section. From the Dataset list, choose Particle 1.
- 3 Locate the Plane Data section.
  - From the Plane list, choose xz-planes.
  - In the y-coordinate text field, type 0.006.
  - Select the Additional parallel planes check box.
  - In the Distances text field, type 0.006 0.016 0.026 0.036 0.042.
- 4 Click the Plot button .

The cut planes will be shown in the Graphics window as shown below.



*Cross-sections at which Poincaré maps will be plotted are shown in red.*

## *Poincaré Map*

- 1 In the Home toolbar click Add Plot Group  and choose 3D Plot Group .
- 2 In the Settings window for 3D Plot Group locate the Data section. From the Dataset list, choose Particle 1.

The plots only display particle positions in each cut plane, but the plot group should still use Particle 1 or one of the Solution datasets as its data source, since this is used to render the dataset edges in the Graphics window.

- 3 On the 3D Plot Group 4 toolbar, click More Volume Plots  and choose Poincaré Map .
- 4 In the Settings window for Poincaré Map locate the Data section. From the Cut plane list, choose Cut Plane 1.
- 5 Locate the Coloring and Style section.
  - Select the Radius scale factor check box.
  - In the associated text field, type 0.4. This scaling factor is multiplied by the Point radius expression to determine the size of each point in the plot.
- 6 Click the Plot button .
- 7 Right-click Poincaré Map 1  and choose Color Expression .
- 8 In the Settings window for Color Expression locate the Expression section. In the Expression text field, type  $at(0, qx < 0)$ .

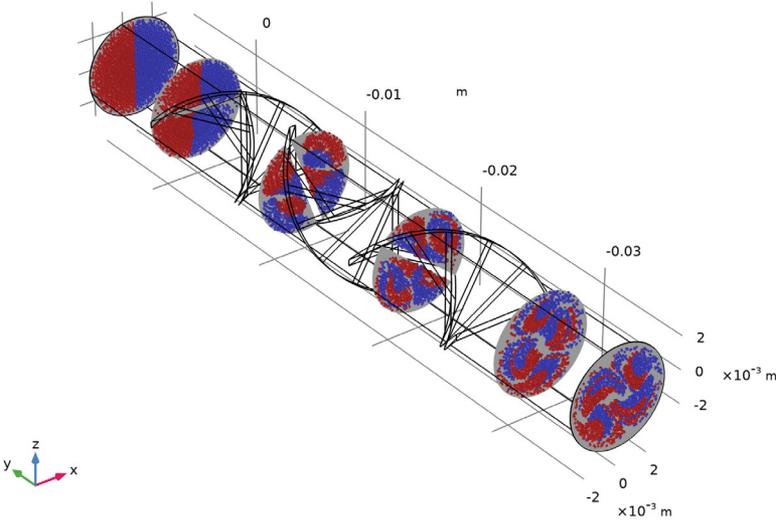
The  $at$  operator evaluates an expression at the time specified in the first argument. Here, it is a logical expression based on the initial particle position.
- 9 Locate the Coloring and Style section.
  - From the Color table list, choose RainbowLight.
  - Clear the Color legend check box.

Next add a Surface plot to render the cut planes where the intersection points are computed.

## *Surface Plot*

- 1 On the 3D Plot Group 4 toolbar click Surface .
- 2 In the Settings window for Surface locate the Data section. From the Dataset list, choose Cut Plane 1.
- 3 Locate the Expression section. In the Expression text field, type 1.
- 4 Locate the Coloring and Style section.
  - From the Coloring list, choose Uniform.
  - From the Color list, choose Gray.
- 5 Click the Plot button .

6 Click the Go to Default 3D View  button in the Graphics toolbar. The plot should look like the following figure.



In the above figure, the location of the particles at 6 Poincaré sections are shown. The color represents the location of the particle at its initial position. So, particles marked as red had an initial position of  $x < 0$  and particles marked as blue had an initial position of  $x > 0$ . The `at()` operator is used to mark the particles with the color of their initial position. The first Poincaré section (the one furthest to the left in the above figure) clearly indicates which particles start with coordinates of  $x < 0$ . As the particles follow the flow field, they begin to mix together. By the end of the mixer, the particles have not mixed completely — there are still significant pockets of only red and only blue particles.

### PHASE PORTRAIT

Another way to visualize particle positions is using the Phase Portrait plot. A Phase Portrait is a 2D plot in which the axes can be arbitrary expressions in terms of the particle position, velocity, or any other variable that can be evaluated on the particles.

- 1 In the Home toolbar click Add Plot Group  and choose 2D Plot Group .
- 2 In the Settings window for 2D Plot Group, type Phase Portrait in the Label text field.

- 3 Locate the Data section. From the Dataset list, choose Particle 1.
- 4 Locate the Plot Settings section. Clear the Plot dataset edges check box.

### *Phase Portrait 1*

- 5 In the Phase Portrait toolbar, click  More Plots and choose Phase Portrait .
- 6 In the Settings window for Phase Portrait, locate the Expression section.
- 7 From the x-axis list, choose Manual. In the Expression text field, type  $qx$ .
- 8 From the y-axis list, choose Manual. In the Expression text field, type  $qz$ .
- 9 Right-click Phase Portrait 1  and choose Color Expression .
- 10 Locate the Expression section. In the Expression text field, type  $at(0, qx < 0)$ .
- 11 In the Settings window for Color Expression, locate the Coloring and Style section.
  - Clear the Color legend check box.
  - From the Color table list, choose Rainbowlight.
- 12 Click the Plot button .

**Note:** By default the Phase portrait plot scales the coordinate axes so that the plot fits in the Graphics window. This is to ensure that the phase portrait is shown clearly even if the two axes correspond to quantities with vastly different orders of magnitude, like position and momentum. In the present case, both axes represent position components, so by switching to a different view, a 1:1 aspect ratio can be enforced.

- 13 In the Model Builder, click the Phase Portrait  plot group.
- 14 Locate the Plot Settings section. From the View list, choose View 2.
- 15 Click the Plot button . Now the plot will be shown with a 1:1 aspect ratio.

