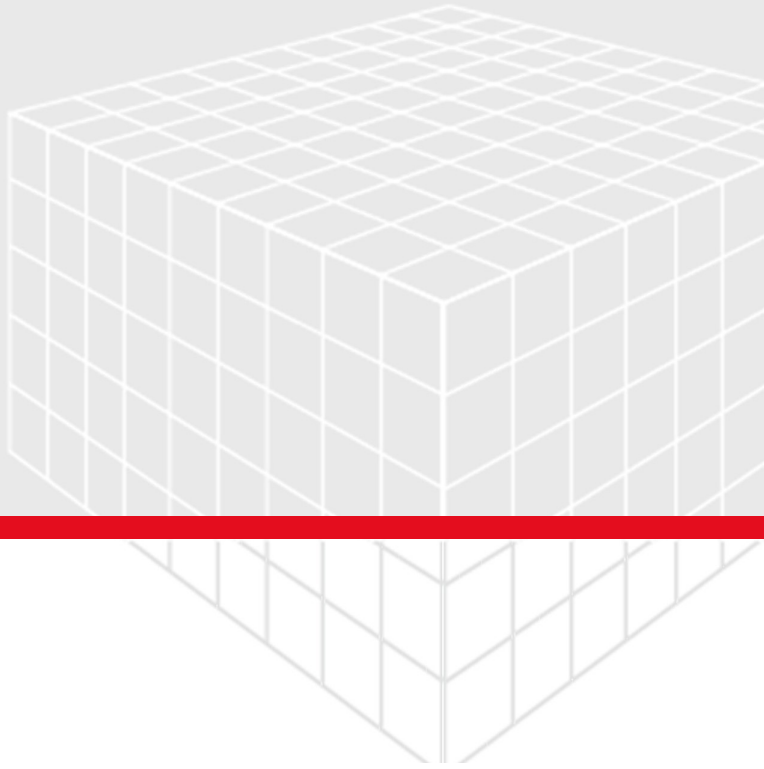
FlowDict

User Guide



GEODICT Release 2026

Published: July 2026

GeoDict

https://doi.org/10.30423/userguide.geodict

FlowDict

1. FlowDict 3

1.1 Theoretical Background 7

1.1.1 Equations of Motion 7

1.1.2 Flow Solvers 12

1.1.3 Post-Processing and Permeability 18

1.2 Stokes(-Brinkman) 23

1.2.1 Options 23

1.2.2 Results 50

1.3 Navier-Stokes(-Brinkman) 58

1.3.1 Options 58

1.3.2 Results 84

1.4 Forchheimer Approximation 93

1.4.1 Options 94

1.4.2 Results 96

1.5 Simulate Flow Experiment 100

1.5.1 Options 101

1.5.2 Results 116

1.6 Darcy Flow 122

1.6.1 Options 122

1.6.2 Results 140

1.7 Single-Phase Flow GeoApps 146

1.7.1 Air Permeability Test 146

1.8 References 149

1 FlowDict

FlowDict computes digital flow experiments and post-processes the results to calculate effective material properties such as permeability.

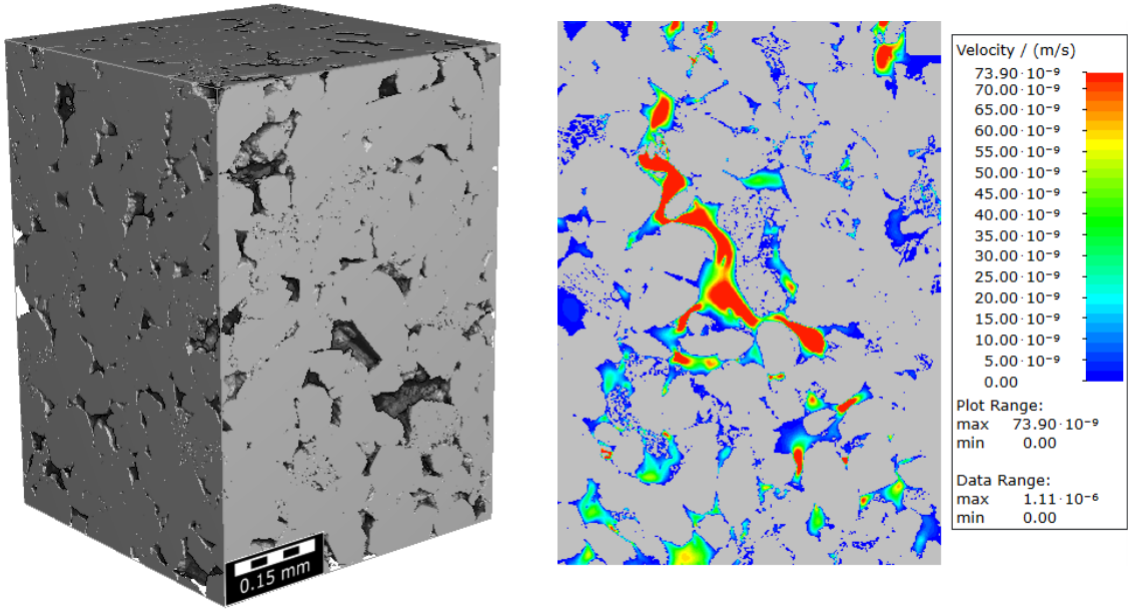
In general, three different use cases are possible:

Determine effective bulk material properties

In this use case, you start with a 3D structure model of a porous material, e.g., a rock, a composite, or a foam. The model shows a representative inner part of the material, but the real material is much larger in all three space directions.

You can use FlowDict to determine the effective permeability of the material.

For this purpose, use the Stokes command 23 if the major through-pores are resolved in the voxel grid, or the Darcy Flow command 122 if all through-pores are unresolved.



Determine pressure drop and flow rate

In this use case, you start with a 3D structure model of a porous layer, e.g., a nonwoven filter, a porous transport layer, or a metal wire mesh. The model shows the entire layer thickness, plus some inflow and outflow area above and below the layer.

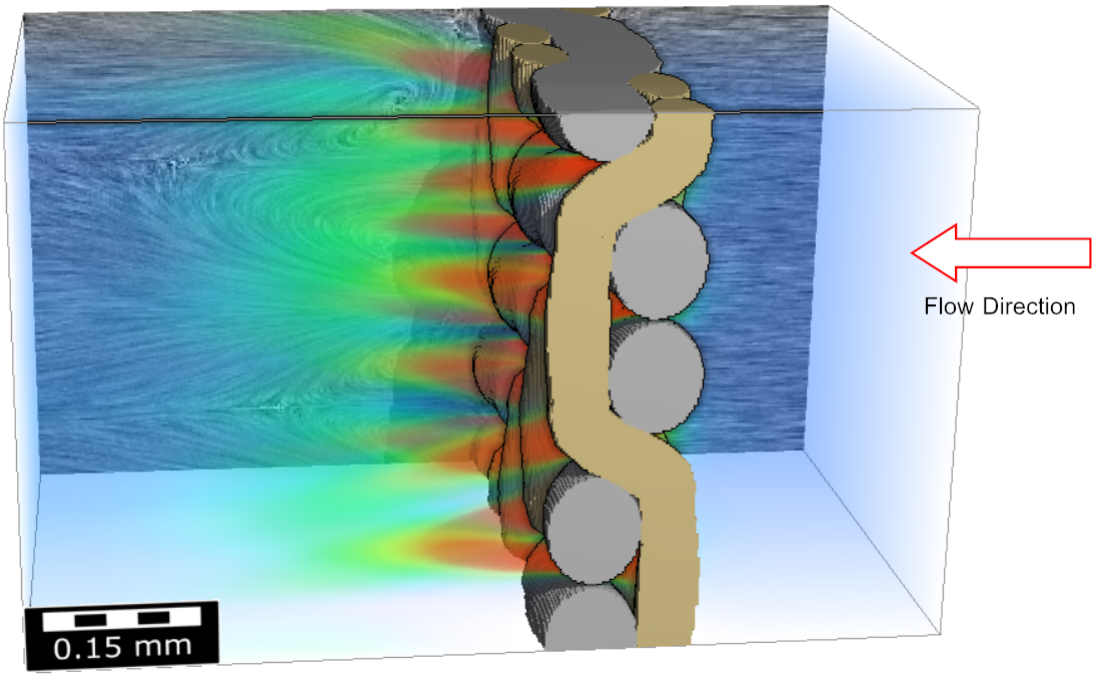
You can use FlowDict to

- predict the volumetric flow rate and mean flow velocity for a given pressure drop, or
- predict the pressure drop for a given mean flow velocity or volumetric flow rate.

For this purpose, use the Navier-Stokes command 58 to prescribe the boundary conditions, compute the stationary flow field, and determine the result. For very high flow velocities, the solution may become turbulent and a stationary solution no longer

exists. In this situation, you can use the Forchheimer Approximation command 93 to extrapolate a result.

You can also use the Air Permeability Test app 146 to simulate standardized air permeability tests with predefined test setups.



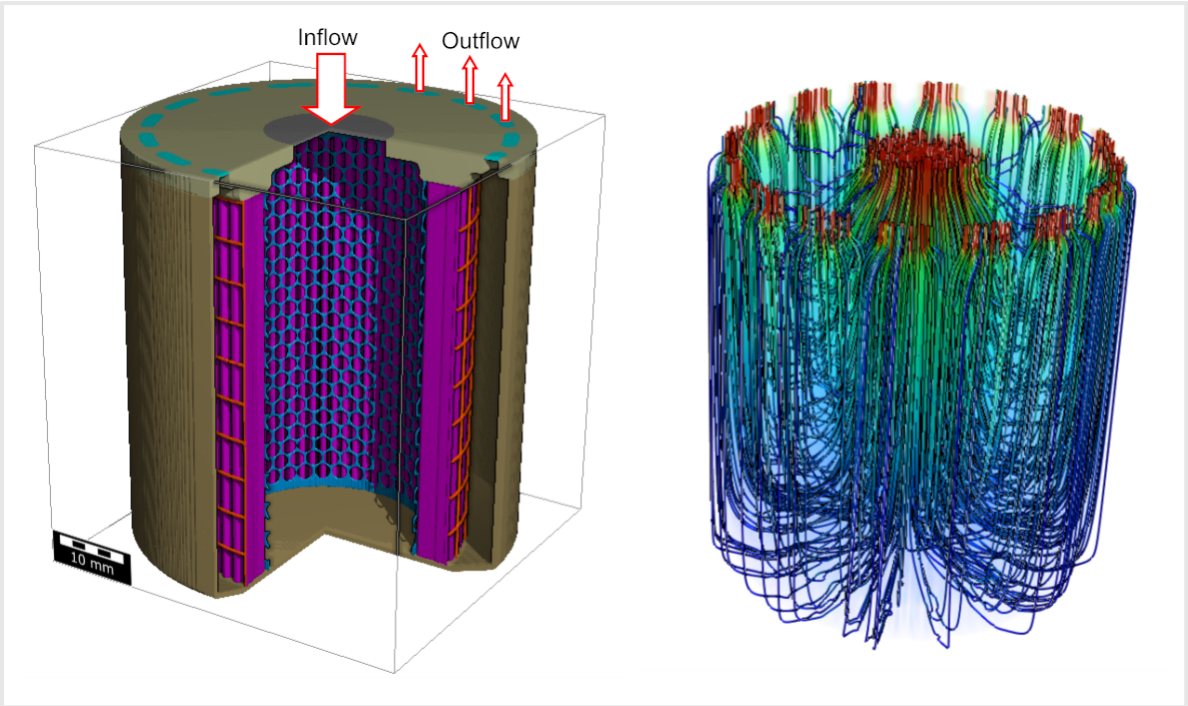
Simulate fluid flow in CAD data

In this use case, you start with a 3D structure model of, e.g., a complete filter including the filter housing or a packed bed reactor including casing and pipes. The 3D structure is typically designed with CAD software tools, and the inner pore structure of the porous medium is often unresolved.

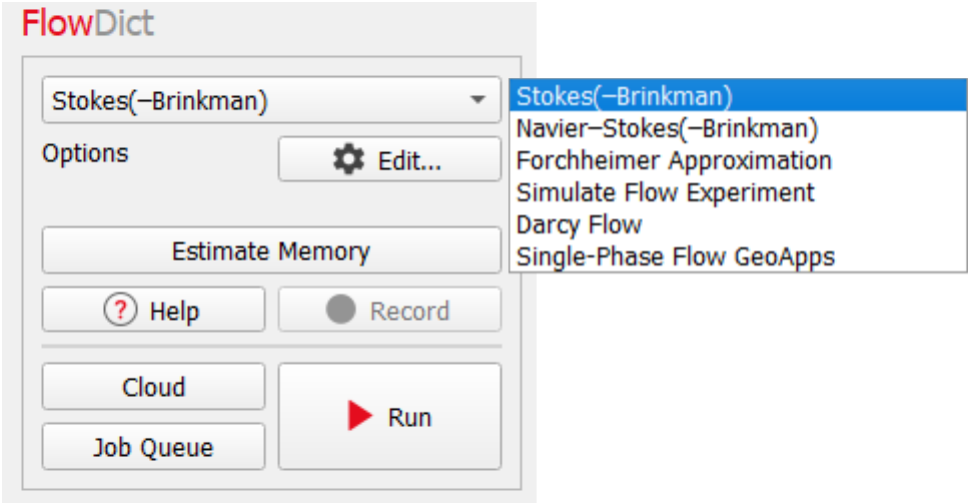
You can use FlowDict to

- predict the volumetric flow rate for a given pressure drop, or
- predict the pressure drop for a given volumetric flow rate.

For this purpose, use the Simulate Flow Experiment command 100 to prescribe the boundary conditions, compute the stationary flow field and determine the result.



In the FlowDict section, the available commands are listed in the pull-down menu.



When clicking **Estimate Memory**, FlowDict estimates the needed memory 16 for the computations based on the size of the structure and the parameters entered in the solver options. The estimated computational memory requirements are shown in a **Memory & Disk-Space** pop-up message.

Learn to use FlowDict

- Theoretical Background 7
  Learn which boundary value problems FlowDict can solve, how the results are post-processed to derive averaged properties, and which solvers are available for these tasks.
- Stokes(-Brinkman) 23
  Simulate fluids with high viscosity and slow, laminar flows.
- Navier-Stokes(-Brinkman) 58

Learn to use FlowDict

Simulate high velocity flow with non-linear effects.

- Forchheimer Approximation 93
  Estimate pressure drop or mean velocities for high Reynolds numbers.
- Simulate Flow Experiment 100
  Simulate flow for inlets and outlets that are not axis-aligned.
- Darcy Flow 122
  Simulate flow for large voxel lengths and porous materials with small permeabilities.
- Single-Phase Flow GeoApps 146
  Choose from the available FlowDict GeoApps.
- References 149
  See the references for more relevant literature.

Join the conversation

Visit the Community on GeoDict Forum to be inspired and get answers to top questions.
Find illustrative application examples on the website.
Our tutorials and seminar videos give you a first hands-on experience. They are collected in the Learning Center.
Send your Support Request to our technical Support.

1.1 Theoretical Background

FlowDict solves a boundary value problem consisting of

- domain boundary conditions that set up the experimental conditions,
- an equation of motion that describes stationary fluid flow in the pores, and
- boundary conditions at the pore walls.

The solution of this boundary value problem is then analyzed and post-processed. FlowDict derives material properties and averaged flow characteristics from the computed 3D result fields.

To solve the partial differential equations, three different flow solvers are available: EJ, SimpleFFT, and LIR. Each of the solvers uses a different numerical approach and thus has different advantages.

Theoretical Background

- Equations of Motion 7
  Learn how to select the equation of motion that needs to be solved.
- Flow Solvers 12
  Solve the Partial Differential Equations using one of three available solvers.
- Post-Processing and Permeability 18
  Understand how FlowDict derives material properties and averaged flow characteristics from the computed 3D result fields

1.1.1 Equations of Motion

FlowDict computes the fluid flow through the 3D structure model.

All experiments in FlowDict assume a steady flow regime, i.e., do not allow for time-dependent behavior such as turbulence. In practice, this means that the velocity and pressure drop cannot be arbitrarily high. FlowDict therefore searches for a stationary solution and the partial differential equations that need to be solved do not contain derivatives regarding time.

Other assumptions are that the fluid is a Newtonian fluid and it is incompressible.

Navier-Stokes Equations

If the geometry of the pores is resolved by the computational grid, all voxels are either pore space or solid material. The fluid flow in the pores is then described by the stationary Navier-Stokes equations:

$$-\mu\Delta\vec{u} + \rho(\vec{u}\cdot\nabla)\vec{u} + \nabla p = \vec{f} \quad (1)$$

Momentum balance

$$\nabla \cdot \vec{u} = 0$$ (2) Mass conservation

Here, ρ is the fluid density, μ is the dynamic viscosity, $\vec{u}$ is the velocity vector, p is the pressure and $\vec{f}$ is a force density.

The Navier-Stokes momentum balance equation includes inertia. For such flows, the resulting relation between pressure drop and mean velocity is nonlinear.

Stokes Equations

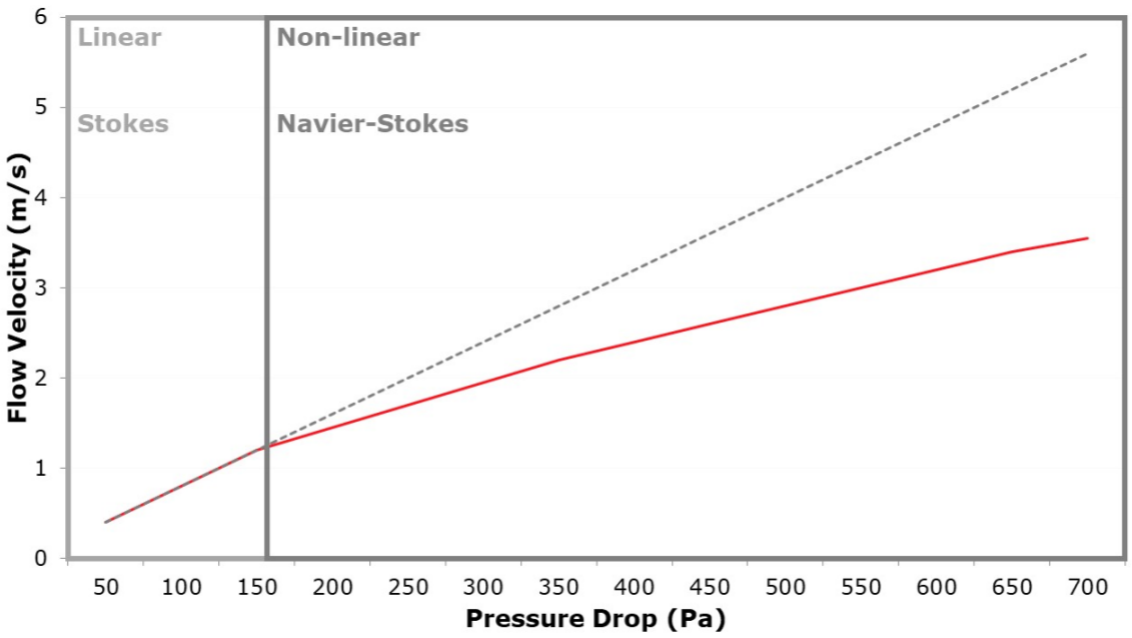
When the flow is slow and laminar, the Navier-Stokes[7] equations can be simplified by neglecting the term that describes inertial effects. The resulting equations are the Stokes equations, which read as follows:

$$-\mu\Delta\vec{u} + \nabla p = \vec{f} \quad (3)$$

$$\nabla \cdot \vec{u} = 0 \quad (4)$$

Here, μ is the dynamic viscosity, $\vec{u}$ is the velocity vector, p is the pressure and $\vec{f}$ is a force density.

For such flows, the resulting relation between pressure drop and mean velocity is linear and Darcy's law can be applied. The sketch below illustrates this behavior: when the pressure drop is small, Stokes and Navier-Stokes equation give identical results, and the flow velocity depends linearly on the applied pressure drop. When the pressure drop gets larger, the results of Stokes and Navier-Stokes differ - in this case, only the Navier-Stokes equation describes the physically correct solution and the velocity depends non-linear on the applied pressure drop.



Navier-Stokes-Brinkman Equations

If the geometry of the pores is not fully resolved by the computational grid, a voxel may describe pore space, solid material, or porous material. An additional term, called *Brinkman* term, allows to describe a Darcy-type flow in an unresolved porous medium. The Navier-Stokes-Brinkman equations are given as follows:

$$-\mu\Delta\vec{u} + \rho(\vec{u}\cdot\nabla)\vec{u} + \sigma\vec{u} + \nabla p = \vec{f} \tag{5}$$

$$\nabla\cdot\vec{u} = 0 \tag{6}$$

Here, $\vec{u}$ is the velocity vector, p is the pressure, $\vec{f}$ is a force density, and μ is the dynamic viscosity. The local viscous flow resistivity σ of a voxel is related to the local permeability by

$$\sigma = \frac{\mu}{\kappa}, \tag{7}$$

where κ is the isotropic permeability of the voxel. In the pore space, the flow resistivity is zero and the Brinkman term disappears.

Stokes-Brinkman Equations

Under the assumption of a slow and laminar flow, the Navier-Stokes-Brinkman

equations can be simplified by neglecting the term that describes inertial effects. The resulting equations read as follows:

$$-\mu\Delta\vec{u} + \sigma\vec{u} + \nabla p = \vec{f} \tag{8}$$

$$\nabla \cdot \vec{u} = 0 \tag{9}$$

Here, $\vec{u}$ is the velocity vector, p is the pressure, $\vec{f}$ is a force density, and μ is the dynamic viscosity. The local viscous flow resistivity σ of a voxel is related to the local permeability by

$$\sigma = \frac{\mu}{\kappa}, \tag{10}$$

where κ is the isotropic permeability of the voxel. In the pore space, the flow resistivity is zero and the Brinkman term disappears.

Darcy Flow

When we analyze the asymptotic behavior of the terms of the Stokes-Brinkman[9] equation, we find that

$$\mu\Delta\vec{u} \sim \frac{\mu}{h^2} \tag{11}$$

Viscous term

$$\frac{\mu}{\kappa}\vec{u} \sim \frac{\mu}{\kappa} \tag{12}$$

Brinkman term

where h is the voxel length.

Therefore, when the permeability κ is small, and the voxel length h is relatively large, the Brinkman term dominates, and the viscous term can be neglected. This leads to

$$\vec{u} = -\frac{\kappa}{\mu}(\nabla p - \vec{f}) \tag{13}$$

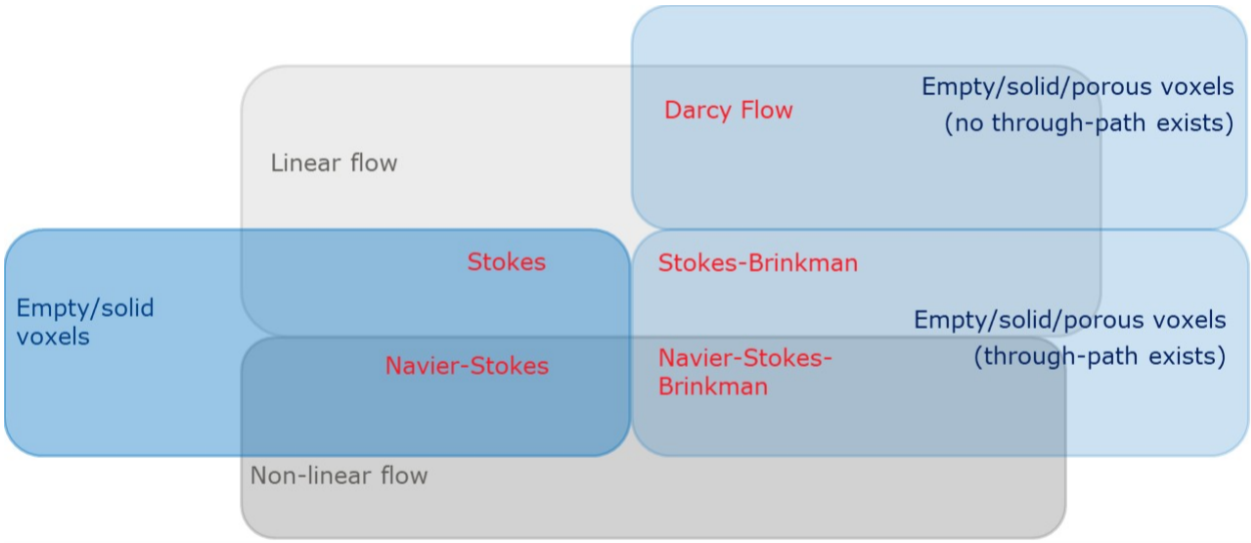
Darcy flow

Numerically, finding a solution for the Darcy flow equation is much faster that finding a solution for the Stokes-Brinkman or Navier-Stokes-Brinkman equation. However, this simplified equation only holds true when the assumptions used in its derivation it are valid:

1. Creeping flow, i.e., very small Reynolds number.
2. Porous voxels with small permeabilities dominate, all through pores have diameters smaller than the voxel length.

The following sketch illustrates which equation must be solved. The decision is based on the points:

- Creeping flow or fast flow (gray boxes): In the creeping flow case, the Reynolds number is small, inertia can be neglected, and a linear relationship between pressure and velocity exists. In the fast flow case, the Reynolds number is large, inertial effects cannot be neglected, and a non-linear relationship between pressure and velocity exists.
- Resolution (blue boxes): If all pores are resolved, each voxel either represents fluid or solid material. If some pores are unresolved, a voxel may also represent porous material. In the extreme case, all through-pores are unresolved.



All five derived momentum balance equations are shown in the table below:

$-\mu\Delta\vec{u}$			$+\nabla p = \vec{f}$	Stokes momentum balance
$-\mu\Delta\vec{u}$		$+\mu\kappa^{-1}\vec{u}$	$+\nabla p = \vec{f}$	Stokes-Brinkman momentum balance

$-\mu\Delta\vec{u}$	$+(\rho\vec{u}\cdot\nabla)\vec{u}$		$+\nabla p=\vec{f}$	Navier-Stokes momentum balance
$-\mu\Delta\vec{u}$	$+(\rho\vec{u}\cdot\nabla)\vec{u}$	$+\mu\kappa^{-1}\vec{u}$	$+\nabla p=\vec{f}$	Navier-Stokes-Brinkman momentum balance
		$+\mu\kappa^{-1}\vec{u}$	$+\nabla p=\vec{f}$	Darcy Flow

1.1.2 Flow Solvers

To solve the partial differential equations[7] describing the fluid flow, three solution methods, called solvers, are available in FlowDict:

LIR

The **LIR** (Left Identity Right) solver uses a non-uniform adaptive grid which results in very low memory requirements. While the LIR solver's speed is comparable to the **SimpleFFT** solver for low porous materials, it is very fast (and should be chosen) for highly porous materials. See Linden et al. (2015)[149] for more information.

It can be used to solve the Stokes[8], Darcy Flow[10], Stokes-Brinkman[9], Navier-Stokes[7], and Navier-Stokes-Brinkman[9] equations.

SimpleFFT

The **SimpleFFT** (Simple Fast Fourier Transform) solver takes longer to converge for highly porous materials and requires more memory, but it can deal with non-linear (non-laminar) fluid flow as well as with linear flow. Therefore, faster flows, which are not necessarily laminar and are modeled by the Navier-Stokes[7] equation can be computed with the **SimpleFFT** solver.

It can be used to solve the Stokes[8], Stokes-Brinkman[9], Navier-Stokes[7], and Navier-Stokes-Brinkman[9] equations

The **SimpleFFT** solver is very fast for low porosity materials and is a much better choice than the **EJ** solver for this type of materials.

EJ

The **EJ** (Explicit Jump) solver is very fast for highly porous materials and has low memory requirements but can only be applied to flows when the pressure drop/flow velocity dependence is linear (laminar), i.e., for the Stokes 8 equation and Darcy Flow 10 only.

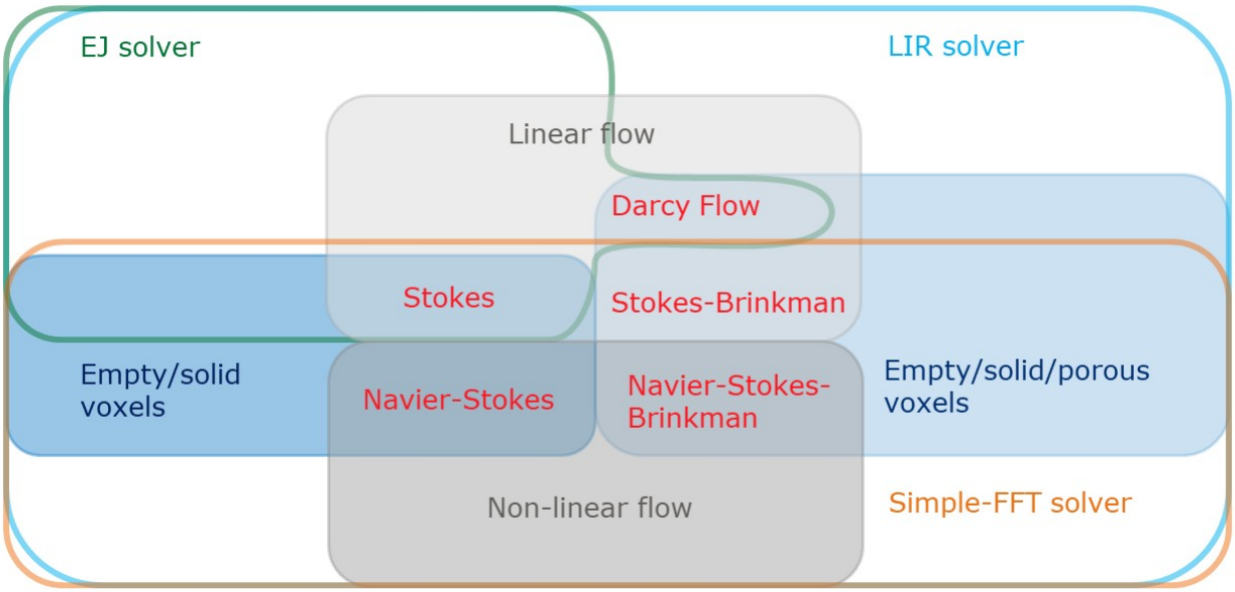
For Stokes simulations, the **EJ** solver should not be used for low porous materials (50% or less porosity) such as reservoir rocks or ceramics. For these kinds of materials, the **SimpleFFT** or **LIR** solver perform much better.

Know how! It is important to distinguish between *porous materials* and *porous voxels*.

A flow simulation is run on a 3D material model, e.g., glass fibers, where the pores are filled with air. Thus, a highly **porous material** is a material with more fluid voxels than solid voxels, e.g., 80% air and 20% glass fibers.

Alternatively, a voxel, i.e. a grid cell, can be defined as **porous voxel**, if the pores in a material are too small to be resolved by the computational grid.

The sketch below shows which solver can be applied in which use case. The equations are denoted in red. The gray boxes show where the solution exhibits a linear relation between pressure drop and flow rate, and the blue boxes show where the 3D structure can have porous voxels:



Solver Benchmarks

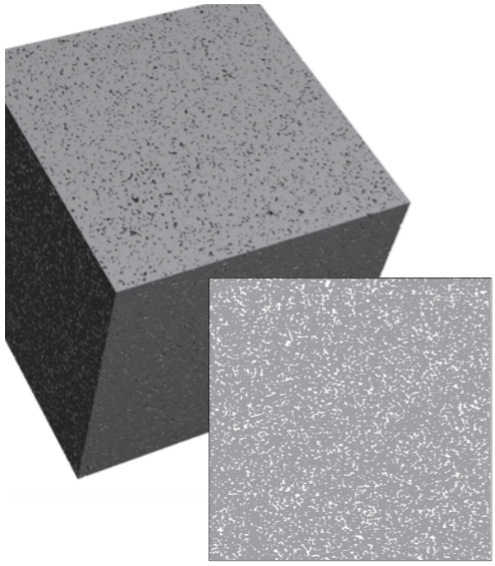
Three parallelization benchmark results for FlowDict are shown here. All benchmark computations were run on a server with 2 x Intel E5-2697A v4 processors with 16 cores each, running with a maximum of 3.60 GHz, and 1,024 GB RAM.

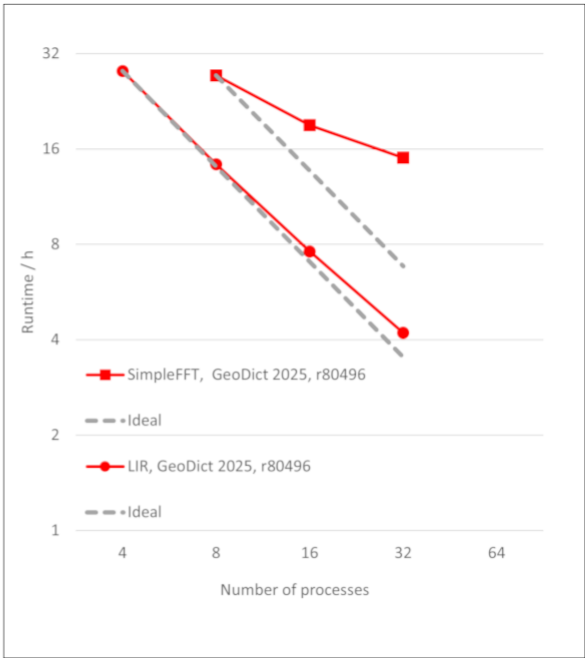
Sandstone Benchmark

The first example is the computation of the flow through a synthetic X-ray tomography data of a sandstone structure (Mattila et al., 2016[149]). The structure has a porosity of 13.5% and a size of 2,048 x 2,048 x 2,032 voxels.

Together with an inlet and outlet of 8 voxels each, the flow computation is performed on a structure size of $2{,}048^3$. This structure size is optimal to speed up runtime, since 2,048 is a power of two. The smaller the factors in the prime factorization of the number of voxels in each coordinate direction, the better for the runtime of SimpleFFT[12].

The flow through the structure is computed by solving the Stokes[8] equation with the LIR[12] solver (speed optimization and memory optimization[46]) and with the SimpleFFT[12] solver. The following figure shows the runtime for a different number of processes to compute the permeability of $3.3e^{-13}$ m^2 with an error bound of 0.01 with both solvers. Runtime and speedup of the parallelization are better for the LIR[12] solver. For both solvers, the ideal speedup, i.e., getting half of the runtime for twice of the number of processes is also shown. The computation with the LIR[12] solver is close to the ideal speedup, which is usually not possible to reach for real life examples.





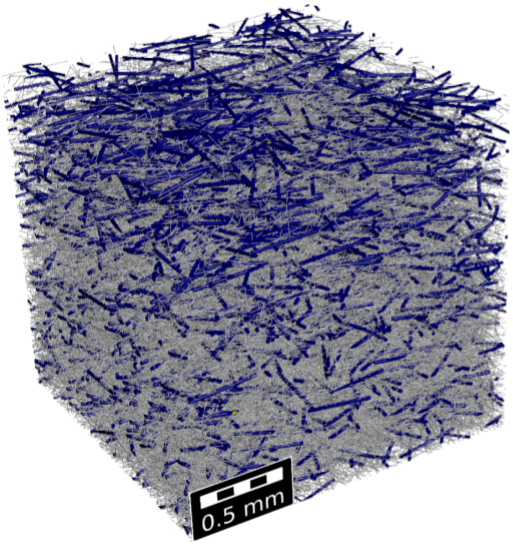
Filter Benchmark

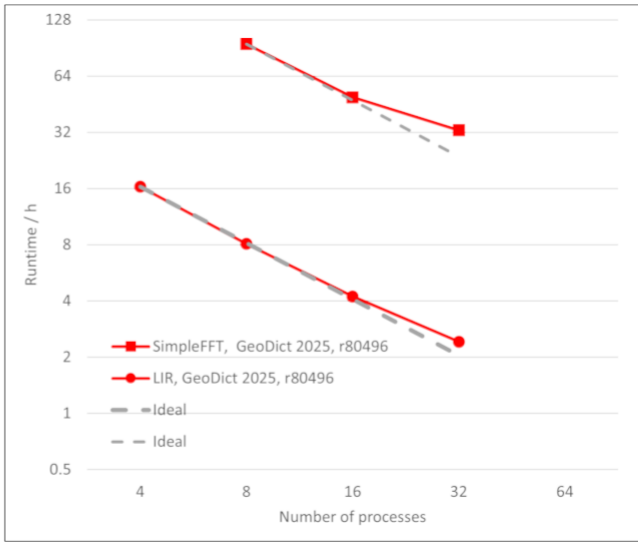
The second benchmark is the computation of the flow through a filter structure with linear increasingly fiber density.

The structure size is 2,048 x 2,048 x 2,032 voxels, with an inlet and an outlet of 8 voxels (i.e., in total 8.6 billion voxels). So, like for the sandstone structure, the size is optimal to get a short runtime even for a large structure. The porosity of 94.2% is much larger than for the sandstone structure example.

The flow through the structure is computed by solving the Stokes[8] equation, with the LIR[12] (speed optimization[46]) and the SimpleFFT[12] solver.

The following figure shows the runtime for a different number of processes to compute the permeability of $5.63e^{-11}$ m^2 for SimpleFFT and $5.69e^{-11}$ m^2 for LIR[12]. Both results are equal within the chosen error bound of 1%. The benefit of using the LIR[12] solver compared to SimpleFFT[12] is large for the structure with high porosity. The computation with the LIR[12] solver is close to the ideal speedup, which is usually not possible to reach for real life examples.



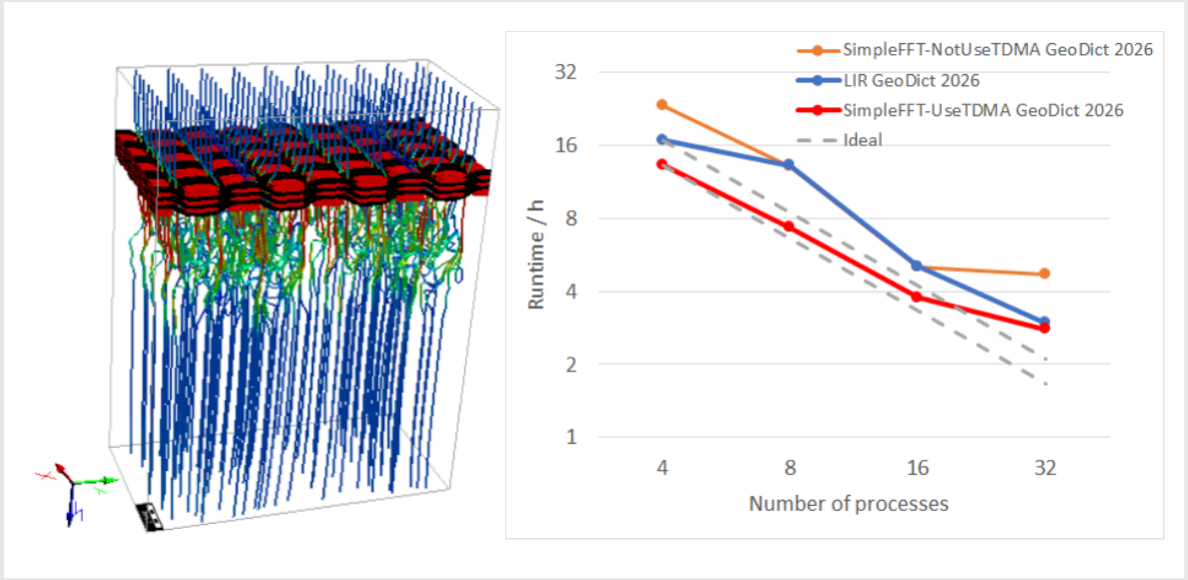


Multilayer Weave Benchmark

The third benchmark is the computation of fast airflow through a multilayer weave. The structure has a size of 400 x 800 x 1024 including 150 voxels inlet and 800 voxels outlet (i.e., in total ~300 million voxels).

The flow through the structure is computed by solving the Navier-Stokes[7] equation, with the LIR[12] (speed optimization[46]) and the SimpleFFT[12] solver with and without using Tdma[43]. The Velocity Inlet, Pressure outlet[28] boundary condition is used and the average flow velocity is 3 m/s.

The SimpleFFT[12] solver with using Tdma[43] reduces the runtime in this benchmark by up to 50%.



Memory requirements for the examples and both solvers are listed in the following table:

Benchmark	RAM used (LIR)	RAM used (SimpleFFT with 32 Processes)
Sandstone	149 GB (Speed Optimization 46) 103 GB (Memory Optimization 46)	655 GB
Filter	184 GB	654 GB
Multilayer Weave	6.2 GB	35.1 GB (not using Tdma 43) 35.0 GB (using Tdma 43)

Memory Requirements

The RAM required for a flow computation depends on the set of flow equations, structure size, porosity, boundary conditions, parallelization, and the selected solver itself. Here, basic hints are given to estimate the memory requirements for the EJ 13 solver, the SimpleFFT 12 solver, and the LIR 12 solver.

We assume a structure with

1. NX, the voxels in X-direction,
2. NY, the voxels in Y-direction,
3. NZ, the voxels in Z-direction,
4. NI, additional inlet voxel layers,

5. NO, additional outlet voxel layers, and
6. P, the number of parallel processes

In addition, we assume that a flow in Z-direction is computed. Then, the total number of voxels is given by

$$\mathrm{NumberOfVoxels} = \mathrm{NX} \cdot \mathrm{NY} \cdot (\mathrm{NZ} + \mathrm{NI} + \mathrm{NO}) \quad (14)$$

The EJ[13] and SimpleFFT[12] solver are MPI-parallel and need to have some overlap slices for each process in memory. The number of voxels including overlap slices is then given by

$$\mathrm{NumberOfVoxelsWithOverlap} = \mathrm{NX} \cdot \mathrm{NY} \cdot (\mathrm{NZ} + \mathrm{NI} + \mathrm{NO} + 2 \cdot \mathrm{P}) \quad (15)$$

The memory requirements can now be roughly estimated using the following formulas for each of the solvers:

Memory estimation for EJ

The memory required by the EJ[13] solver depends on the total number of voxels, the boundary conditions, the number of pore-solid interfaces and the number of parallel processes. The number of pore-solid interfaces depends on the solid volume fraction (SVF). A rule of thumb is:

$$\mathrm{RequiredMemory} = 60 \cdot \mathrm{NumberOfVoxelsWithOverlap} \quad (16)$$

A more accurate rule of thumb that considers the SVF is:

$$\mathrm{RequiredMemory} = (-132 \cdot \mathrm{SVF}^2 + 166 \cdot \mathrm{SVF} + 30) \cdot \mathrm{NumberOfVoxelsWithOverlap} \quad (17)$$

For symmetric boundary conditions in flow direction the memory requirements are doubled. The unit of the estimated memory is in bytes.

Memory estimation for SimpleFFT

The memory required by the SimpleFFT[12] solver depends on the total number of voxels, the boundary conditions, and the number of parallel processes. A rule of thumb is:

$$\mathrm{RequiredMemory} = 80 \cdot \mathrm{NumberOfVoxelsWithOverlap} \quad (18)$$

For symmetric boundary conditions in flow direction the memory requirements are doubled. The unit of the estimated memory is in bytes.

Memory estimation for LIR

The estimation of the LIR[12] memory requirements is more complicated due to the adaptive grid structure. The memory that is required for a flow computation depends on the set of equations (Stokes[8] or Stokes-Brinkman[9]), the total number of voxels, and the solid volume fraction (SVF). The memory requirements are directly correlated to the number of adaptive grid cells. The number of adaptive grid cells depends on the number of pore-solid interfaces. A rule of thumb is:

$$\mathrm{RequiredMemory} = 5 \cdot 10^7 + \mathrm{NumberOfCells} \cdot (63 + 44 \cdot \mathrm{OptimizeForSpeed} + 8 \cdot \mathrm{WithBrinkman}) \quad (19)$$

where

$$\mathrm{NumberOfCells} = \mathrm{NumberOfVoxels} / \left(60 \cdot \mathrm{SVF}^2 - 56 \cdot \mathrm{SVF} + 15\right) \quad (20)$$

$$\mathrm{OptimizeForSpeed} = 1 \quad (21)$$

if **Optimize** is set for **Speed**, else the value is 0, and

$$\mathrm{WithBrinkman} = 1 \quad (22)$$

if Stokes-Brinkman[9] or Navier-Stokes-Brinkman[9] is computed, otherwise the value is 0.

1.1.3 Post-Processing and Permeability

FlowDict solves the partial differential equation[7] for the defined boundary values with the selected flow solver[12]. The direct result of the computation is a 3D data field consisting of a vector-valued velocity and a scalar-valued pressure for each fluid or porous voxel that describes the resulting steady-state flow.

This field can be visualized to illustrate the flow. Additionally, a number of quantities describing the flow characteristics or the material properties are derived from the result:

Average (superficial) flow velocity and pressure drop

For each layer in the selected computational direction[27], the average pressure and the average flow velocity in the computational direction are computed. One of these quantities is typically fixed by the selected boundary conditions:

- If the pressure drop is set as boundary condition, the average flow velocity is reported as result.
- If the mean flow velocity or the flow rate is set as boundary condition, the pressure drop (the pressure difference between top and bottom layer) is reported as result.

There are two possibilities to compute an average flow velocity. The *superficial flow velocity* reported here averages over all grid cells, including the solid cells which are included in the averaging using a velocity of 0.0. The *interstitial flow velocity* averages only over the grid cells that contain a fluid. This velocity is found in a plot in the result file.

Reynolds number

The Reynolds number is a dimensionless value used to predict flow patterns. It represents the ratio of inertial forces to viscous forces within a fluid. The Reynolds number is an indicator if the flow is

- slow and linear (Stokes), or
- fast and non-linear (Navier Stokes), or
- turbulent (no stationary solution exists).

The Reynolds number (Re) is defined as

$$\mathrm{Re} = \frac{\rho u L}{\mu} \quad (23)$$

Reynolds Number

where ρ is the density of the fluid (SI units: kg/m^3), u is the velocity of the fluid (m/s), μ is the dynamic viscosity of the fluid (Pa·s or N·s/m^2 or kg/m·s), and L is a characteristic length (m). Depending on the different types of porous media and different applications, there are different choices for characteristic length. Four options are provided:

1. **Numerical Length:** The square root of the permeability is taken as characteristic length, $L = \sqrt{\kappa}$ (Zheng and Grigg, 2006[149]). This option is used for Stokes[8] by default.
2. **Maximum Pore Diameter:** The characteristic length is the diameter of the largest through pore. From structure analysis the maximum pore diameter is found. This option is used for Navier-Stokes[7] by default.
3. **Maximum Fiber/Particle Diameter:** The characteristic length is the largest fiber diameter for fibrous media and the largest particle size for grain structures. The largest fiber/particle diameter is found by an analysis of the 3D structure.
4. **Manually specified:** When none of the previous three options fits, you can set a characteristic length manually.

Permeability

If the flow is slow and laminar, a linear relationship between the average flow velocity and the pressure drop is observed. This is known as Darcy's law

$$\bar{u} = \frac{\kappa}{\mu}\frac{\Delta p}{L} \quad (24)$$

Darcy's law (1D)

where $\bar{u}$ is the average (superficial) flow velocity, κ is the permeability, μ is the dynamic fluid viscosity, Δp is the pressure drop, and L is the thickness of the media.

The resulting permeability is independent of the applied pressure drop, as well as from the used fluid viscosity. Thus, the permeability is considered a material property. For isotropic materials, the permeability is a scalar value. For anisotropic materials, the permeability depends on the flow direction and can be described as a 3x3 tensor as described below.

Important! The permeability can only be determined from the solution of the Stokes equation. For fast flows, Darcy's law no longer holds true and the relation between pressure drop and flow velocity is no longer described by the material's permeability alone.

Flow permeability tensor

The structure's permeability tensor is found from the three-dimensional version of Darcy's law:

$$\bar{u}^{\hat{i}} = -\frac{1}{\mu}\begin{bmatrix} \mathrm{K}_{xx} & \mathrm{K}_{xy} & \mathrm{K}_{xz} \\ \mathrm{K}_{yx} & \mathrm{K}_{yy} & \mathrm{K}_{yz} \\ \mathrm{K}_{zx} & \mathrm{K}_{zy} & \mathrm{K}_{zz} \end{bmatrix} \nabla p^i \quad (25)$$

Darcy's law (3D)

Here $\bar{u}^i$ is the averaged velocity vector (averaged flux with i=1 corresponding to a pressure drop in the **X**-direction, i=2 corresponding to a pressure drop in the **Y**-direction and i=3 corresponding to a pressure drop in the **Z**-direction). μ denotes the fluid viscosity, and

$$\nabla p^i = \frac{\Delta p^i}{L^i} \quad \text{with} \quad \Delta p^1 = \begin{bmatrix} \Delta p_x^1 \\ 0 \\ 0 \end{bmatrix}, \Delta p^2 = \begin{bmatrix} 0 \\ \Delta p_y^2 \\ 0 \end{bmatrix}, \Delta p^3 = \begin{bmatrix} 0 \\ 0 \\ \Delta p_z^3 \end{bmatrix} \quad (26)$$

is the pressure gradient (or pressure difference) in the ith direction. Δp_x^1, Δp_y^2, and Δp_z^3 represent the pressure drop in the corresponding direction in Pa.

$$L^1 = \mathrm{NX} \cdot l_v, L^2 = \mathrm{NY} \cdot l_v, L^3 = \mathrm{NZ} \cdot l_v \quad (27)$$

are the physical lengths of the computational domain in the directions of interest. l_v is the physical length of a voxel, and NX, NY and NZ are the numbers of voxels in the three coordinate directions (**X**, **Y**, and **Z**).

The diagonal entries of the permeability tensor describe the measurable one-dimensional permeabilities in the x, y, and z direction.

The off-diagonal entries of the permeability tensor describe if a pressure drop in one direction will cause some flux in another, perpendicular direction. This might happen, e.g., for diagonal pores. If symmetric boundary conditions are used in perpendicular (or transverse) directions[32], no net flow is possible over the sides of the domain, and the off-diagonal entries will be zero.

Important! Note that the length definition in (27)[20] includes the complete thickness of the 3D grid. This is an appropriate definition if the 3D model shows a representative inner part of the material, and does not include any inflow or outflow regions.

If the 3D model includes empty regions as inflow or outflow region, this definition is no longer correct and the length used for the permeability computation has to be adjusted. You can do that during the post-processing, and define the material thickness in flow direction, which is then used to compute the permeability.

For some boundary conditions (e.g., constant velocity at the inflow), the solver needs an empty inflow area to be able to compute a solution. These contradicting demands can be handled in two different ways:

- **Explicit** inflow/outflow region: The inlet or outlet is part of the 3D structure model. You can, e.g., use ProcessGeo - Embed to add this areas explicitly to the structure. You then have to enter the material thickness (without inflow and outflow) manually in the post-processing to compute the correct permeability.
- **Implicit** inflow/outflow region: The inlet or outlet is not part of the 3D structure model. The flow solver internally adds empty regions to the domain and solves the boundary value problem. The solver then removes these regions from the result files and does not use them for the post-processing. With this, the permeability computation uses the correct thickness by default. The disadvantage is that you cannot visualize the flow in the implicit regions as they are not part of the results.

The following two properties are derived from the flow permeability. Thus, they can also only be computed from the solution of the Stokes[8] or Stokes-Brinkman[9] equation, and not for fast flows.

Flow Resistivity

In experiments, the mean velocity in the direction of the pressure drop is measured. The measured pressure drop is then divided by the measured mean velocity to obtain the measured flow resistivity.

The flow resistivity σ in one space direction is therefore given by

$$\sigma = \frac{\mu}{\kappa} \tag{28}$$

where κ is the permeability and μ is the dynamic fluid viscosity. In 3D, a flow resistivity tensor would be defined as $\mu \mathrm{K}^{-1}$. Thus, flow resistivity is **NOT** a material property, because it always depends on the fluid viscosity.

In FlowDict, the flow resistivity is not determined as a 3D tensor, but reported for each computational direction. In X-direction, μ / K_{xx} is computed, and similar in Y- and Z-direction.

Gurley Value

The Gurley second or Gurley unit describes the number of seconds required for 100 cubic centimeters (1 deciliter) of air to pass through 1.0 square inch of a given material at a pressure differential of 4.88 inches of water (0.176 psi) (ISO 5636-5:2003).

With these values gathered in the table:

Δp	= 0.176 psi	= 1213.48 Pa
A	= 1 in^2	= 0.00064516 m^2
V	= 100 cm^3	= 10^{-4} m^3

the Gurley value can be obtained from Darcy's law[20] with:

$$G = \frac{\mu V L}{\kappa \Delta p A} = \frac{1.834 \cdot 10^{-5} \cdot 10^{-4} \cdot L}{\kappa \cdot 1213.48 \cdot 0.00064516} = 2.3426 \cdot 10^{-9} \cdot \frac{L}{\kappa} \tag{29}$$

The Gurley value depends on the permeability and the physical length of the computational domain in the directions of interest.

1.2 Stokes(-Brinkman)

This command solves a boundary value problem to determine the stationary flow through the media. If all material IDs present in the 3D structure model describe either fluid or solid materials, the Stokes 8 equation is solved. If some material IDs represent unresolved porous materials, the Stokes-Brinkman 9 equation is solved.

These equations are valid for small Reynolds numbers, as is the case for slow laminar flows, or for fluids with high viscosity, such as oil. For faster flows (higher Reynolds numbers), you must use the Navier-Stokes(-Brinkman) command 58.

In the **Stokes(-Brinkman)** command, the flow boundary conditions are aligned with the space direction - one side of the domain is the inflow area and the opposite side of the domain is the outflow area. As boundary conditions, you can prescribe flow rate, inflow velocity, or pressure.

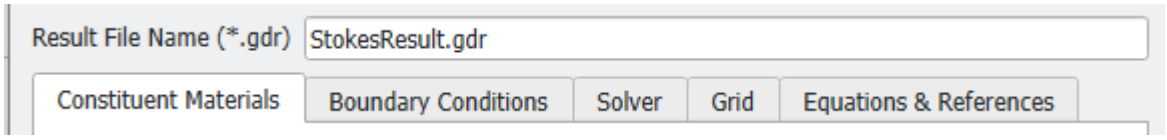
The permeability of the media is determined as a post-processing step. You can select to compute the permeability in one space direction only, or compute the full 3D tensor by solving the boundary value problems for each space direction.

The Stokes-Brinkman command

- Options 23
  How to set the input parameters of the command.
- Results 50
  View the simulation results.

1.2.1 Options

Select **Stokes(-Brinkman)** from the pull-down menu and click the **Option**'s **Edit...** button to open a dialog to enter or edit the simulation options. Choose a **Result File Name** for the result file (*.gdr) fitting to your current project.



The parameters are organized under the tabs: **Constituent Materials**, **Boundary Conditions**, **Solver**, and **Grid**. The **Equations & References** tab gives a summary of the used equations and symbols which are explained in more detail here 9.

Tabs of the Stokes-Brinkman dialog

- Constituent Materials 24
  Define the materials in the structure.
- Boundary Conditions 27
  Set up the boundary value problem that describes your experiment.
- Solver 36
  Select a flow solver.

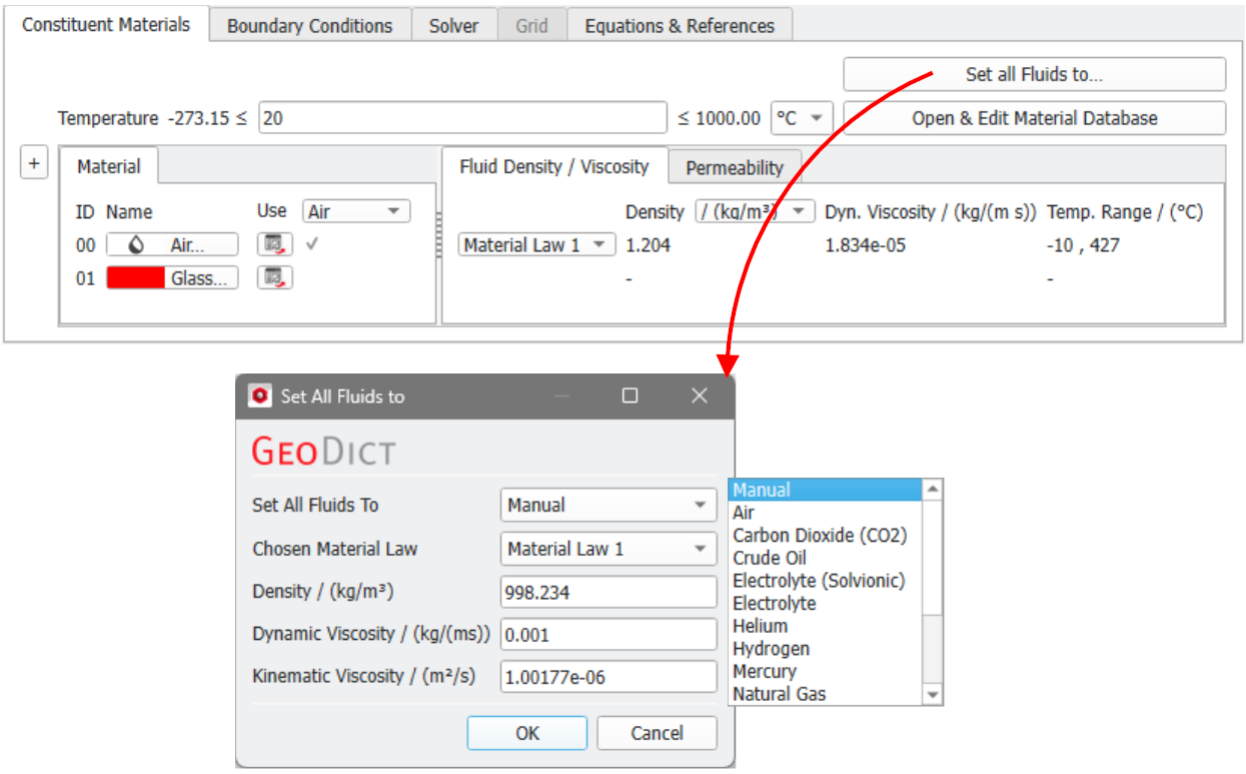
Tabs of the Stokes-Brinkman dialog

- Grid 49
  Adjust the computational grid.

Constituent Materials

For the flow solver's computation, the materials in the structure and the fluids flowing through it need to be defined under the **Constituent Materials** tab, so that their physical properties parameters are taken into account.

To set the fluid flowing through all pore and porous materials, click **Set all Fluids to** and select it from the pull-down menu.

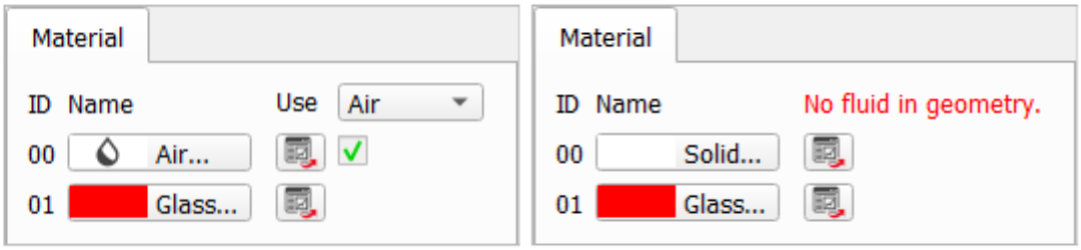


If one of the predefined fluids is used, the values for **Density**, **Dynamic Viscosity**, and **Kinematic Viscosity** are taken from the GeoDict Material Database. They are dependent on the given **Temperature** value and the **Chosen Material Law**. If **Manual** is selected, these values can be entered manually.

Any values entered manually, e.g., for a special type of oil, can also be added to the material database through the **Open & Edit Material Database** button. Additionally, different properties for the same material can be saved as different **Material Laws** and can then be selected from the pull-down menu in the **Set all Fluids to** dialog. **Material Law 1** is the default shipped with GeoDict. Click here for more information on editing, expanding, and using the GeoDict Material Database.

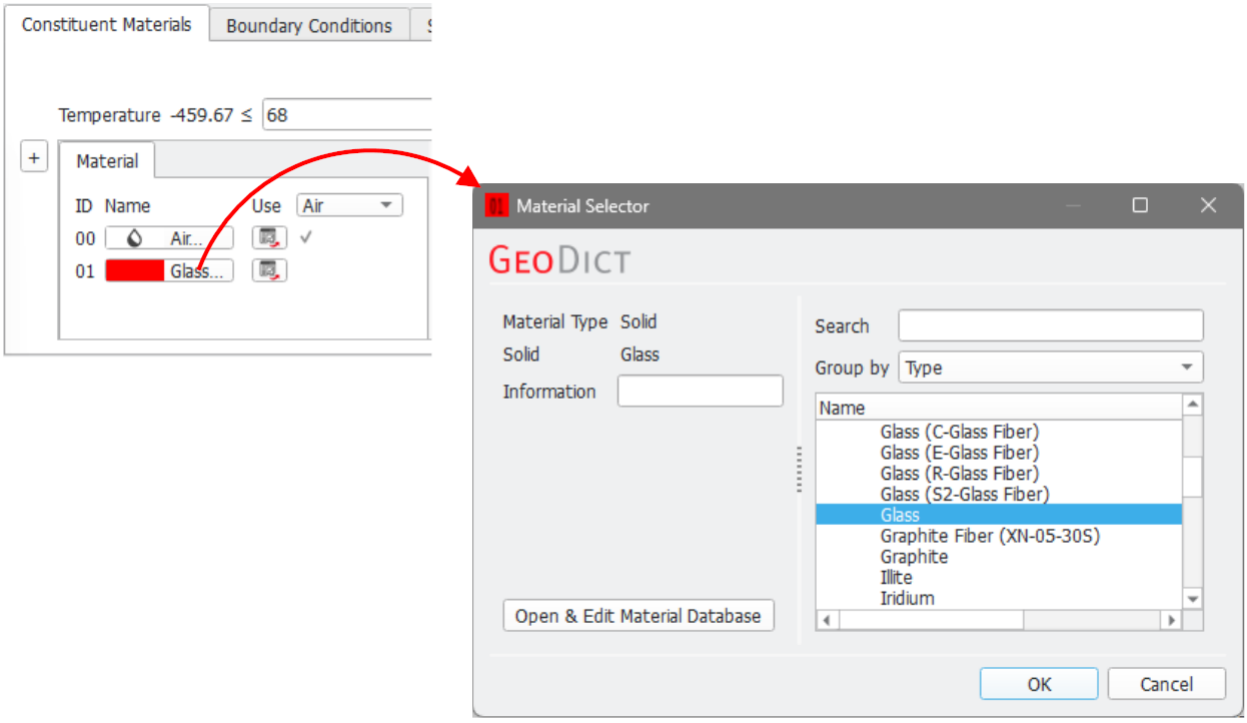
The **Temperature** is selectable in Kelvin (K), Celsius (°C), and Fahrenheit (F), and has a default value of 293.15 K (20.0°C, 68 F).

Under the **Material** subtab, select the fluid used in the simulation from the **Use** pull-down menu. If other fluids are contained in the structure, they will be treated as immobile (solid) during the simulation. If the structure contains no fluid material, **No fluid in geometry** is displayed in the top of the **Material** tab. Select a material for the pore phase by clicking on the corresponding material button.

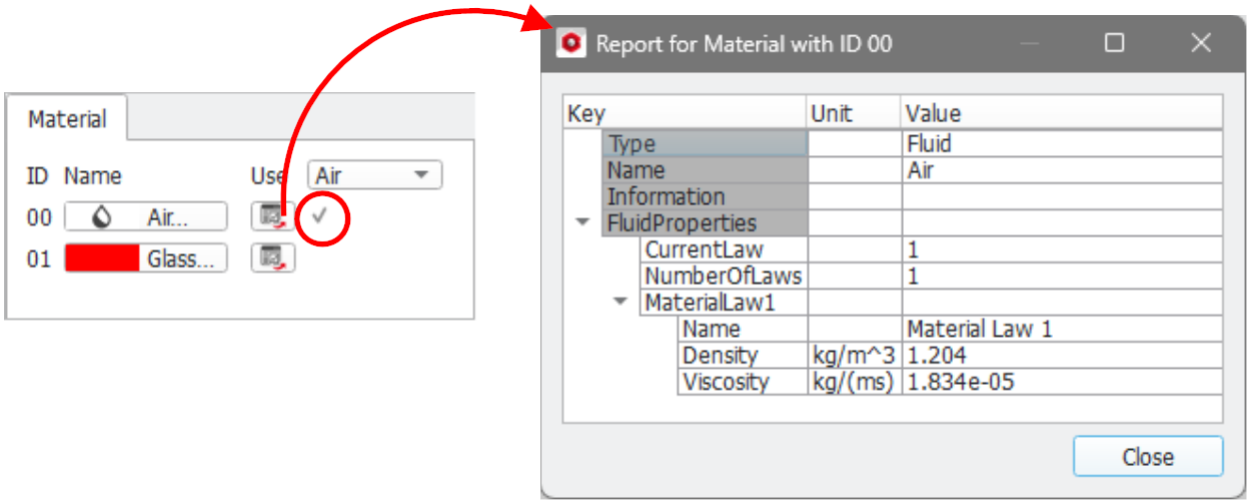


The density and the dynamic viscosity of this fluid, at the selected temperature, is shown under the **Fluid Density/Viscosity** subtab.

The **Material** subtab shows also the material **IDs** assigned to the materials (00, 01, 02, etc.). For each material ID, you can open the **Material Selector** dialog and select a material by clicking on the corresponding material buttons.

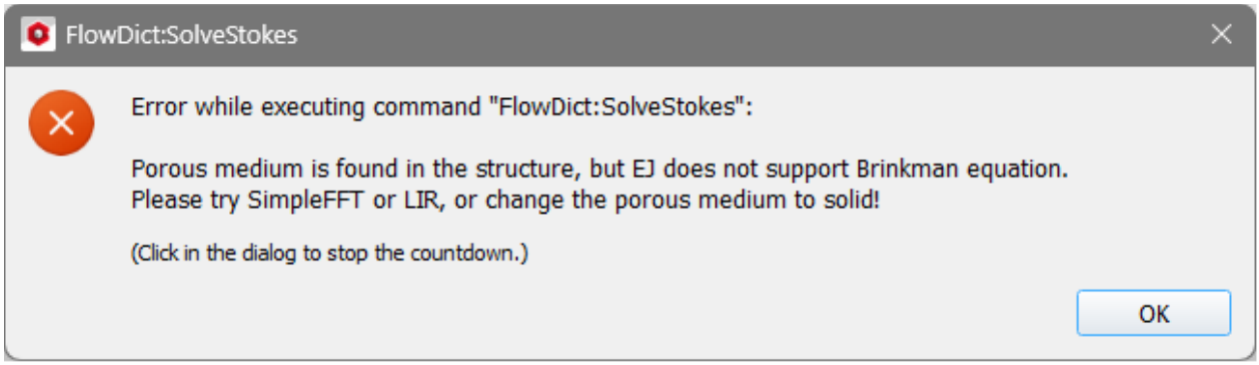


Click the **Material Report** button to see the relevant parameters for the material with a selected ID.

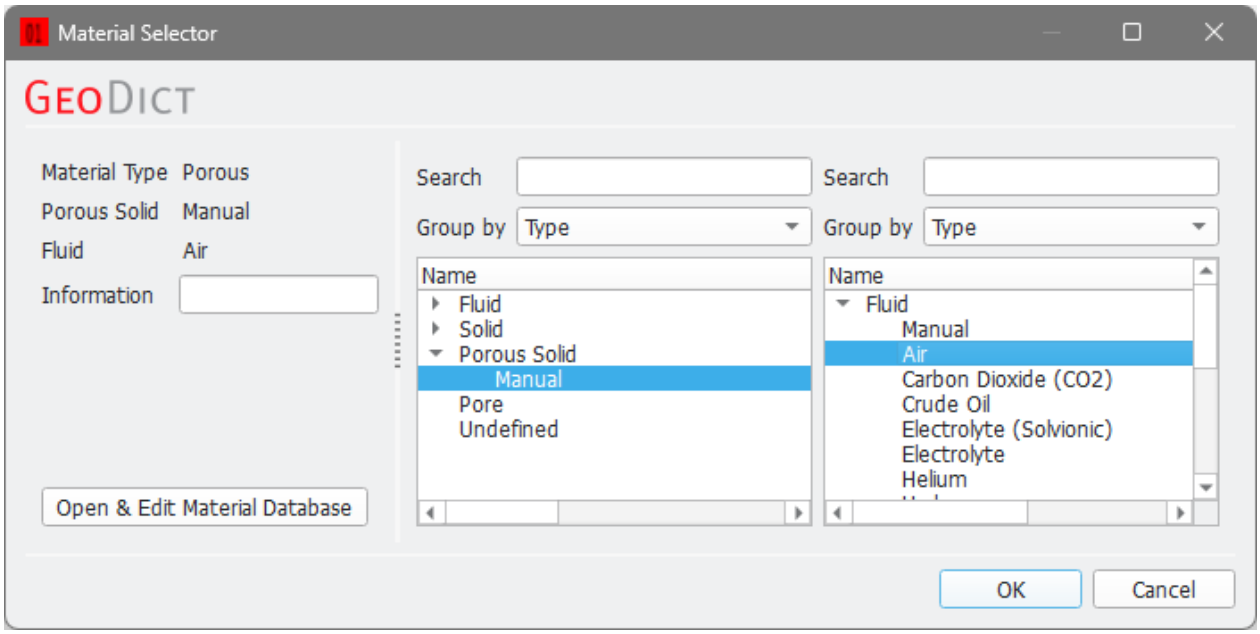


Fluid flow can happen in pores and porous materials, but not in solid materials. The check sign next to the **Material Report** icon marks all materials in which flow happens.

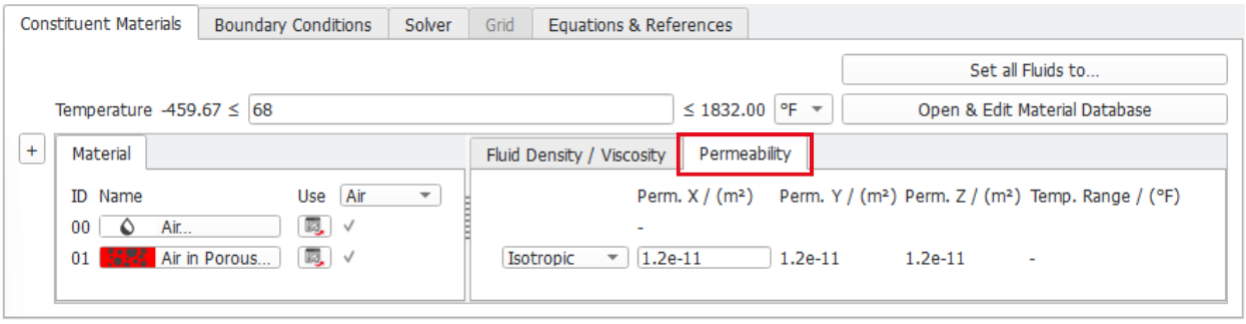
When the structure model not only contains solid and empty (pore) voxels, but also porous voxels, you must select either the SimpleFFT 12 or the LIR 12 solver in the Solver 36 tab. If you select the EJ 13 solver, an error message will show after clicking **Run**.



For porous voxels, set the **Material Type** (in the Material Selector dialog) to **Porous** and choose the fluid that occupies the pores.



The permeability of porous voxels is set under the **Permeability** subtab. You can define if the material is isotropic (same permeability in all flow directions) or anisotropic (permeability depends on flow direction). When the material is isotropic, the permeability entered in the X-direction is used in all directions. For an anisotropic material, the permeabilities in the three Cartesian directions are needed.



The local orientation of porous materials with anisotropic properties is not considered. You must select the **SimpleFFT** solver if anisotropic permeabilities are entered.

Boundary Conditions

To simulate the flow through the domain, GeoDict solves a boundary-value problem. In the fluid-filled part of the domain, the flow is governed by a differential equation. The boundary values describe what happens at the boundaries, and are a necessary and important part of the problem description. Boundaries are the six faces of the cuboid (X-, X+, Y-, Y+, Z-, Z+), and the pore-solid walls within the structure.

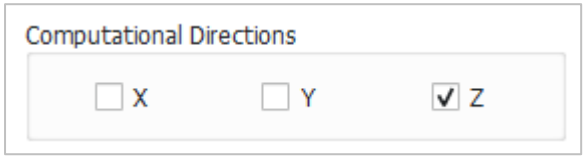
For the six cuboid faces, FlowDict distinguishes between the domain faces in the computational direction (also called flow direction or longitudinal direction) and the domain faces in the two other space directions (called transversal or tangential directions). The computational direction is the direction in which pressure drop and flow rates are measured.

Computational Directions

As **Computational Directions**, select the directions for which you want to calculate the flow. To obtain the full 3x3 permeability matrix and a pressure drop value for each direction in the results file, select all three directions.

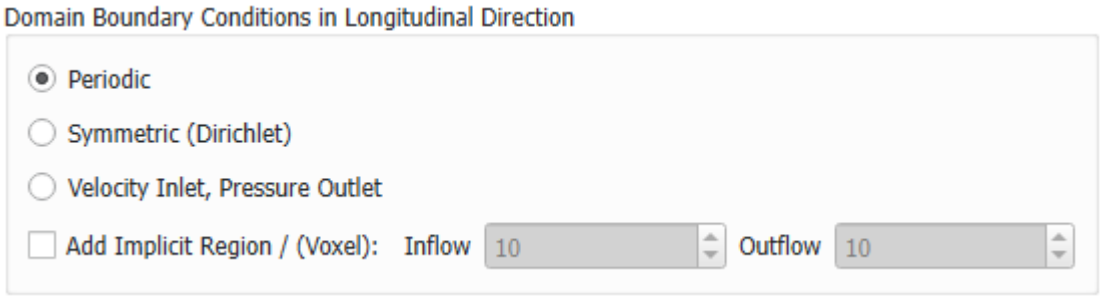
For every selected direction, a boundary value problem is solved in which the **Domain Boundary Conditions in Longitudinal Direction** and the corresponding **Experiment Input/Output** are applied in the computational direction, and the **Domain Boundary Conditions in Transverse Directions** are applied on the other two space directions.

Any directions not selected will not be calculated and will be marked as 'unknown' in the permeability matrix in the **Report** tab of the result file.



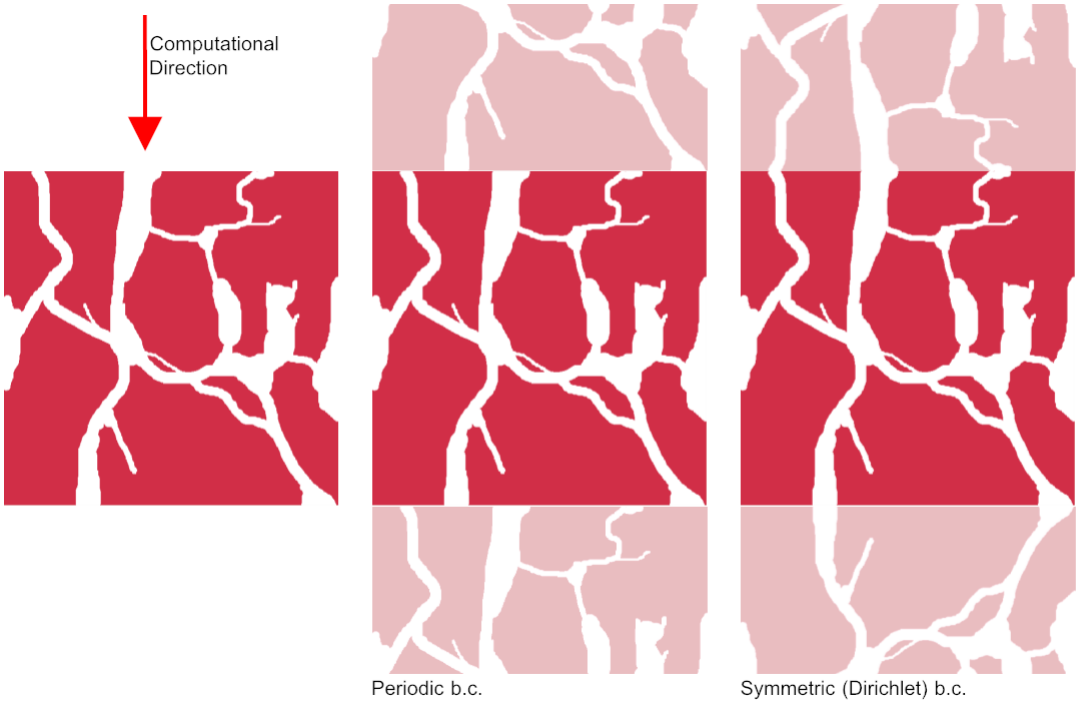
Domain Boundary Conditions in Longitudinal Direction

These boundary values are applied in the **Computational Direction**. Select either **Periodic**, **Symmetric (Dirichlet)**, or **Velocity Inlet, Pressure Outlet** conditions.



Use **Periodic** boundary conditions for periodically generated structure models and for non-periodic structures with high porosity. The structure is internally repeated periodically at the domain boundaries in the computational directions.

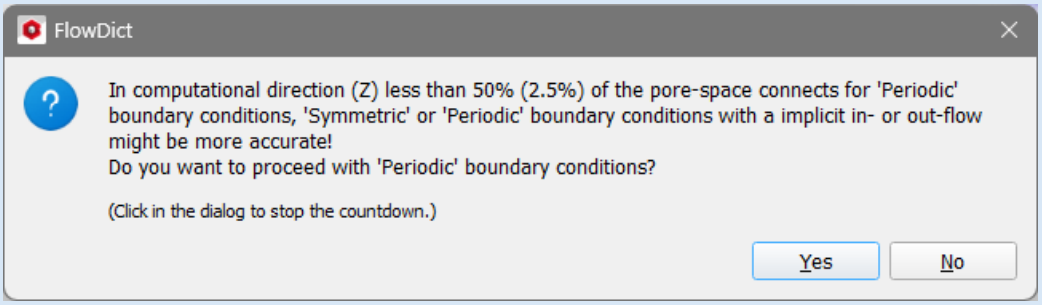
Use **Symmetric** boundary conditions for non-periodic structures if the percentage of solid voxels is high at the inlet. The solver treats the structure as mirrored at the domain boundaries in the computational directions.



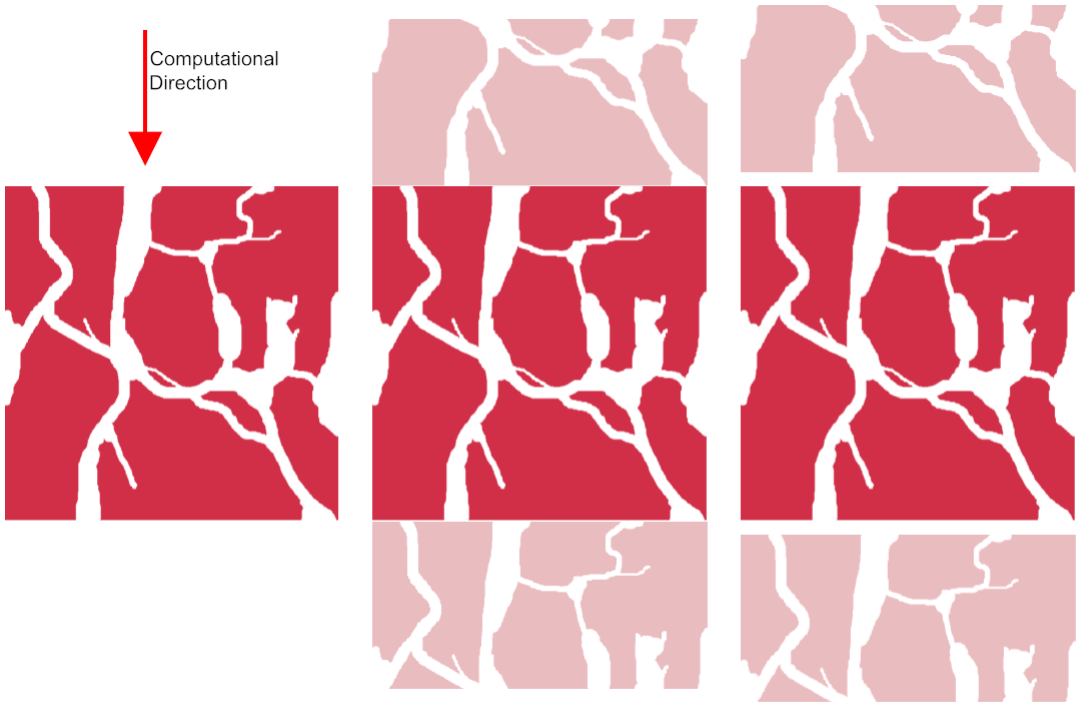
The option to **Add** an **Implicit Region** is not available for **Symmetric** boundary conditions. The size of the implicit region in the computational directions is given in voxel, where the default added implicit inflow and outflow are 10 voxels, respectively.

Know how! The term implicit means that the region is only added by the solver internally. Thus, these regions do not contribute to the computed permeability and flow resistivity.

Note! A warning may appear when trying to run flow computations without adding an inflow region when the pore space at the domain boundaries in the computational directions is less than 50%.

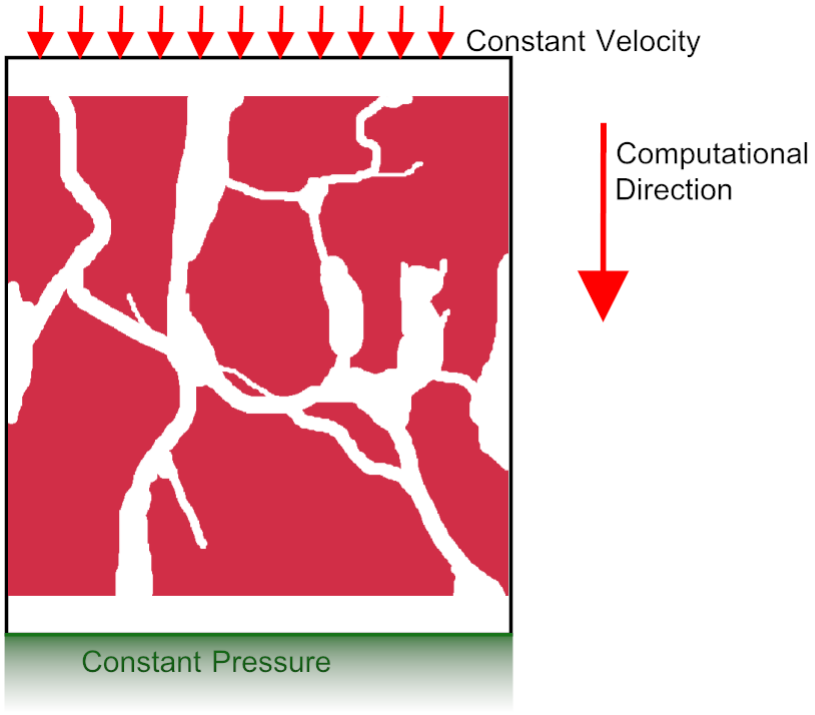


The inlet and outlet are essential to avoid the possibility of closing the flow channels when the structure is periodically repeated. In the example shown below choosing periodic boundary conditions in the direction of the flow without adding an implicit region results in the flow channels being artificially closed.



To open the channels and enable flow, you can either add an inflow and/or outflow region, or select symmetric boundary conditions. However, when possible, we recommend using periodic boundary conditions, as these require less computational memory and take less time to run.

The **Velocity Inlet, Pressure Outlet** (VinPout) boundary conditions apply a constant flow velocity at the inlet and a constant pressure at the outlet. These boundary conditions can only be applied if the structure contains empty inlet and outlet regions. A constant flow velocity cannot be applied directly on a solid surface, as this would lead to an unphysical solution. Therefore, you either have to assure that the current structure already contains such regions or you have to add those regions for the flow computation by additionally selecting **Add Implicit Region**.



Note, that selecting **Velocity Inlet, Pressure Outlet** automatically disables the EJ solver in the **Solver** tab.

Experiment Input / Output

In the **Experiment Input / Output** panel select which property is prescribed in the computational direction. This will also define the property that will be computed. You can prescribe the **Pressure Drop** and obtain the **Mean Velocity** as result. Alternatively, you can prescribe the **Mean Velocity** or the **Flow Rate** and obtain the **Pressure Drop** as result.

This selection corresponds to the typical experimental setup, where you either apply a constant pressure drop over the media and measure the through flow rate or you apply a fixed pump flow rate and measure the resulting pressure drop over the media.

The **Pressure Drop** is the difference between the inflow and outflow pressures.

The **Mean Velocity** is the average speed of the flow in the positive direction, i.e., from Z- to Z+ if the Z-direction is the computational direction.

The **Flow Rate on Flow Area** is a volumetric flow rate. It is the fluid volume which

passes per unit time through a given area.

For each property, the desired unit can be selected from the pull-down menu.

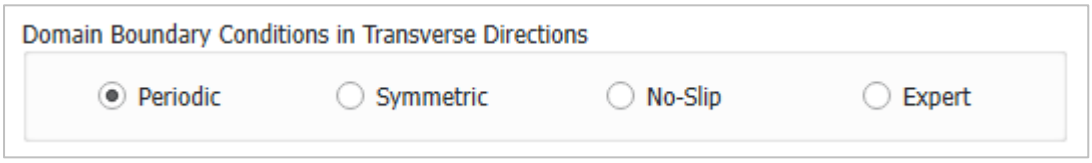
Which options are available here depends on the **Domain Boundary Conditions in Longitudinal Direction** you selected.

For **Velocity Inlet, Pressure Outlet** boundary conditions, you must prescribe the **Mean Velocity** or **Flow Rate**.

For **Periodic** or **Symmetric** boundary conditions, any selection is possible due to the linearity of the Stokes(-Brinkman) equation. Internally, the solver always prescribes a pressure drop and computes the mean velocity. However, GeoDict can rescale the computed result to match any mean velocity or flow rate and thus also compute the pressure drop for a prescribed mean velocity or flow rate.

Domain Boundary Conditions in Transverse Directions

The **Domain Boundary Conditions in Transverse Directions** can be **Periodic**, **Symmetric**, **No-Slip**, or **Expert**.

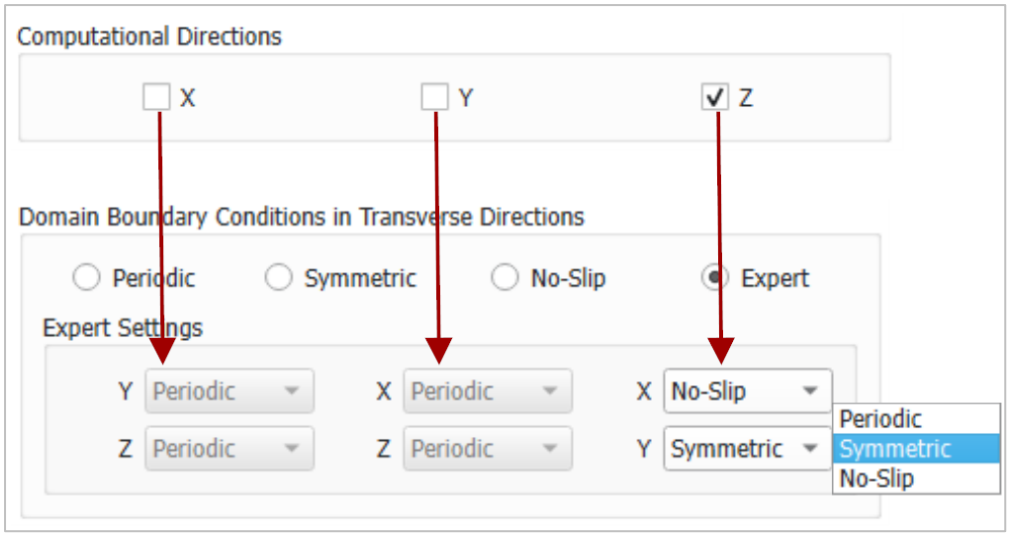


With **Periodic** boundary conditions, the structure is repeated in the transverse directions.

With **Symmetric** boundary conditions, the structure is mirrored at the domain boundary. The LIR and EJ solver solve the symmetric boundary condition by using Neumann (or zero flux) condition at the domain boundary.

With **No-Slip** (or Encase) boundary conditions, the domain boundary is treated as closed by solid material. When **No-Slip** is used, the solver internally adds a one-voxel layer in the required direction and solves with periodic boundary conditions. This is effectively equivalent to solving the structure with casing at both sides in the transverse direction. So, the structure size in the corresponding transverse direction becomes enlarged by one voxel internally.

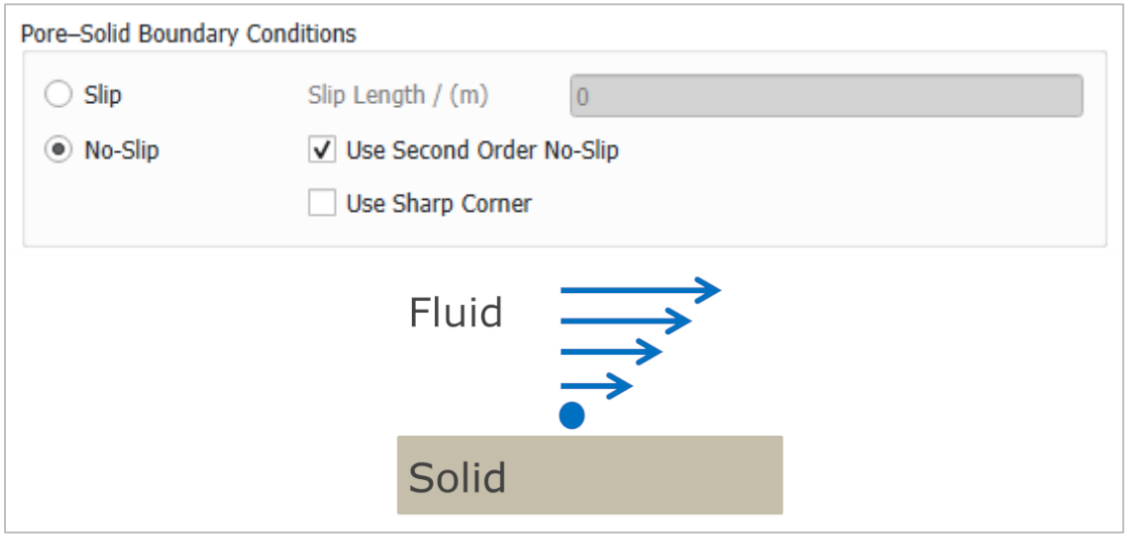
With **Expert**, it is possible to define different boundary conditions for each side in each computation. The boundary conditions in the two directions transverse to the flow can also be different by checking **Expert** boundary conditions. For example, when the fluid is chosen to flow in the Z-direction, the boundary conditions could be chosen to be **No-slip** in X-direction and **Symmetric** in the Y-direction.



Pore-Solid Boundary Conditions

The **Slip Length** allows to include sliding effects in the flow simulation. Sometimes the permeability of gases can be somewhat different from the permeability of liquids in the same media. One difference is attributable to "slippage" of gas at the interface with the solid when the gas mean free path is comparable to the pore size.

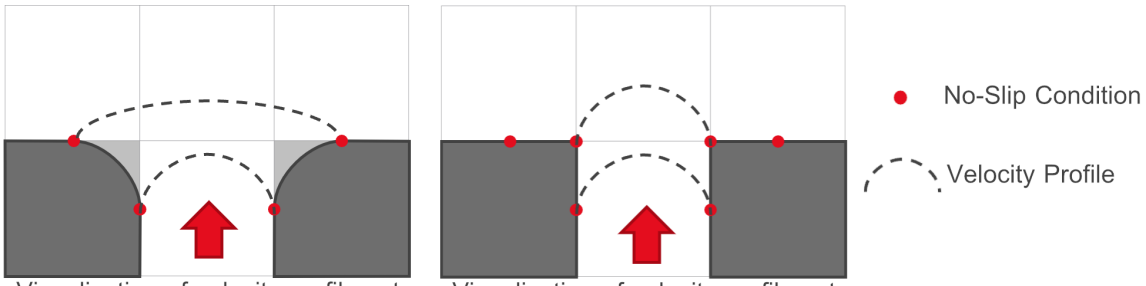
The default setting **No-Slip** corresponds to a flow velocity of zero along the structure.



Two settings are available for the **No-Slip** boundary condition. **Use Second Order No-Slip** is activated by default. This sets the tangential velocity to zero at the center of the voxel surfaces and uses a second order approximation of the velocity field. The first order setting, which is used when the box is left unchecked uses a linear approximation (Harlow-Welch, 1965 149).

Use Sharp Corner can be used for digital rocks where permeability is usually overestimated or cases where voxels represent exact geometry. Activating it sets the tangential velocity to zero at the voxel corners. When the box is unchecked the default

Round Corner is applied, which should be used for structures with round obstacles like fibers or grains.

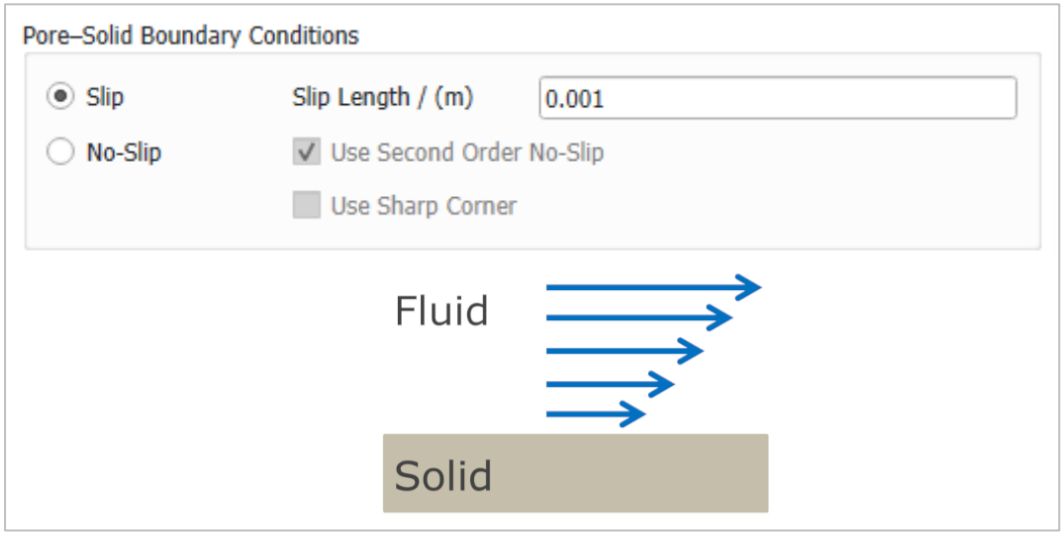


Visualization of velocity profiles at round corners

Visualization of velocity profiles at sharp corners

Slip with a non-zero **Slip Length** simulates the sliding of the fluid along the solid, increasing the fluid mean flux and thus, the permeability of the structure. This option might be used when it is realistic for a given physical material. Currently, the same slip length value must be set for all materials in the structure.

In earlier releases, this feature worked correctly for axis aligned walls only, but since GeoDict 2020 the expression of the slip velocity, which assumes the slip velocity proportional to the shear stress at the surface, is reformulated for different velocity components when the angle of fiber surface is known, and reimplemented in the flow solver. Thus, direct simulation of the slip flow is possible.



$$\vec{n} \cdot \vec{u} = 0 \text{ on } \Gamma$$

(30) No flow into solid

$$\vec{t} \cdot \vec{u} = -\lambda \vec{n} \cdot \nabla(\vec{u} \cdot \vec{t}) \text{ on } \Gamma$$

(31) Slip flow along solid

Here, $\vec{u}$ is the fluid flow velocity as it was introduced in Darcy's law [19] and the

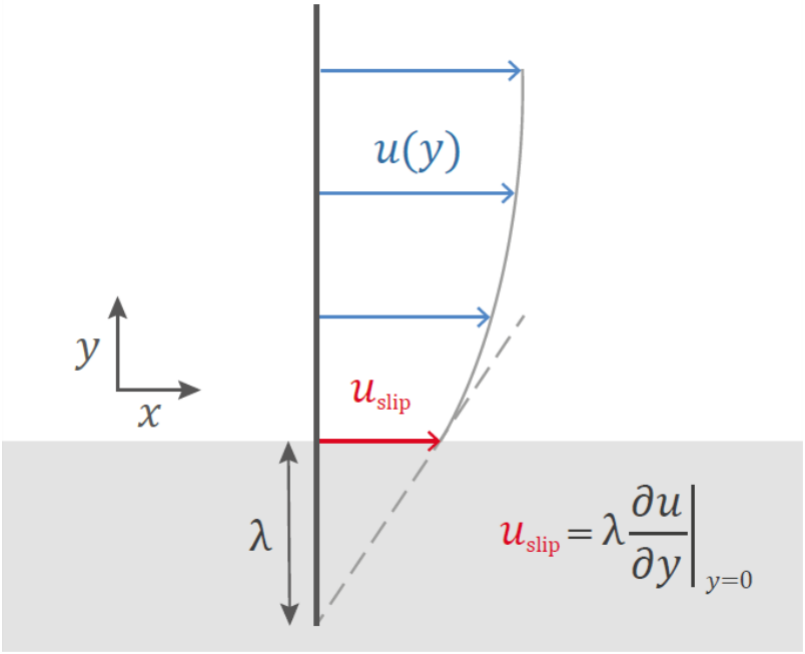
Stokes[8] equation, $\vec{n}$ is the normal direction to the solid surface, Γ is the fiber surface, λ is the **Slip Length** and $\vec{t}$ is any tangential direction with $\vec{t} \cdot \vec{n} = 0$. For the same pressure drop, the computed velocities for slip boundary conditions are higher than for no-slip boundary conditions. Conversely, for a given velocity or equivalently, a given mass flux, the pressure drop is lower when computed with slip boundary conditions.

For a straight channel structure, the two slip flow equations above become:

$$u = \lambda \frac{\partial u}{\partial y}$$

(32) Slip flow in straight channels

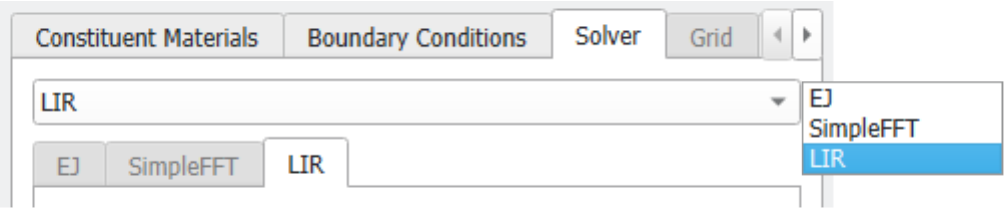
In this following example figure, the gray solid serves as a wall for a straight channel. To obtain the flow velocity along the solid (u_{slip}) a coordinate system is defined by the solid orientation.



For a **Slip Length** of zero ($\lambda = 0$), the graph that describes the flow velocity in dependence of the distance to the solid surface, would start at the surface of the solid. However, for a positive **Slip Length** ($\lambda > 0$) it is shifted into the solid by λ . Thus, the flow velocity u_{slip} is defined.

Solver

Depending on the equation to be solved, two or three different flow solvers[12] are available for solving the equation. The Stokes equation can be solved by the **LIR**, **SimpleFFT**, or **EJ** solver. The Stokes-Brinkman equation can be solved by the **LIR** or **SimpleFFT** solver. Select the solver in the pull-down menu to activate the corresponding tab below.



The LIR solver requires the least computational resources (RAM) and is therefore selected by default. Often, the LIR is also the fastest solver, but that may depend on the parameter selection and the 3D structure model.

Parameters of the Solver tab

- Simulation Stopping Criterion 37
  Define when the computation is stopped.
- Restart Save Interval 41
  Configure an auto-save interval.
- Parallelization 41
  Parallelize the FlowDict computations.
- Restart from a *.gdr File 41
  Continue a previous FlowDict computation.
- Discard PDE Solver Files 42
  Delete intermediate files.
- Advanced Options 42
  Additional solver options.

Simulation Stopping Criterion

The equations are solved by an iterative approach.

Iterative Solver

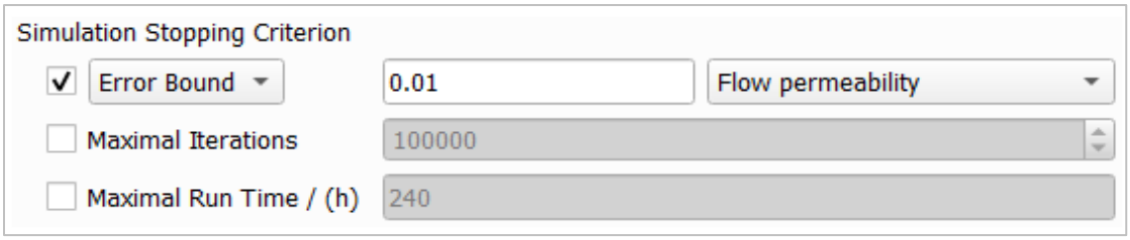
The basic idea of an iterative method is to

1. Start with some initial guess for the unknown values
2. Improve the current values in each iterative step
3. Repeat the iterative process until one of the stopping criteria is reached.

The iterative process is controlled by setting the values and activation for **Error Bound, Tolerance, Residual**, **Maximal Iterations**, and **Maximal Run Time (h**). The stopping criteria apply for each computational direction individually.

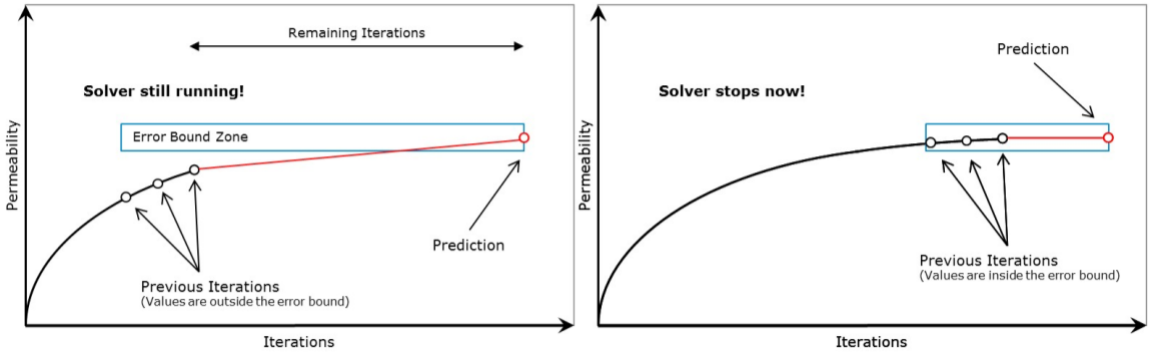
If multiple stopping criteria are selected, the first criterion that is reached causes the solver to stop.

The stopping criterion that has been reached can be viewed in the Result Viewer of the GeoDict result file (*.gdr) under the **Results Report** tab.



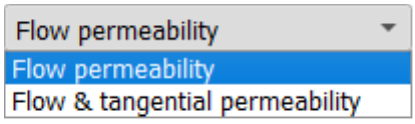
Error Bound (LIR and SimpleFFT)

The stopping criterion **Error Bound** uses the result of previous iterations, and predicts the final solution based on linear and quadratic extrapolation. The solver stops if the relative difference between computed and predicted solution is smaller than the specified error bound. The stopping criterion recognizes oscillations in the convergence behavior and prevents premature stopping at local minima or maxima. A damped convergence curve is fit through the oscillating curve and the solver stops then regarding the damped convergence curve.



If the Krylov method (under LIR - Advanced Options) is activated, the definition of **Error Bound** is somewhat different. Here, no prediction is made, instead the continuity between neighboring cells and the conservation of mass is checked. The maximal value is normalized by the mean flow and if this value is smaller than the **Error Bound** the simulation stops.

It is possible to choose if the stopping criterion should consider the permeability only in flow direction or in both flow and tangential directions. For flow simulations where the permeability in flow direction converges much faster than the permeability in tangential directions, it is recommended to change from **Flow permeability** to **Flow & tangential permeability**.



Tolerance (all solvers)

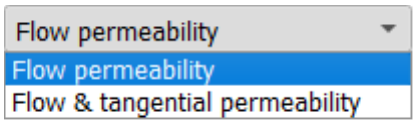
The **Tolerance** stopping criterion looks for stagnation of the method when the process becomes stationary, i.e., the improvement in the permeability value becomes extremely small from iteration to iteration.

In each iteration, the solver checks for the current computed value x_c against the values x_{c-i} of the last 100 iterations if there are any changes.

$$\max_{i \in [1,100]} \frac{x_c - x_{c-i}}{x_{c-i}} < \text{tolerance} \quad (33)$$

The computation is stopped if the maximal relative change is smaller than the value entered for **Tolerance**. When there is doubt about the quality of the solution, decrease the **Tolerance** value by a factor of ten. The drawback of this criterion is that the solver sometimes might stop too early in case of slow convergence rate or at local extrema of oscillatory convergence curves.

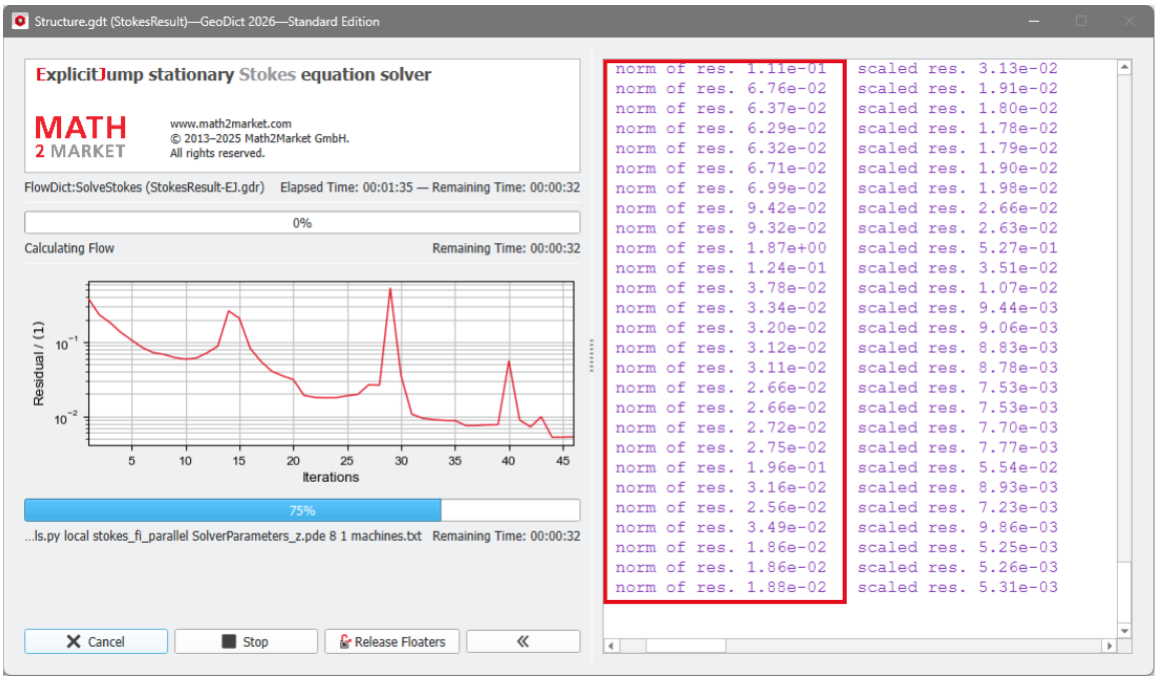
It is possible to choose if the stopping criterion should consider the permeability only in flow direction or in both flow and tangential directions. For flow simulations where the permeability in flow direction converges much faster than the permeability in tangential directions, it is recommended to change from **Flow permeability** to **Flow & tangential permeability**.



Residual (EJ and SimpleFFT)

When the **Residual** stopping criterion is used, the iteration is stopped if the solution satisfies the equation up to the required accuracy.

By setting the stopping criterion to **Residual**, the computations terminate as soon as the relative norm drops below the selected residual threshold. The relative norm of the Schur Complement residual is computed and displayed in the console window during the calculations. The console is visible by clicking the double arrow button on the lower right corner in the progress dialog.



The recommendation to choose **Tolerance** or **Residual** for the **EJ** solver is based on the structure's porosity. Both give similar results for highly porous structures. For dense structures, if using the Schur Complement **Residual**, the relative norm of the residual may be small even though the correct permeability has not been reached. So, when in doubt, use the **Tolerance** criteria – the default option.

Maximum Iterations and Maximum Run Time (all solvers)

Use the **Maximum Iterations** value or the **Maximum Run Time (h)** stopping criteria causes the solver to stop if the maximal number of iterations and/or the maximal runtime (in hours) is exceeded.

When the solver stops because one of these criteria has been reached, no guarantee on the quality of solution can be given. In this case, a warning is printed into the report. The following possibilities might help:

- Check the corresponding .log file to see how large the residual values and the permeability values are. If the permeability values are in reasonable range and they do not have changed during the last iterations, you may decide to use the current result.
- Double-check the structure and parameter values. Unphysical parameters or too rough resolution of the structure (leading, e.g., to artificial unconnected components) can cause an iterative solver to fail.

Know how! We recommend to use the **Error Bound** criterion for the **LIR** and **SimpleFFT** solver. This stopping criterion approximates the "relative deviation from the final solution".

Restart Save Interval

The calculations run by the solvers can be restarted from saved intermediate result files and the interval between auto-saves can be configured from the value entered in **Restart Save Interval (h)**.

Parallelization

The simulation process can be parallelized to speed-up the computation process. It depends on your license, how many parallel processes you may use. The solver benchmarks 13 illustrate the speed-up that you can achieve.

The **Parallelization Options** dialog box opens when clicking the **Edit...** button, to choose between **Sequential**, **Parallel (Shared Memory)**, **Automatic Maximum of Threads**, and (if available) **Cluster**.

For details on how to set up and run parallel computations, consult the High Performance Computing chapter.

Restart from a *.gdr File

In some situations, it may be useful to re-use previously computed results and save a great amount of time by restarting the process from an existing GeoDict result file (*.gdr). Typical examples would be

- non-sufficient accuracy of some computation, when it is suspected that more iterations may improve the quality of the result,
- trying another boundary condition (e.g., Dirichlet instead of Periodic or other way around),
- continuing an iteration process,
- or using similar parameter values for the computation.

To restart and initialize the solver from a result file (*.gdr), check **Restart from .gdr File** and enter the name of a file, or **Browse** for it in the project folder.

When using **Restart From .gdr File** only the volume field (*.vap file) associated to the loaded *.gdr file will be used to continue the simulation.

Important! The *.vap file is saved in the result folder of a previous run. Make sure that this file was not deleted from the results folder. If the *.vap file is not available the restart option cannot be used.

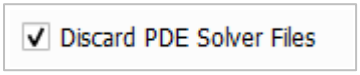
No other parameters from the options dialog will be loaded from the *.gdr file, but it is possible to load them by clicking **Load Parameters** in the Result Viewer. This way the

simulation can be continued while using the same settings as in the initial simulation.

Important! The structure from the run of which you want to use the result file must be loaded into GeoDict. If this is not the case, an error message is displayed.

Discard PDE Solver Files

Checking **Discard PDE Solver Files** causes the deletion of all intermediate computation files, such as log files and flow field files.



Only the content of the final GeoDict result file (*.gdr) is stored.

While having the benefit of saving hard disk storage place, discarding intermediate solver files has also the effect of disabling the 3D visualization of the results.

Advanced Options

Unfold **Advanced Options** to gain access to several advanced solver options.

Advanced Options for EJ:

Analyze Geometry

Checking **Analyze Geometry** performs an analysis of the geometry before the solver computations.

If no percolation path through the structure is found, the partial differential equation does not need to be solved, and a permeability of 0.0 is directly reported as the solution.

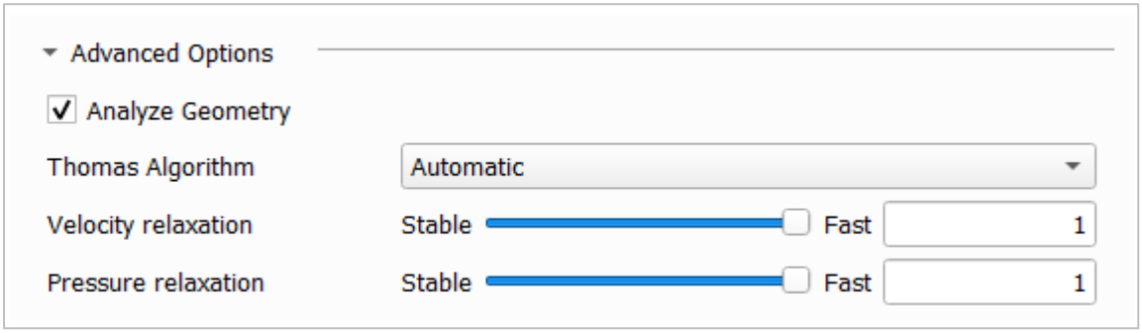
Furthermore, the solver operates on the whole structure, regardless of which parts are connected or unconnected. However, unconnected components are not transport relevant, thus the effort of solving in these parts is not necessary.

For these reasons, a geometrical analysis is routinely run to determine whether a through path exists. After the analysis is finished, unconnected components are

removed from the computational grid.

This may speed up the computations but requires some time for the geometrical analysis, especially for very large structures. So if you know beforehand that there are no unrelevant parts or that the structure has been processed to eliminate them, the geometry analysis can be switched off by unchecking **Analyze Geometry**.

Advanced Options for SimpleFFT:



Analyze Geometry

Checking **Analyze Geometry** performs an analysis of the geometry before the solver computations.

If no percolation path through the structure is found, the partial differential equation does not need to be solved, and a permeability of 0.0 is directly reported as the solution.

Furthermore, the solver operates on the whole structure, regardless of which parts are connected or unconnected. However, unconnected components are not transport relevant, thus the effort of solving in these parts is not necessary.

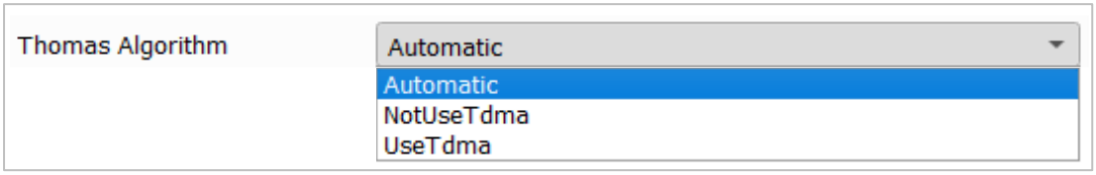
For these reasons, a geometrical analysis is routinely run to determine whether a through path exists. After the analysis is finished, unconnected components are removed from the computational grid.

This may speed up the computations but requires some time for the geometrical analysis, especially for very large structures. So if you know beforehand that there are no unrelevant parts or that the structure has been processed to eliminate them, the geometry analysis can be switched off by unchecking **Analyze Geometry**.

Thomas Algorithm

The tridiagonal matrix algorithm (Tdma), also known as the **Thomas Algorithm**, may help to improve the convergence for high porosity structure, especially for Navier-Stokes. The algorithm transforms the 3D flow problem to 2D. Thus, it leads to a shorter runtime and increases the convergence speed.

Three options can be chosen from the pull-down menu: **Automatic**, **NotUse Tdma** or **Use Tdma**.



If **Automatic** is selected, the SimpleFFT solver decides if **Tdma** will be used, regarding the porosity of the structure. Thus, **Tdma** is used if the porosity is high. For most cases it is recommended to choose **Automatic**.

If a message dialog appears, displaying that NaN is detected in iterations, **NotUse Tdma** should be tried.

If the structures porosity is very high, it might decrease the runtime a little bit to explicitly choose **Use Tdma**. But for most cases **Automatic** will work well.

For more information about the Thomas algorithm see here.

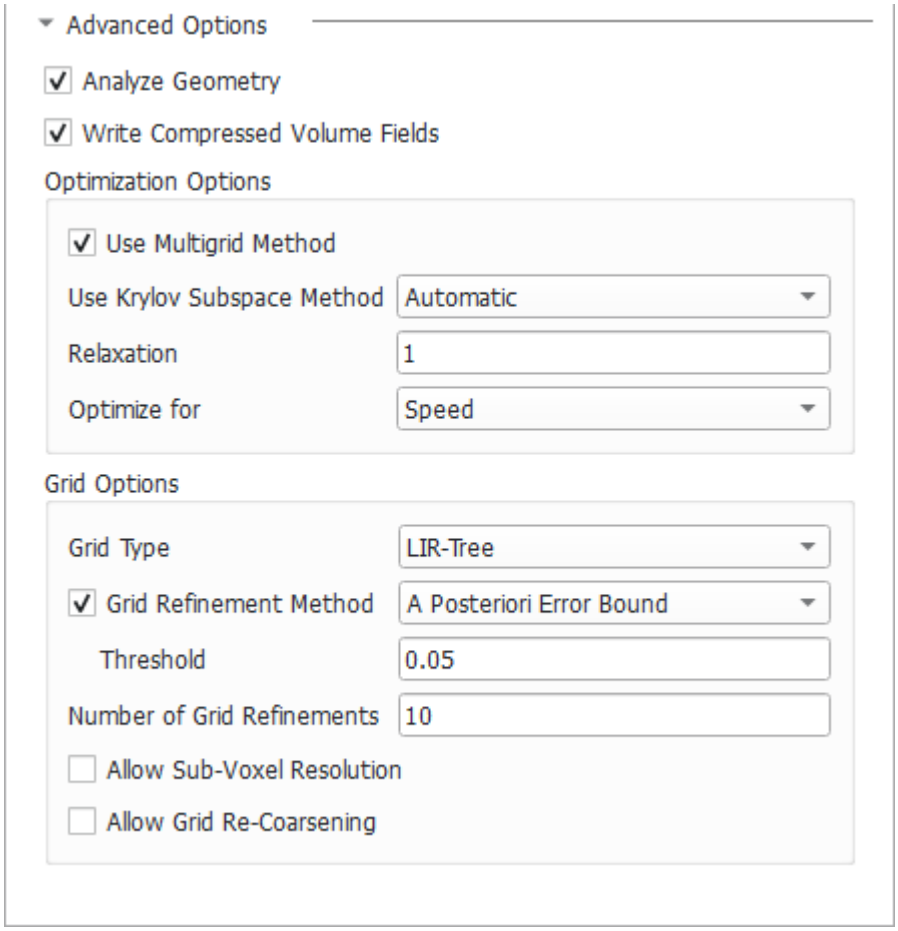
Relaxation

Depending on the material parameters and geometrical structure of the structure, the underlying mathematical problem can vary in complexity, thus influencing the behavior of the solver. This is directly related to the Reynolds number, an indication of the complexity of the flow solver computations. The higher the Reynolds number, the more **Stable** the flow solver settings should be, resulting in higher number of iterations, slower time stepping, and longer flow solver run times. However, making the solver run less iterations and, thus, faster (**Fast**), implies the risk that the solver does not converge.

For the SimpleFFT solver, the management of this balance is done through the **Velocity relaxation: Stable ↔ Fast** and **Pressure relaxation: Stable ↔ Fast** slide bars.

Setting the balance of **Stable** versus **Fast** is a trial-and-error process. Although there is no general rule to optimize it, the log files and the visualization of the structure might help finding the balance. Non-sense structure visualization results indicate that a more stable, slower solver is needed.

Advanced Options for LIR:



Analyze Geometry

Checking **Analyze Geometry** performs an analysis of the geometry before the solver computations.

If no percolation path through the structure is found, the partial differential equation does not need to be solved, and a permeability of 0.0 is directly reported as the solution.

Furthermore, the solver operates on the whole structure, regardless of which parts are connected or unconnected. However, unconnected components are not transport relevant, thus the effort of solving in these parts is not necessary.

For these reasons, a geometrical analysis is routinely run to determine whether a through path exists. After the analysis is finished, unconnected components are removed from the computational grid.

This may speed up the computations but requires some time for the geometrical analysis, especially for very large structures. So if you know beforehand that there are no unrelevant parts or that the structure has been processed to eliminate them, the geometry analysis can be switched off by unchecking **Analyze Geometry**.

Write Compressed Volume Fields

The LIR solver uses a very memory efficient adaptive grid structure for the simulations.

If the option **Write Compressed Volume Fields** is checked, then the adaptive grid is used as compression method for writing out *.vap files. This option allows to save 80-90% space on hard drive. The runtime for writing *.vap files is also reduced significantly. But the runtime for loading and uncompressing of compressed *.vap is increased by the amount of runtime that was saved for writing out compressed *.vap files.

If the option **Write Compressed Volume Fields** is not checked, then a usual regular grid is used for writing out *.vap files.

Optimization Options

The **Multigrid Method** (see, e.g., Wesseling, 2004) was introduced to speed-up the computation and reduce the runtime significantly. The main idea of **Multigrid** is the usage of multiple coarser adaptive grids to speed up convergence behavior but requires only little more memory.

The method is available to solve the Stokes and Stokes–Brinkman equations as well as for solving mechanics, diffusion, thermal, and electrical conduction and is enabled by default.

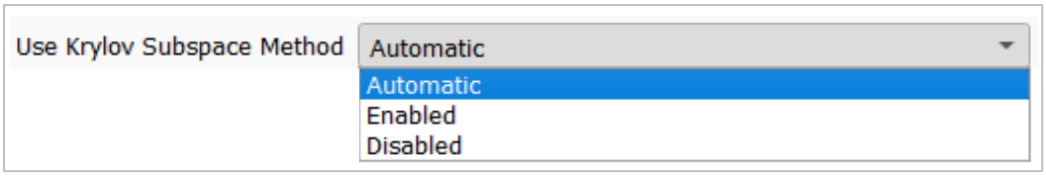
Another speed-up option to accelerate the convergence behavior of the LIR solver is called **Krylov Subspace Method**.

The runtime of the LIR solver depends on many different properties of the structure and the simulation parameters. The BiCGStab algorithm is used, which can reduce the runtime for challenging simulation very drastically.

Note! Using the BiCGStab method approximately doubles the amount of RAM needed for the computation.

Unfortunately, the Krylov method is not always faster than a simulation without the Krylov method and therefore we introduced an **Automatic** mode which uses some heuristics to choose the most efficient method based on structure, material parameters,

and boundary conditions automatically. Of course, it is possible to explicitly enable (**Enabled**) or disable (**Disabled**) the method.

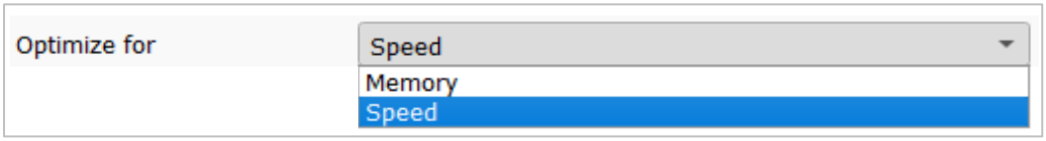


Note! In case that the Krylov subspace method (BICGStab) is used, the **Relaxation** may also be adapted automatically.

Depending on the material parameters and geometry of the structure, the underlying mathematical problem can vary in complexity, thus influencing the behavior of the solver. The more complex the problem is, the more stable the solver settings should be.

With the **Relaxation** number, the solver is adjusted from **Stable** (which results in higher number of iterations, slower time stepping, and longer solver run times), to **Fast**, which makes the solver run less iterations but implies the risk that the solver does not converge. The **Relaxation** is a parameter of the SOR method and must be between 0 and 2 to ensure convergence. For relaxation values smaller than one (<1.0), the simulation is more stable. For relaxation values larger than one (>1.0), the simulation converges faster.

The **LIR** solver can **Optimize for Speed** or **Memory**.



- If **Speed** is chosen, the solver constructs additional optimization structures. The runtime is decreased by up to 30% but requires up to 50% more memory compared to the other option.
- If **Memory** is chosen, the runtime is increased by up to 40% but the solver requires up to 50% less memory.

Grid Options

The **Grid Type** decides what kind of tree structure is used for the simulation.

The default option is **LIR-Tree** and should always be used. The solver uses an adaptive

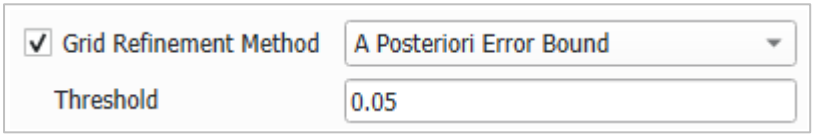
grid structure called LIR-tree and needs up to 10 times less runtime and memory compared to the **Regular Grid** option.

The solver can analyze the result field during the computation and improves the adaptive grid in places where more accuracy is needed. The LIR solver splits cells where a high gradient occurs.

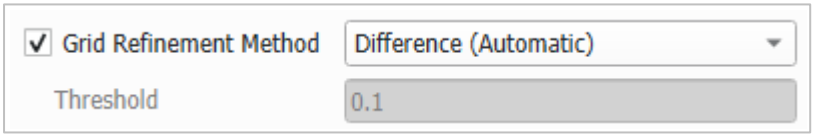
The solver can analyze the computed field and refines the adaptive grid during the computation at locations where more accuracy is needed. The LIR solver splits cells where a high gradients occur when the **Grid Refinement Method** is enabled. In this case, the additional parameters **Number of Grid Refinements**, **Allow Sub-Voxel Resolution**, and **Allow Grid Re-Coarsening** become available.

Select one of the three available refinement methods from the drop-down menu.

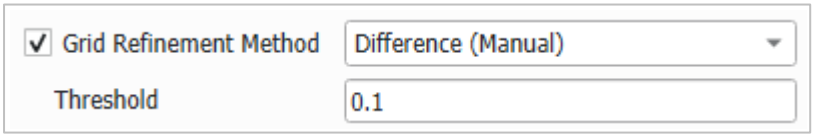
When **A Posteriori Error Bound** is selected, the solver targets the specified accuracy **Threshold**. While the **Error Bound** (set as Simulation Stopping Criterion) determines the relative error in the solution of the linear system, **A Posteriori Error Bound** refers to the relative error estimated by comparing high-order and low-order discrete solutions. The accuracy **Threshold** refers to the relative error to the analytical solution and the value must be between 0.0 and 1.0.



Choosing **Difference (Automatic)** leads to computational cells being split when the difference in values between neighboring cells exceeds a certain **Threshold**. The solver automatically chooses all internal parameters based on the structure and simulation settings. For this option, cells are split where the current gradient is greater than the **Threshold** multiplied by the maximum gradient, where the threshold is determined automatically.

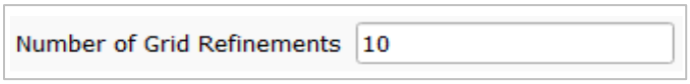


Selecting **Difference (Manual)** also causes the computational cells to be split when the difference in values between neighboring cells exceeds the specified **Threshold**. You can specify all parameters (Threshold and Number of Grid Refinements) for the grid refinement manually. For this option, cells are split where the current gradient is greater than the **Threshold** multiplied by the maximum gradient. The **Threshold** value must be between 0.0 and 1.0. The recommended value range is between 0.05 and 0.1.

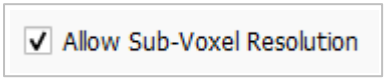


The **Number of Grid Refinements** controls the maximum number of grid refinements

that the solver can perform during the simulation. The value should be set between 0 and 10, where a value of 0 means that no grid refinements will be made. Grid refinements may increase the number of iterations, runtime, and memory requirements.



When **Allow Sub-Voxel Resolution** is enabled, the solver is allowed to split computational cells to sizes smaller than the voxel length.



This feature is beneficial for low-porosity structures for which the pore throats require finer resolution or for modeling fast Navier-Stokes[7] flows with strong vortices.

Enabling this feature may increase the number of iterations, runtime, and memory requirements.

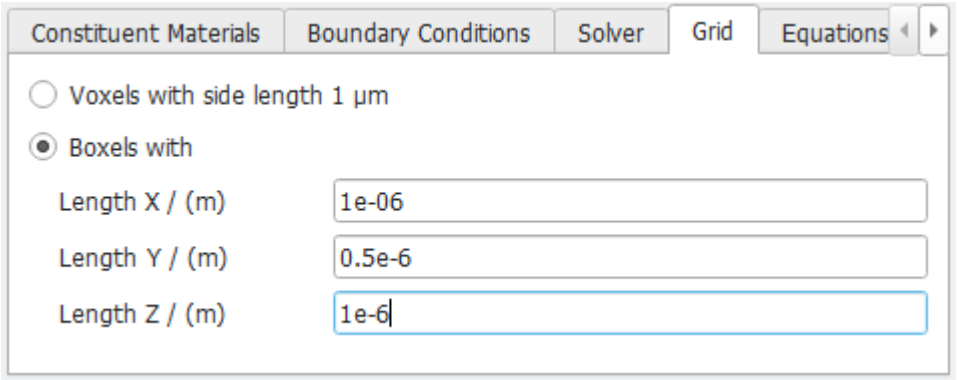
Check **Allow Grid Re-Coarsening** to allow the solver to automatically revert the grid refinement and coarsen the computational cells.

This means, grid refinement is done temporarily. Afterwards, the cells are merged as soon as accuracy is not needed anymore. This reduces the memory and runtime requirements.

Grid

For the EJ[13] and the SimpleFFT[12] solver, the computational grid can be adjusted under the **Grid** tab if necessary.

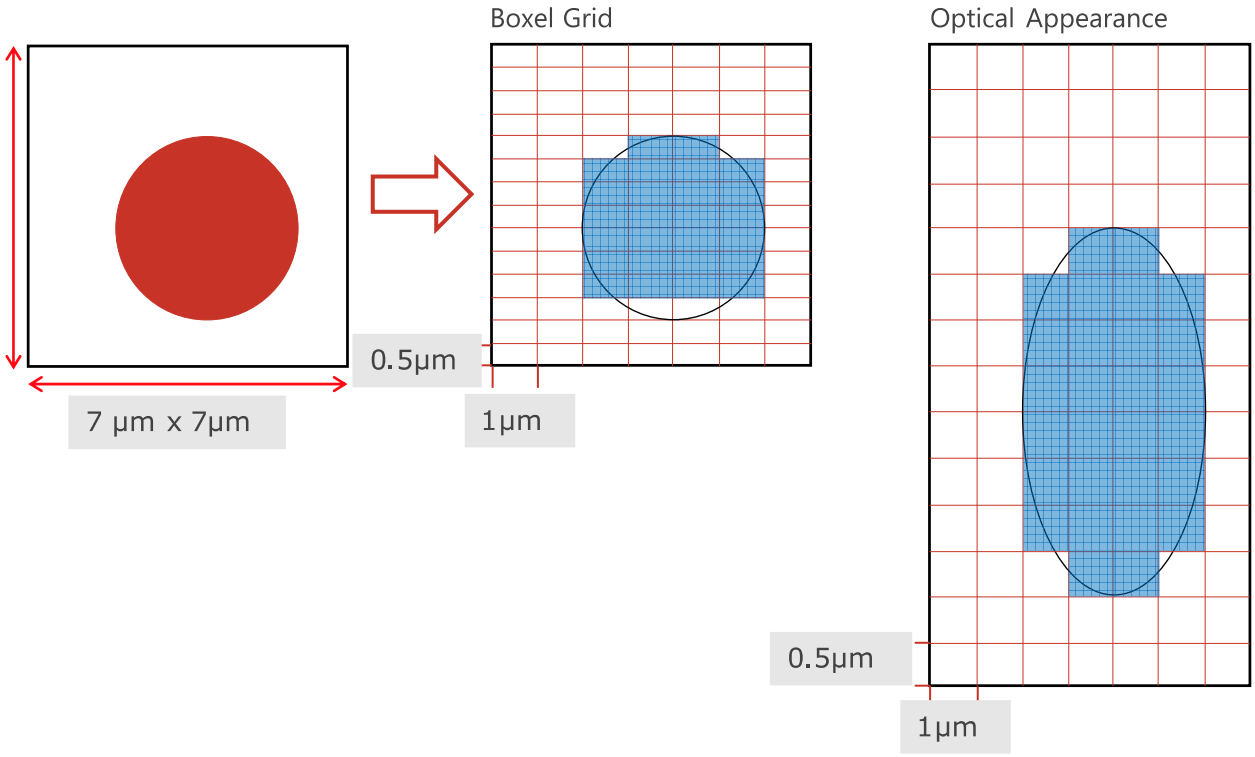
If **Boxels** (cuboid) instead of **Voxels** (cube) are used to resolve the structure, the boxel lengths can be set in the **Grid** tab.



Sometimes, grid lengths may be different in each direction, e.g., for Focused Ion Beam (FIB) images, the grid size in Z-direction is often different from the size in X- and Y-directions.

For example, a structure is sampled in a computational grid with different lengths in X- and Y-direction (e.g., LX = 2*LY). When the structure is loaded in GeoDict, each grid cell gets assigned

1 voxel and the visual representation of the structure looks stretched. To correct this effect, boxels set as, e.g., LX = 1 µm and LY = 0.5 µm, need to be chosen in the **Grid Size** panel.

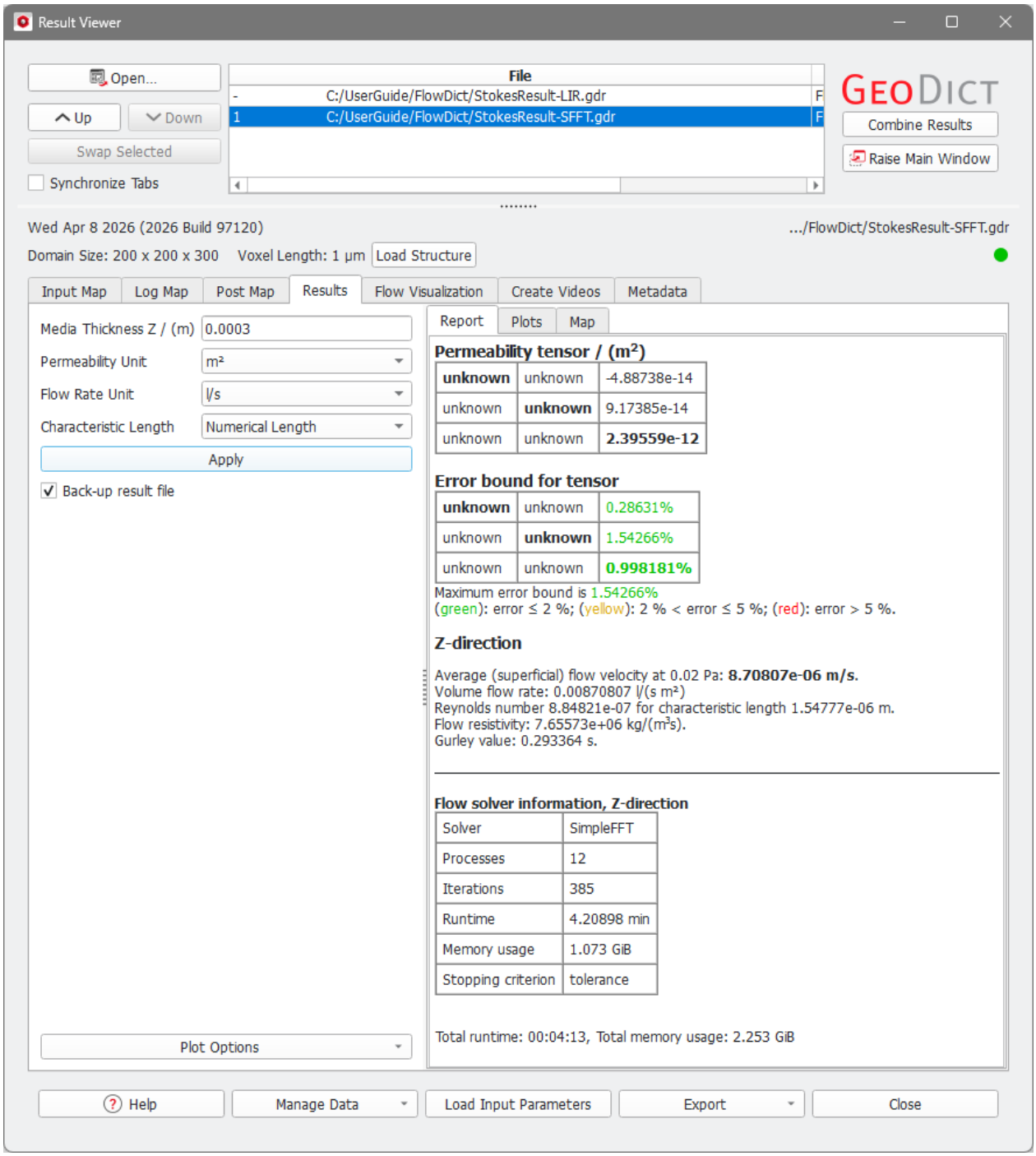


1.2.2 Results

After the solver has finished, the result file (*.gdr) is saved in the project folder and the **Result Viewer** automatically opens. Find detailed information on **Result Viewer** options in the Result Viewer user guide.

Results of the Stokes-Brinkman command

- Report 51
  Shows the computed results.
- Plots 53
  View result plots.
- Flow Visualization 55
  Load results for a graphical visualization.
- Create Videos 57
  Use a predefined macro to create a video.



Report

The **Report** tab under the **Results** tab shows the computed results.

It starts with the Permeability 20 (3x3) tensor. Below the Permeability tensor, the Error bound 38 tensor (or Tolerance 38 tensor) is located. This special tensor indicates the accuracy of the off-diagonal permeability entries. If the color code in the table is yellow or red, it might be a good idea to repeat the simulation and consider the Flow & tangential permeability 38 in the stopping criterion.

Report | Plots | Map

Permeability tensor / (m²)

6.91508e-12	1.52082e-13	-3.12919e-14
1.39242e-13	**6.70321e-12**	6.4284e-14
-4.59362e-14	9.15257e-14	**2.4009e-12**

Error bound for tensor

0.00882459%	0.047398%	0.164151%
0.00149504%	**0.00865068%**	0.0830638%
0.172667%	0.224541%	**0.25269%**

Maximum error bound is 0.25269%
(green): error ≤ 2 %; (yellow): 2 % < error ≤ 5 %; (red): error > 5 %.

Below, for each selected Computational Direction[27], detailed results are shown:

- Pressure drop / flow velocity[18]: the determined result (pressure drop or flow velocity) is always shown in bold face.
- Volume flow rate: either prescribed as boundary condition or computed in the simulation.
- Reynolds number[19]: estimated with the given characteristic length. You can change the characteristic length in the post-processing options on the left.
- Flow resistivity[21]: derived from the determined permeability.
- Gurley value[22]: derived from the determined permeability.
- Asymptotic estimate of pressure: the asymptotic result predicted by the Error Bound[38] stopping criterion

X-direction

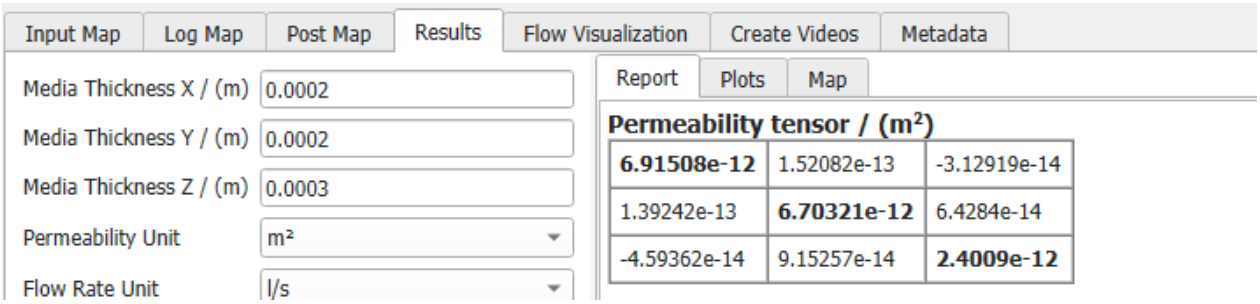
Average (superficial) flow velocity at 0.02 Pa: **3.77049e-05 m/s.**
Volume flow rate: 0.0377049 l/(s m²)
Reynolds number 6.50914e-06 for characteristic length 2.62965e-06 m.
Flow resistivity: 2.65217e+06 kg/(m³s).
Gurley value: 0.0677534 s.
Asymptotic estimate of permeability: 6.91513e-12 m²

Information about the convergence of the iterative solver is also displayed for each selected computation direction: iterations, runtime, memory usage, and applied stopping criterion.

In the last line the total runtime and total memory usage are given. Here, not only the resources used by the flow solver are taken into account, but also the time needed for the solver initialization and writing of the result files, and the additional RAM needed by GeoDict. Thus, even if only one flow direction is computed, the total runtime and total memory values are usually higher than the solver specific values.

Post-Processing

As described here[20], the computation of the permeability requires that the reference length is defined correctly. By default, the permeability is computed under the assumption that the porous media fills the whole 3D domain. If the domain contains outflow or inflow regions, you can adjust the length here by setting the appropriate **Media Thickness** in the post-processing panel on the left of the report. The **Media Thickness X** corresponds to the symbol L^1 in (27)[20], and similar for the other space directions.



You can also change the **Flow Rate Unit** or the **Permeability Unit** - this will simply change the unit in which the results are reported.

You can also change how the **Characteristic Length** used in the calculation of the Reynolds number is determined. You can find further information on the **Characteristic Length** here[19].

Click **Apply** to change the **Report**. Changing the media thickness will change permeability, Reynolds number, and flow resistivity in the **Results-Report** and the **Results-Map** according to the new length for each direction. Changing the characteristic length with change the Reynolds number.

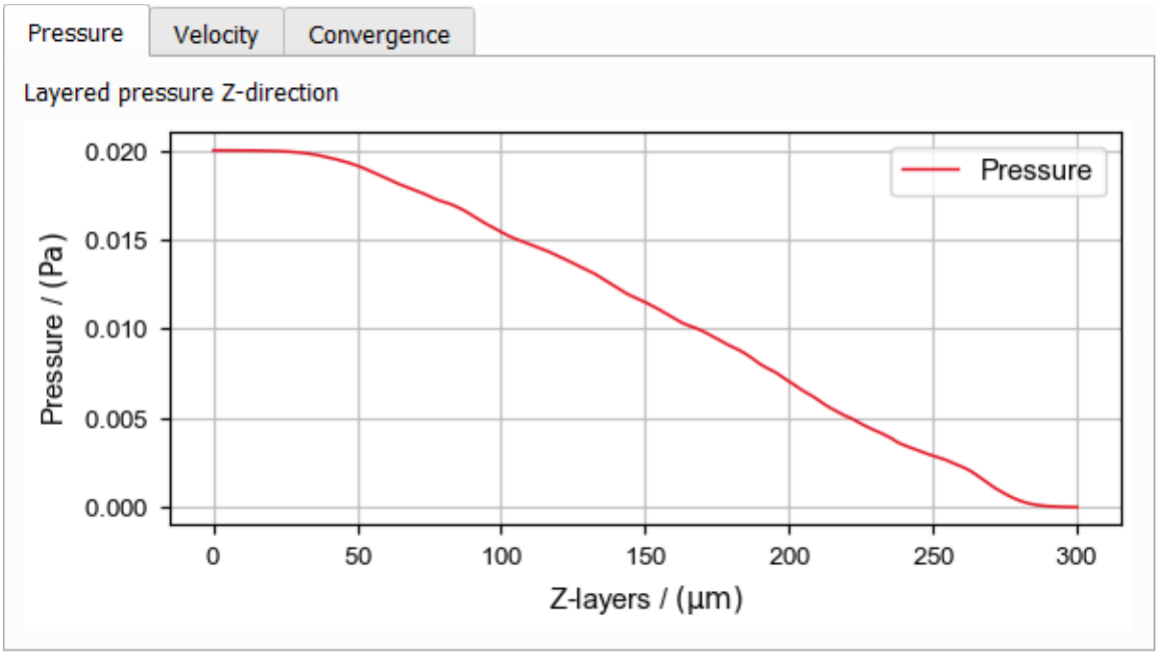
Important! Note that after clicking **Apply**, the result file changes and is not the original simulation result file anymore! If **Back-up result file** is checked, the original *.gdr file is saved in the result folder and can be restored at any time.

Plots

Under the **Plots** tab various result plots are available:

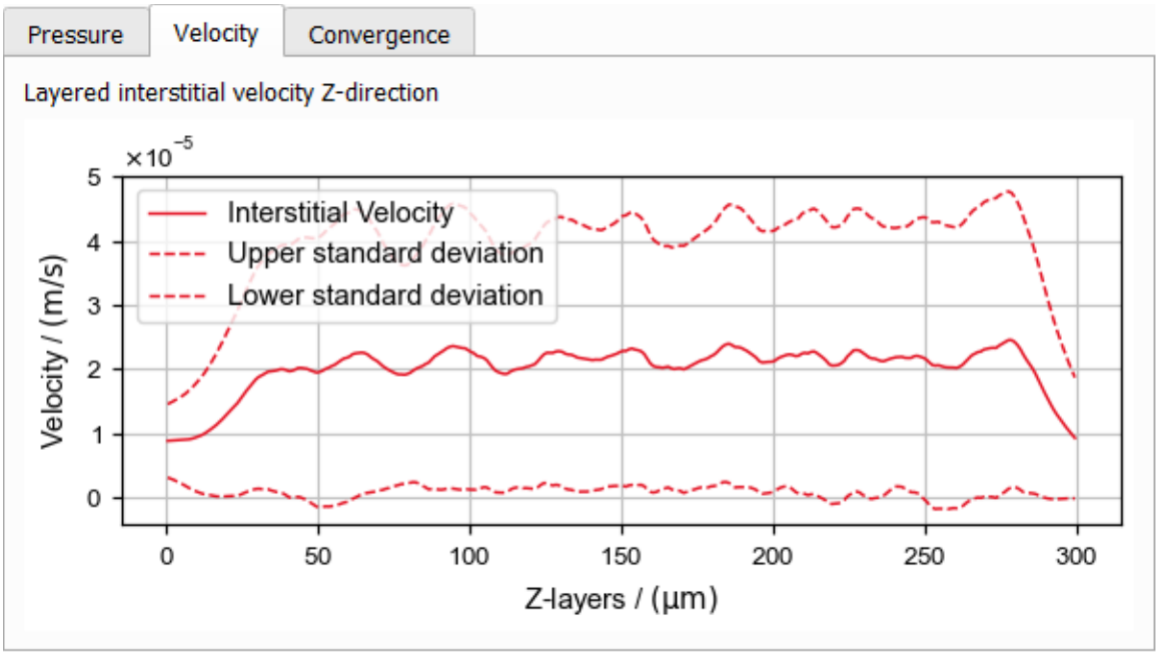
Pressure

Under the **Pressure** tab, a graph of the changes in **Pressure** across the layers is shown. There is one plot for each direction previously chosen in the **Options** dialog (see here[27]). Here, only the Z-direction had been chosen.



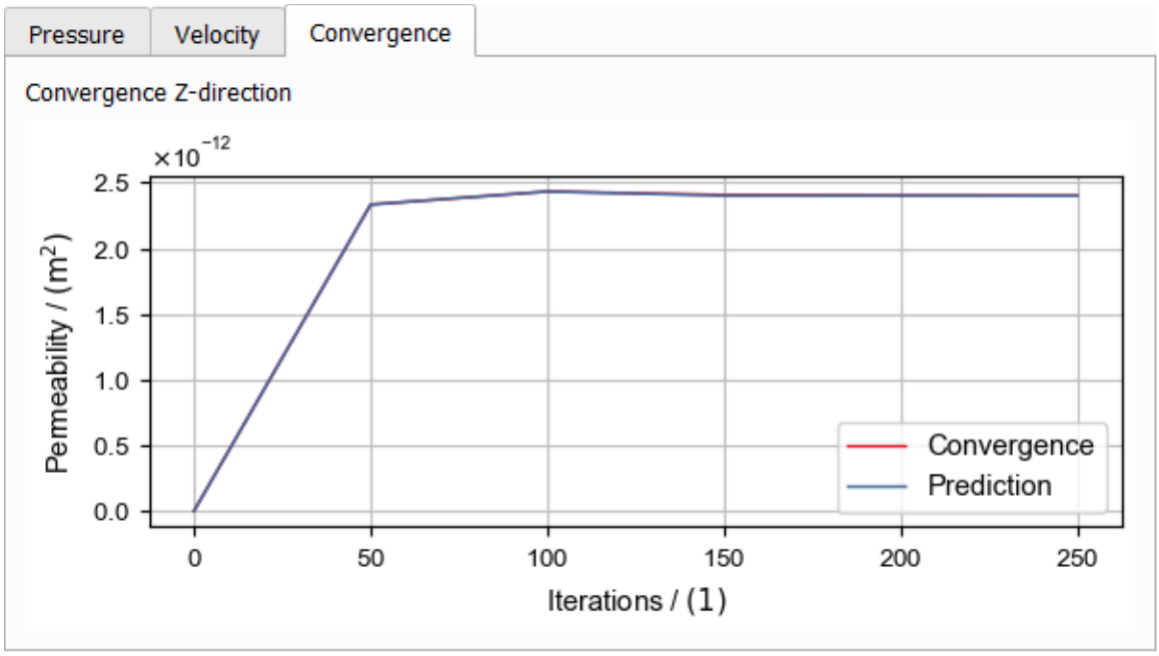
Velocity

Under the **Velocity** tab, a graph of the changes in **Velocity** across the layers is shown. Here, the velocity component in flow direction is shown. There is one plot for each direction previously chosen in the **Options** dialog (see here 27). Here, only the Z-direction had been chosen. The velocity component in the chosen direction is averaged over the pore voxels for each layer in this direction.



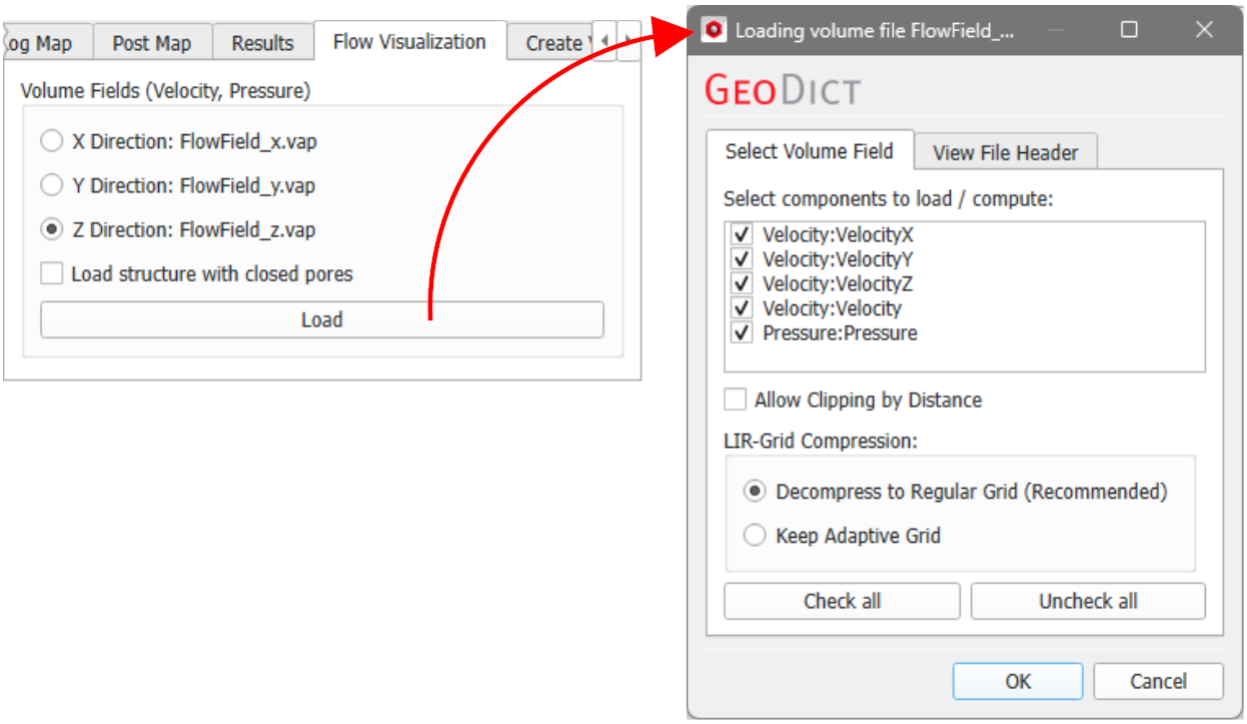
Convergence

The **Plots** tab also includes a **Convergence** tab with a chart of the **Permeability** (or **Mean Velocity** or **Flow Rate**) at given **Iteration** values. See the convergence graphs of the iterative solver for each of the calculated directions previously chosen in the **Options** dialog (see here 27).



Flow Visualization

Select the **Flow Visualization** tab to load the computed flow field (velocity and pressure, *.vap). Dependent on the selected **Computational Direction(s)**, result fields can be available for flow in **X-**, **Y-**, and/or **Z-Direction**.



Select the flow direction, then click **Load**. A dialog pops up, that let's you select which of the stored field components **VelocityX**, **VelocityY**, **VelocityZ**, **Velocity**, and/or **Pressure** you want to load and visualize.

Check **Allow Clipping by Distance** if you want to use this feature in the visualization. If selected, an additional data field is computed and loaded that contains the Euclidean distance to the solid structure. You can use this data field, e.g., to print arrows only with a certain distance from the solids. If you do not need this feature, do not select it, as it requires additional RAM to store the distance field.

Result files computed with the LIR solver are stored using the adaptive grid of the LIR solver (**LIR-Grid Compression**). Select if you want to interpolate the data during import to the regular voxel grid or if you want to keep the adaptive grid:

- **Decompress to Regular Grid:** More RAM is needed to load the data, and you cannot visualize the adaptive grid. This option enables the visualization with Streamlines, Arrows, and Schlieren.
- **Keep Adaptive Grid**: Less RAM is needed to load the data. You can visualize the adaptive grid. You cannot use Streamlines, Arrows, or Schlieren to visualize the results.

The structure that the solver used internally is loaded if **Load structure with closed pores** is checked. This structure shows which pores have been closed due to prior geometry analysis.

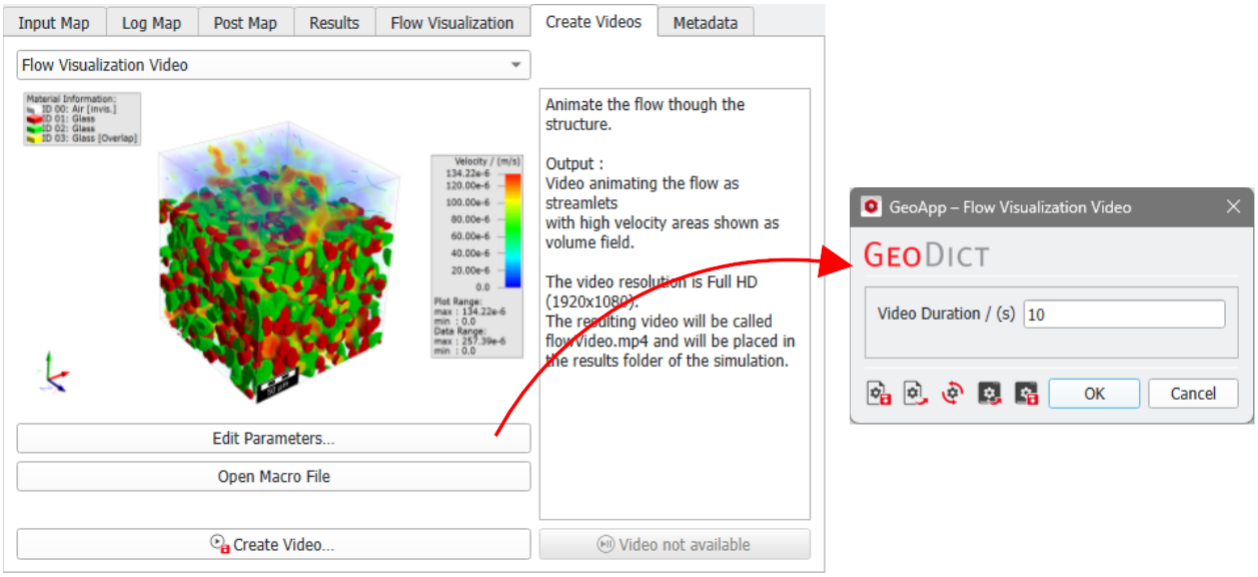
Visualization Options

- Volume Field: Pressure drop or velocity are visualized by a 3D color field - you can clip by threshold values or use transparency to view inside.
- Streamlines: Visualize the flow through the structure with streamlines or animated streamlets.

- Arrows: Visualize the velocity vector classically with arrows.
- Schlieren: This option uses Line Integral Convolution (LIC) fields to visualize vortices.

Create Videos

The **Create Videos** tab provides a predefined Python macro to create a video of the structure with animated flow streamlets with high velocity areas shown as volume field. Click **Edit Parameters** to change the **Video Duration** in the **Flow Visualization Video Parameters** dialog.



Click **Create Video** to start the video generation. This generates a video (*.mp4), GeoDict result file (*.gdr), and a results folder inside the corresponding result folder of the FlowDict simulation.

1.3 Navier-Stokes(-Brinkman)

This command solves a boundary value problem to determine the stationary flow through the media. If all material IDs present in the 3D structure model describe either fluid or solid materials, the Navier-Stokes 7 equation is solved. If some material IDs represent unresolved porous materials, the Navier-Stokes-Brinkman 9 equation is solved.

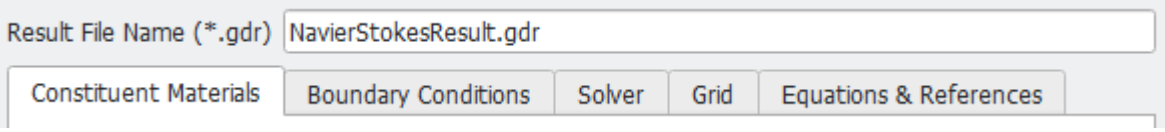
In this command, the flow boundary conditions are aligned with a space direction - one side of the domain is the inflow area and the opposite side of the domain is the outflow area. As boundary conditions, you can prescribe flow rate, inflow velocity, or pressure.

The Navier-Stokes-Brinkman command

- Options 58
  How to set the input parameters of the command.
- Results 84
  View the simulation results.

1.3.1 Options

Select **Navier-Stokes(-Brinkman)** from the pull-down menu and click the **Option**'s **Edit...** button to open a dialog to enter or edit the simulation options. Choose a **Result File Name** for the result file (*.gdr) fitting to your current project.



The parameters are organized under the tabs: **Constituent Materials**, **Boundary Conditions**, **Solver**, and **Grid**. The **Equations & References** tab gives a summary of the used equations and symbols which are explained in more detail here 9.

Tabs of the Navier-Stokes-Brinkman dialog

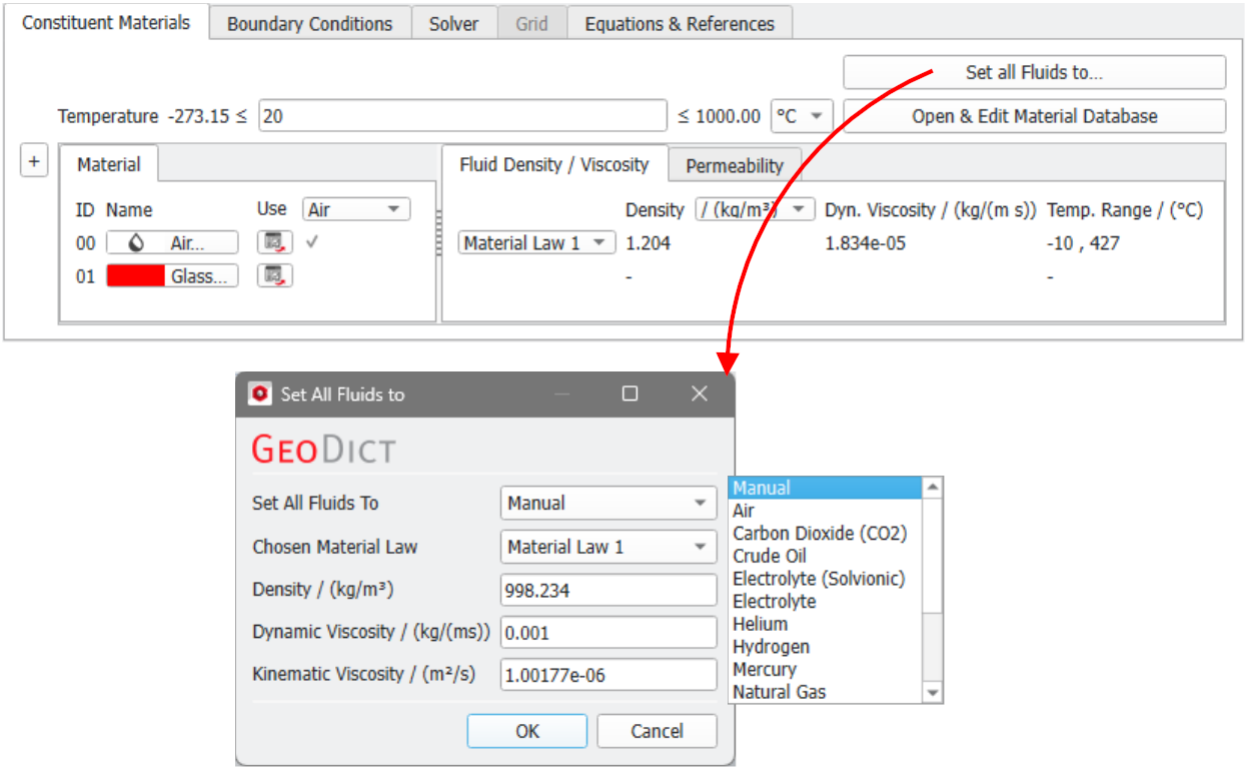
- Constituent Materials 58
  Define the materials in the structure.
- Boundary Conditions 61
  Set up the boundary value problem that describes your experiment.
- Solver 70
  Select a flow solver.
- Grid 83
  Adjust the computational grid.

Constituent Materials

For the flow solver's computation, the materials in the structure and the fluids flowing through it need to be defined under the **Constituent Materials** tab, so that their physical properties

parameters are taken into account.

To set the fluid flowing through all pore and porous materials, click **Set all Fluids to** and select it from the pull-down menu.

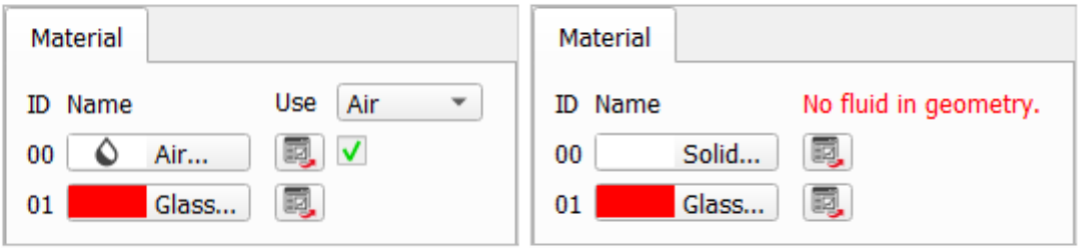


If one of the predefined fluids is used, the values for **Density**, **Dynamic Viscosity**, and **Kinematic Viscosity** are taken from the GeoDict Material Database. They are dependent on the given **Temperature** value and the **Chosen Material Law**. If **Manual** is selected, these values can be entered manually.

Any values entered manually, e.g., for a special type of oil, can also be added to the material database through the **Open & Edit Material Database** button. Additionally, different properties for the same material can be saved as different **Material Laws** and can then be selected from the pull-down menu in the **Set all Fluids to** dialog. **Material Law 1** is the default shipped with GeoDict. Click here for more information on editing, expanding, and using the GeoDict Material Database.

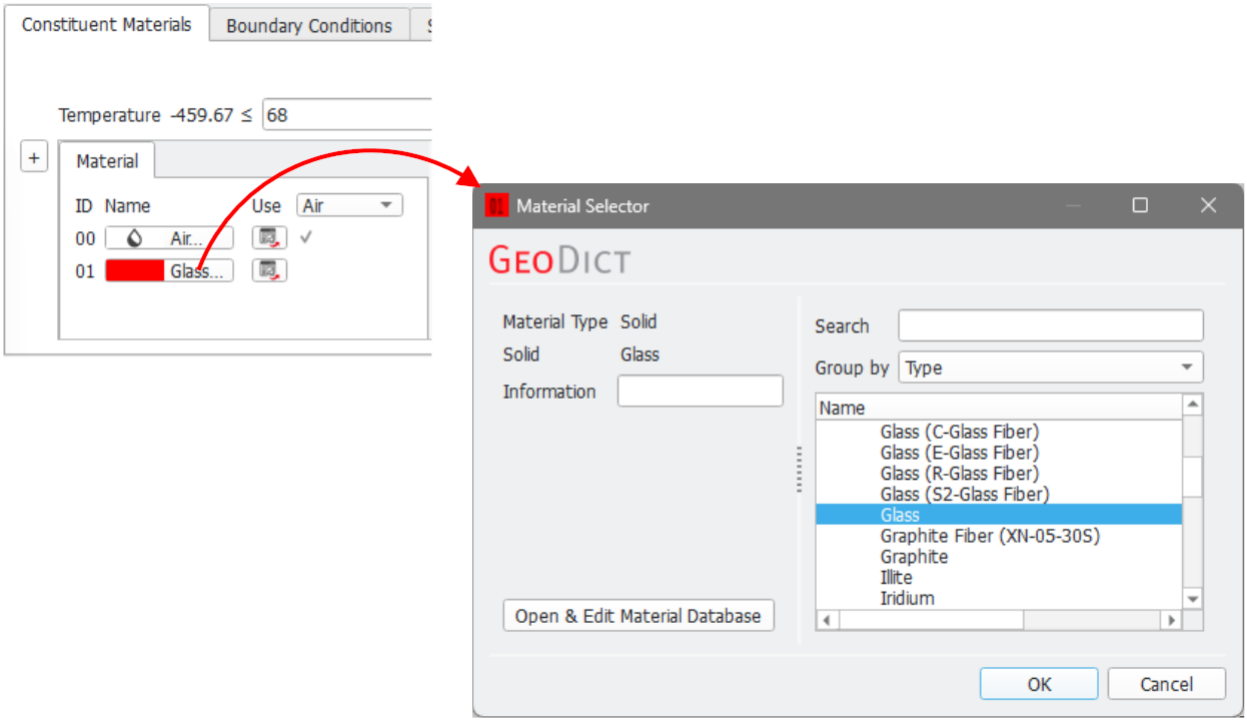
The **Temperature** is selectable in Kelvin (K), Celsius (°C), and Fahrenheit (F), and has a default value of 293.15 K (20.0°C, 68 F).

Under the **Material** subtab, select the fluid used in the simulation from the **Use** pull-down menu. If other fluids are contained in the structure, they will be treated as immobile (solid) during the simulation. If the structure contains no fluid material, **No fluid in geometry** is displayed in the top of the **Material** tab. Select a material for the pore phase by clicking on the corresponding material button.

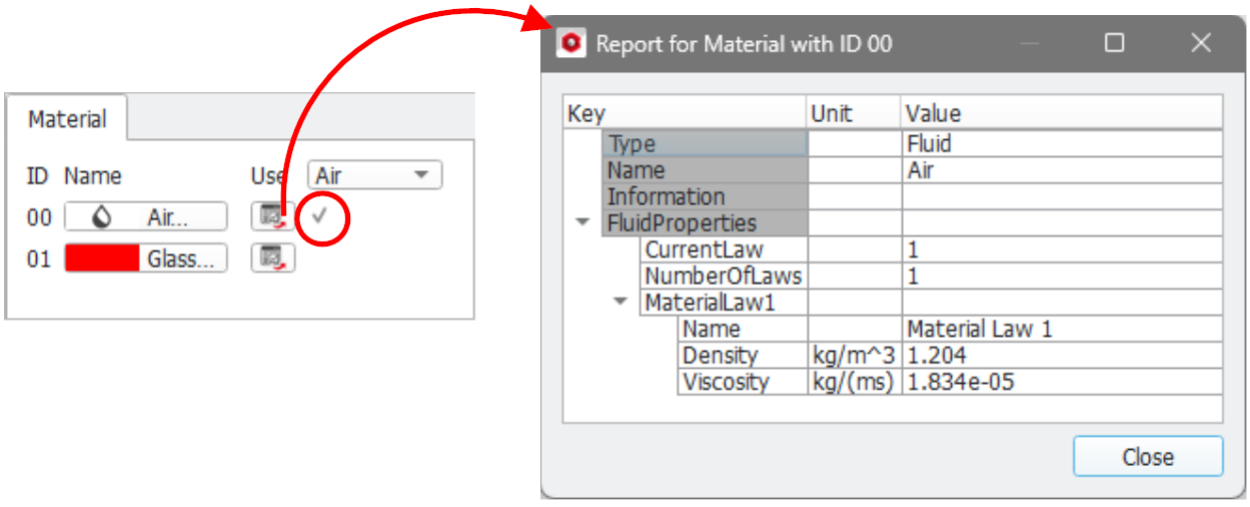


The density and the dynamic viscosity of this fluid, at the selected temperature, is shown under the **Fluid Density/Viscosity** subtab.

The **Material** subtab shows also the material **IDs** assigned to the materials (00, 01, 02, etc.). For each material ID, you can open the **Material Selector** dialog and select a material by clicking on the corresponding material buttons.

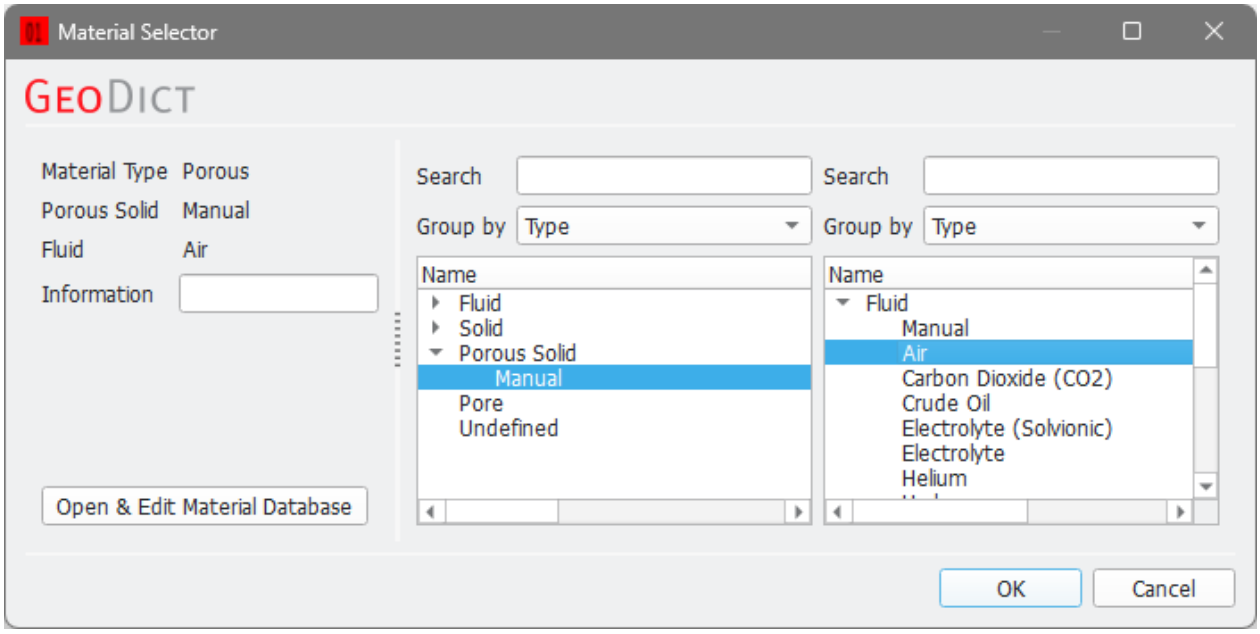


Click the **Material Report** button to see the relevant parameters for the material with a selected ID.

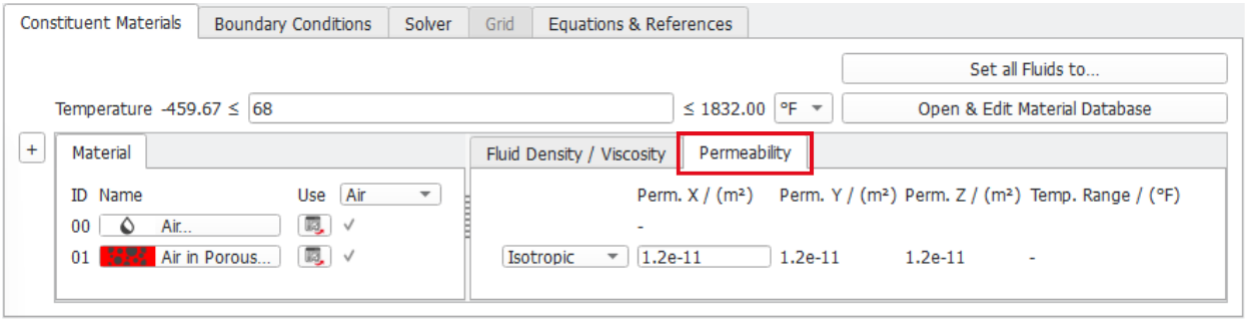


Fluid flow can happen in pores and porous materials, but not in solid materials. The check sign next to the **Material Report** icon marks all materials in which flow happens.

For porous voxels, set the **Material Type** (in the Material Selector dialog) to **Porous** and choose the fluid that occupies the pores.



The permeability of porous voxels is set under the **Permeability** subtab. You can define if the material is isotropic (same permeability in all flow directions) or anisotropic (permeability depends on flow direction). When the material is isotropic, the permeability entered in the X-direction is used in all directions. For an anisotropic material, the permeabilities in the three Cartesian directions are needed.



The local orientation of porous materials with anisotropic properties is not considered. You must select the **SimpleFFT** solver if anisotropic permeabilities are entered.

Boundary Conditions

To simulate the flow through the domain, GeoDict solves a boundary-value problem. In the fluid-filled part of the domain, the flow is governed by a differential equation. The boundary values describe what happens at the boundaries, and are a necessary and important part of the problem description. Boundaries are the six faces of the cuboid (X-, X+, Y-, Y+, Z-, Z+), and the pore-solid walls within the structure.

For the six cuboid faces, FlowDict distinguishes between the domain faces in the

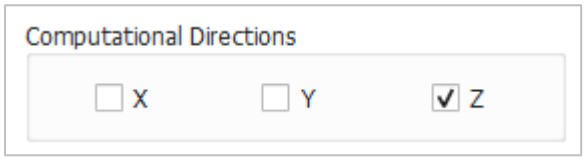
computational direction (also called flow direction or longitudinal direction) and the domain faces in the two other space directions (called transversal or tangential directions). The computational direction is the direction in which pressure drop and flow rates are measured.

Computational Directions

As **Computational Directions**, select the directions for which you want to calculate the flow. To obtain the full 3x3 permeability matrix and a pressure drop value for each direction in the results file, select all three directions.

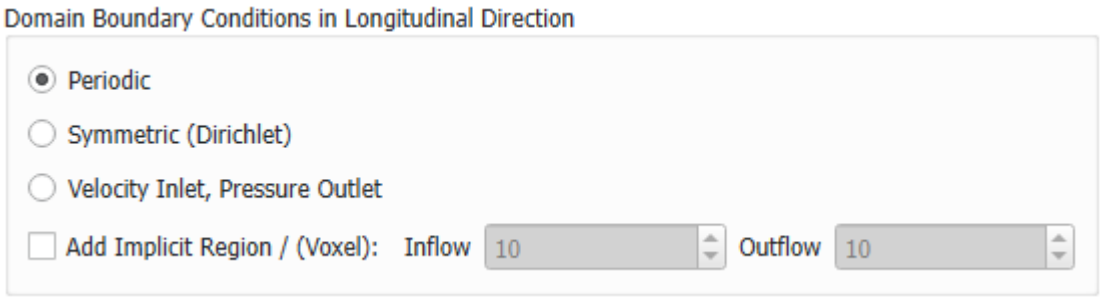
For every selected direction, a boundary value problem is solved in which the **Domain Boundary Conditions in Longitudinal Direction** and the corresponding **Experiment Input/Output** are applied in the computational direction, and the **Domain Boundary Conditions in Transverse Directions** are applied on the other two space directions.

Any directions not selected will not be calculated and will be marked as 'unknown' in the permeability matrix in the **Report** tab of the result file.



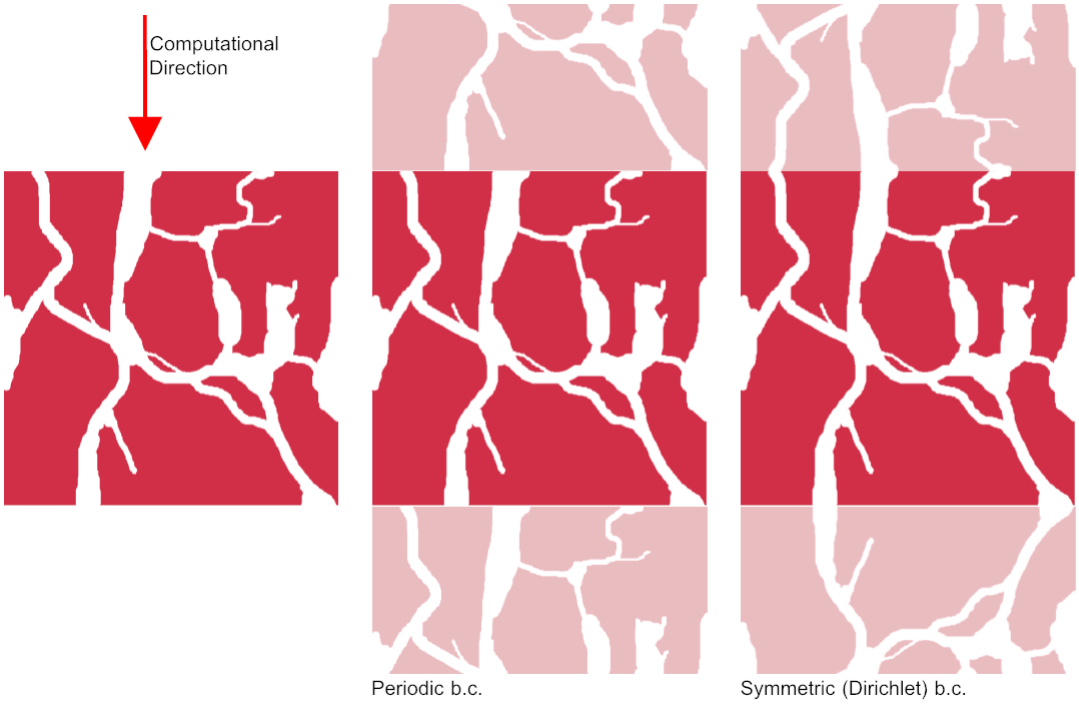
Domain Boundary Conditions in Longitudinal Direction

These boundary values are applied in the **Computational Direction**. Select either **Periodic**, **Symmetric (Dirichlet)**, or **Velocity Inlet, Pressure Outlet** conditions.



Use **Periodic** boundary conditions for periodically generated structure models and for non-periodic structures with high porosity. The structure is internally repeated periodically at the domain boundaries in the computational directions.

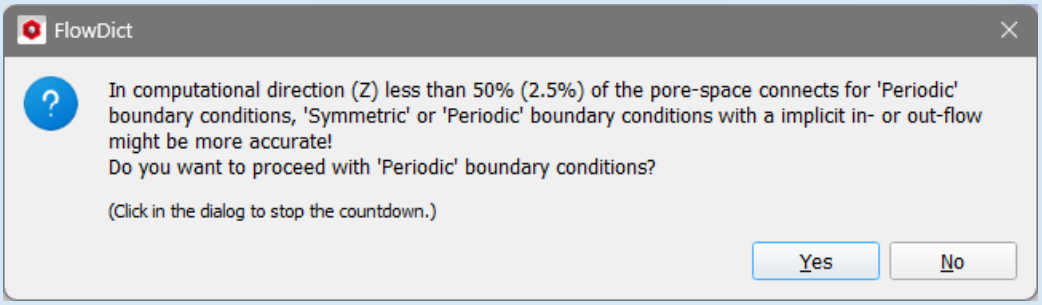
Use **Symmetric** boundary conditions for non-periodic structures if the percentage of solid voxels is high at the inlet. The solver treats the structure as mirrored at the domain boundaries in the computational directions.



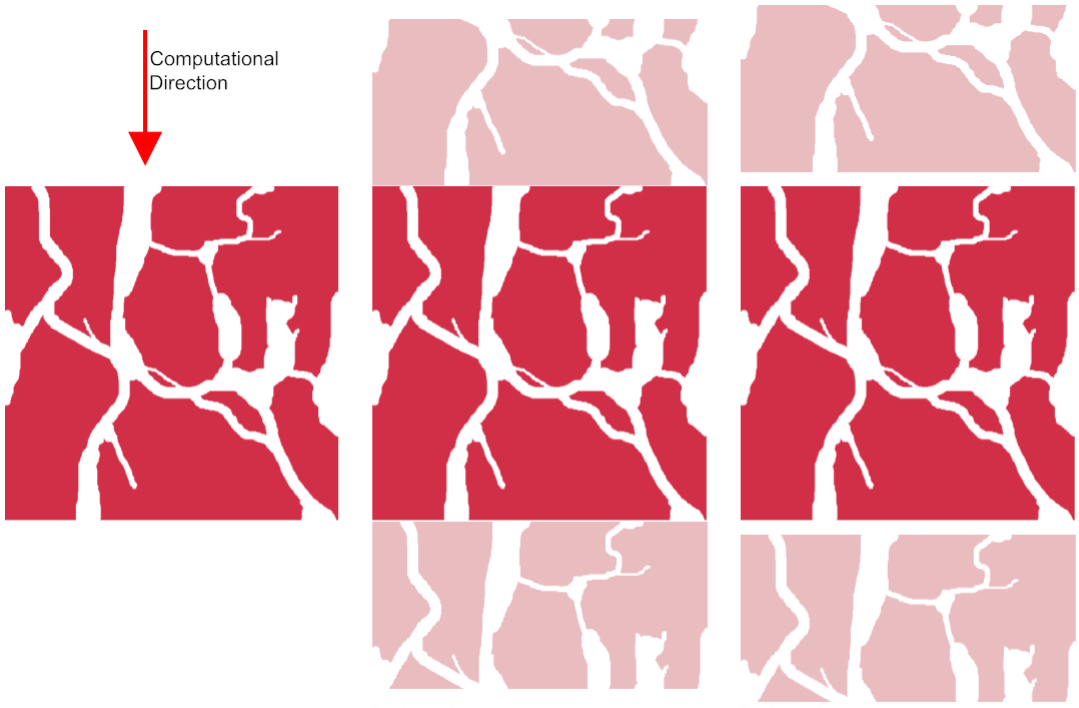
The option to **Add** an **Implicit Region** is not available for **Symmetric** boundary conditions. The size of the implicit region in the computational directions is given in voxel, where the default added implicit inflow and outflow are 10 voxels, respectively.

Know how! The term implicit means that the region is only added by the solver internally. Thus, these regions do not contribute to the computed permeability and flow resistivity.

Note! A warning may appear when trying to run flow computations without adding an inflow region when the pore space at the domain boundaries in the computational directions is less than 50%.

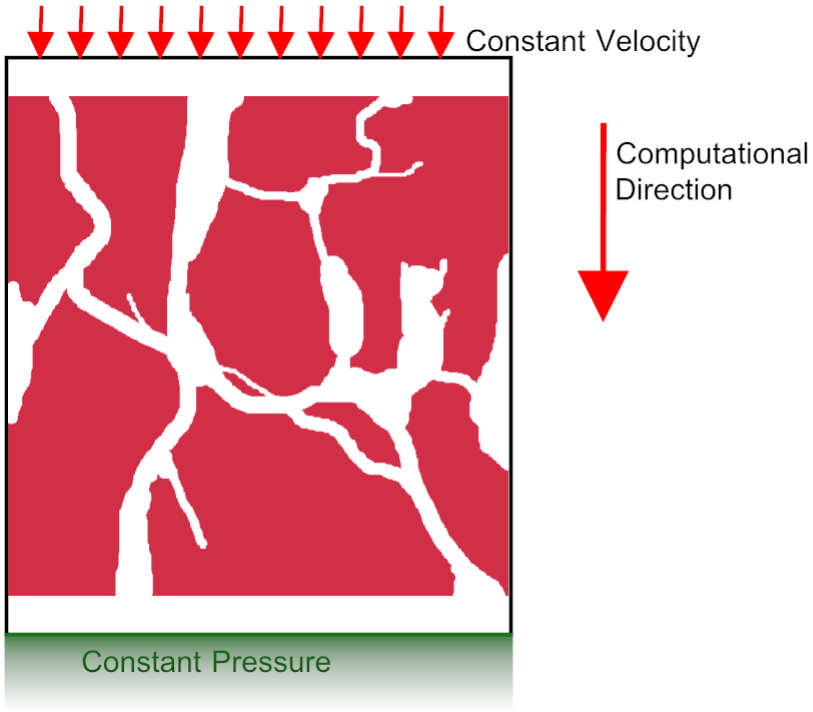


The inlet and outlet are essential to avoid the possibility of closing the flow channels when the structure is periodically repeated. In the example shown below choosing periodic boundary conditions in the direction of the flow without adding an implicit region results in the flow channels being artificially closed.



To open the channels and enable flow, you can either add an inflow and/or outflow region, or select symmetric boundary conditions. However, when possible, we recommend using periodic boundary conditions, as these require less computational memory and take less time to run.

The **Velocity Inlet, Pressure Outlet** (VinPout) boundary conditions apply a constant flow velocity at the inlet and a constant pressure at the outlet. These boundary conditions can only be applied if the structure contains empty inlet and outlet regions. A constant flow velocity cannot be applied directly on a solid surface, as this would lead to an unphysical solution. Therefore, you either have to assure that the current structure already contains such regions or you have to add those regions for the flow computation by additionally selecting **Add Implicit Region**.



Note, that selecting **Velocity Inlet, Pressure Outlet** automatically disables the EJ solver in the **Solver** tab.

Important! In some cases, it is very important to choose a large inflow and outflow region. We recommend using at least 100 voxels for the inflow and outflow regions for weaves and for fast flows computed with the Navier-Stokes equations.

This can be done either by checking **Add implicit region** and entering values or by adding the region *explicitly* to the 3D structure using ProcessGeo - Embed. If the region is added explicitly, it is included in the 3D *.vap file and considered as part of the structure during post-processing. If it is added implicitly, the region is not contained in the resulting *.vap file, and not considered as part of the structure during post-processing.

Experiment Input / Output

In the **Experiment Input / Output** panel select which property is prescribed in the computational direction. This will also define the property that will be computed. You can prescribe the **Pressure Drop** and obtain the **Mean Velocity** as result. Alternatively, you can prescribe the **Mean Velocity** or the **Flow Rate** and obtain the **Pressure Drop** as result.

This selection corresponds to the typical experimental setup, where you either apply a constant pressure drop over the media and measure the through flow rate or you apply a fixed pump flow rate and measure the resulting pressure drop over the media.

The **Pressure Drop** is the difference between the inflow and outflow pressures.

The **Mean Velocity** is the average speed of the flow in the positive direction, i.e., from Z- to Z+ if the Z-direction is the computational direction.

The **Flow Rate on Flow Area** is a volumetric flow rate. It is the fluid volume which passes per unit time through a given area.

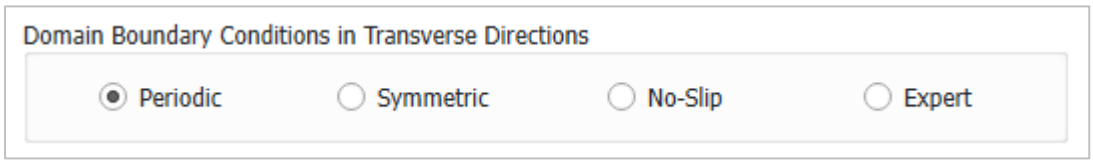
For each property, the desired unit can be selected from the pull-down menu.

Since the Navier-Stokes 7 (-Brinkman 9) equation describes non-linear flow, only the **Pressure Drop** can be used as input parameter when using **Periodic** or **Symmetric (Dirichlet)** domain boundary conditions in longitudinal direction 62. The other two properties are grayed out.

For **Velocity Inlet, Pressure Outlet** boundary conditions, you must prescribe the **Mean Velocity** or **Flow Rate**.

Domain Boundary Conditions in Transverse Directions

The **Domain Boundary Conditions in Transverse Directions** can be **Periodic**, **Symmetric**, **No-Slip**, or **Expert**.



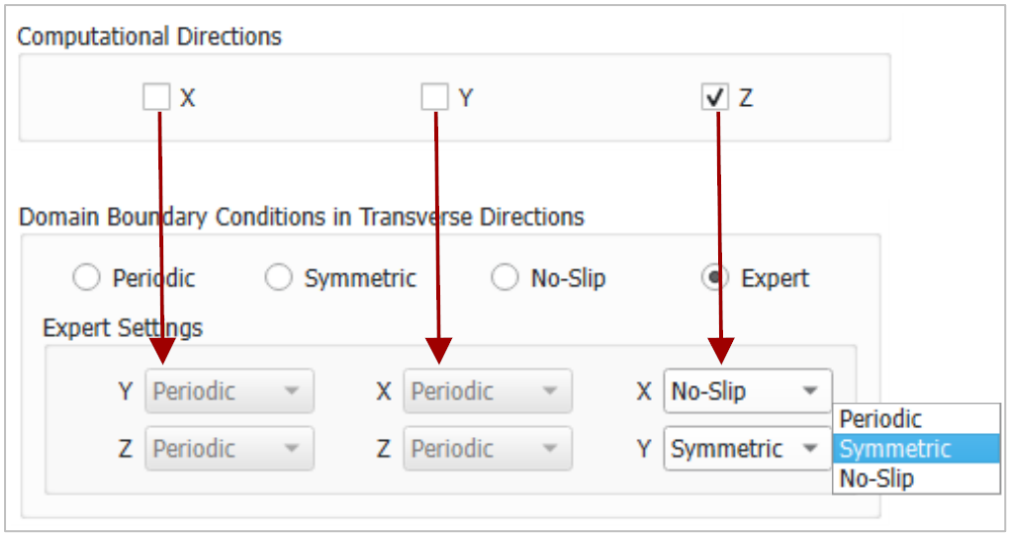
With **Periodic** boundary conditions, the structure is repeated in the transverse directions.

With **Symmetric** boundary conditions, the structure is mirrored at the domain boundary. The LIR and EJ solver solve the symmetric boundary condition by using

Neumann (or zero flux) condition at the domain boundary.

With **No-Slip** (or Encase) boundary conditions, the domain boundary is treated as closed by solid material. When **No-Slip** is used, the solver internally adds a one-voxel layer in the required direction and solves with periodic boundary conditions. This is effectively equivalent to solving the structure with casing at both sides in the transverse direction. So, the structure size in the corresponding transverse direction becomes enlarged by one voxel internally.

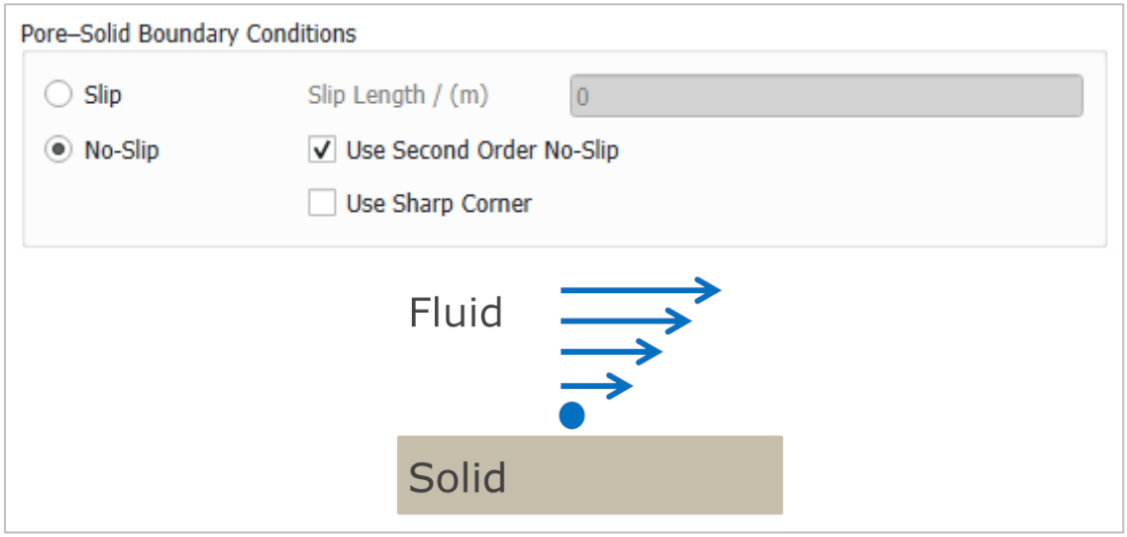
With **Expert**, it is possible to define different boundary conditions for each side in each computation. The boundary conditions in the two directions transverse to the flow can also be different by checking **Expert** boundary conditions. For example, when the fluid is chosen to flow in the Z-direction, the boundary conditions could be chosen to be **No-slip** in X-direction and **Symmetric** in the Y-direction.



Pore-Solid Boundary Conditions

The **Slip Length** allows to include sliding effects in the flow simulation. Sometimes the permeability of gases can be somewhat different from the permeability of liquids in the same media. One difference is attributable to "slippage" of gas at the interface with the solid when the gas mean free path is comparable to the pore size.

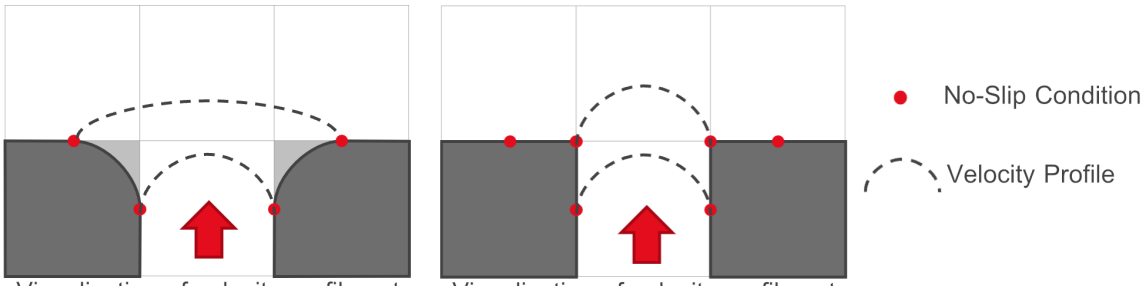
The default setting **No-Slip** corresponds to a flow velocity of zero along the structure.



Two settings are available for the **No-Slip** boundary condition. **Use Second Order No-Slip** is activated by default. This sets the tangential velocity to zero at the center of the voxel surfaces and uses a second order approximation of the velocity field. The first order setting, which is used when the box is left unchecked uses a linear approximation (Harlow-Welch, 1965 149).

Use Sharp Corner can be used for digital rocks where permeability is usually overestimated or cases where voxels represent exact geometry. Activating it sets the tangential velocity to zero at the voxel corners. When the box is unchecked the default

Round Corner is applied, which should be used for structures with round obstacles like fibers or grains.

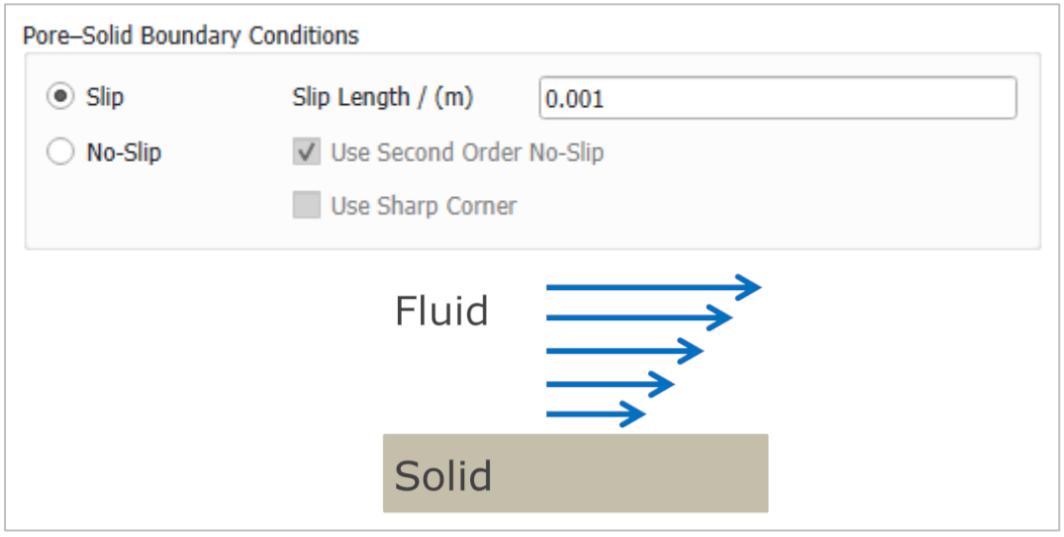


Visualization of velocity profiles at round corners

Visualization of velocity profiles at sharp corners

Slip with a non-zero **Slip Length** simulates the sliding of the fluid along the solid, increasing the fluid mean flux and thus, the permeability of the structure. This option might be used when it is realistic for a given physical material. Currently, the same slip length value must be set for all materials in the structure.

In earlier releases, this feature worked correctly for axis aligned walls only, but since GeoDict 2020 the expression of the slip velocity, which assumes the slip velocity proportional to the shear stress at the surface, is reformulated for different velocity components when the angle of fiber surface is known, and reimplemented in the flow solver. Thus, direct simulation of the slip flow is possible.



$$\vec{n} \cdot \vec{u} = 0 \text{ on } \Gamma$$

(34) No flow into solid

$$\vec{t} \cdot \vec{u} = -\lambda \vec{n} \cdot \nabla(\vec{u} \cdot \vec{t}) \text{ on } \Gamma$$

(35) Slip flow along solid

Here, $\vec{u}$ is the fluid flow velocity as it was introduced in Darcy's law[19] and the

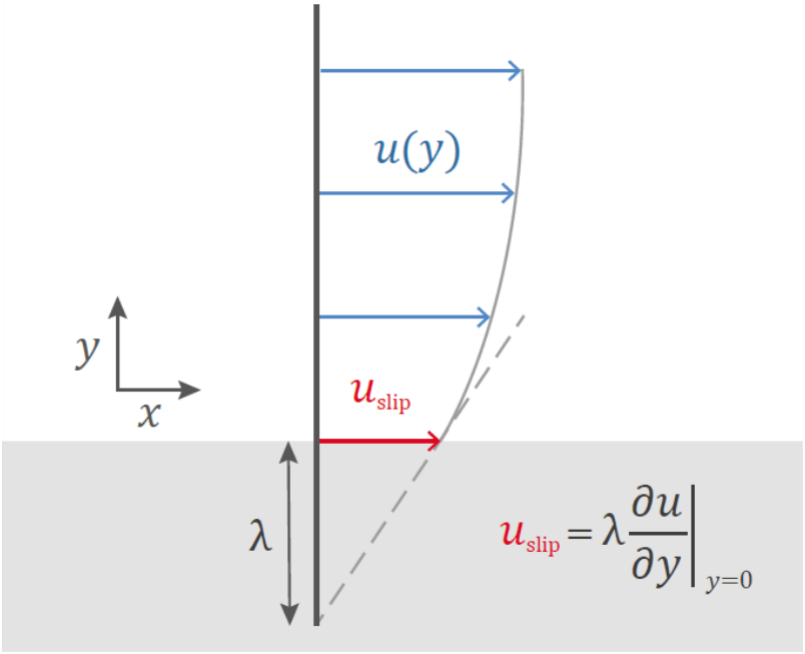
Stokes[8] equation, $\vec{n}$ is the normal direction to the solid surface, Γ is the fiber surface, λ is the **Slip Length** and $\vec{t}$ is any tangential direction with $\vec{t} \cdot \vec{n} = 0$. For the same pressure drop, the computed velocities for slip boundary conditions are higher than for no-slip boundary conditions. Conversely, for a given velocity or equivalently, a given mass flux, the pressure drop is lower when computed with slip boundary conditions.

For a straight channel structure, the two slip flow equations above become:

$$u = \lambda \frac{\partial u}{\partial y}$$

(36) Slip flow in straight channels

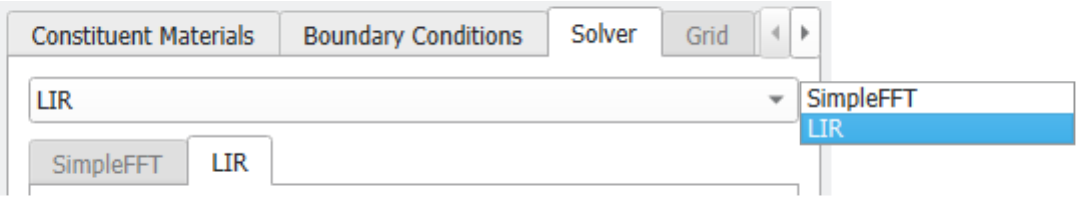
In this following example figure, the gray solid serves as a wall for a straight channel. To obtain the flow velocity along the solid (u_{slip}) a coordinate system is defined by the solid orientation.



For a **Slip Length** of zero ($\lambda = 0$), the graph that describes the flow velocity in dependence of the distance to the solid surface, would start at the surface of the solid. However, for a positive **Slip Length** ($\lambda > 0$) it is shifted into the solid by λ . Thus, the flow velocity u_{slip} is defined.

Solver

Two different flow solvers[12] are available for solving the equation. Select the solver in the pull-down menu to activate the corresponding tab below.



The LIR solver requires the least numerical resources (RAM) and is therefore selected by default. Often, the LIR is also the fastest solver, but that may depend on the parameter selection and the 3D structure model.

Parameters of the Solver tab

- Simulation Stopping Criterion 71
  Define when the computation is stopped.
- Restart Save Interval 75
  Configure an auto-save interval.
- Parallelization 75
  Parallelize the FlowDict computations.
- Restart from a *.gdr File 75
  Continue a previous FlowDict computation.
- Discard PDE Solver Files 76
  Delete intermediate files.
- Advanced Options 76
  Additional solver options.

Simulation Stopping Criterion

The equations are solved by an iterative approach.

Iterative Solver

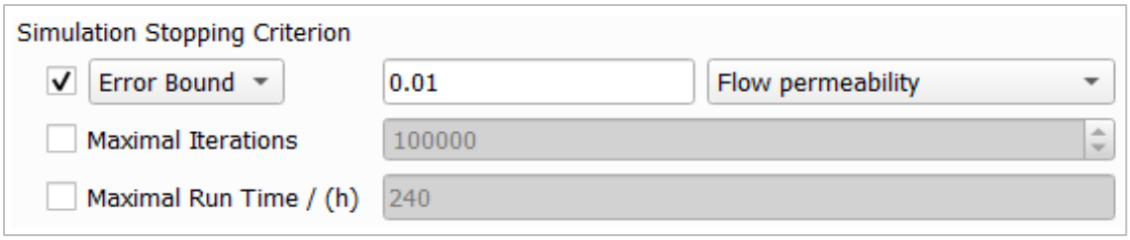
The basic idea of an iterative method is to

1. Start with some initial guess for the unknown values
2. Improve the current values in each iterative step
3. Repeat the iterative process until one of the stopping criteria is reached.

The iterative process is controlled by setting the values and activation for **Error Bound, Tolerance, Residual**, **Maximal Iterations**, and **Maximal Run Time (h**). The stopping criteria apply for each computational direction individually.

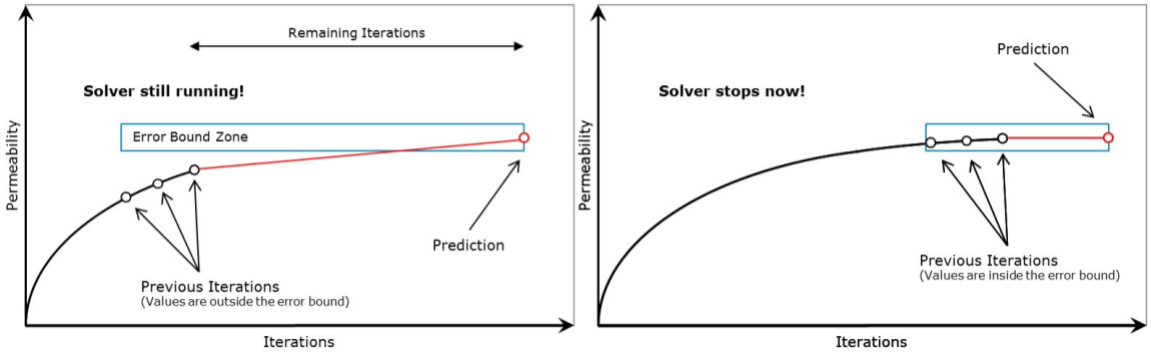
If multiple stopping criteria are selected, the first criterion that is reached causes the solver to stop.

The stopping criterion that has been reached can be viewed in the Result Viewer of the GeoDict result file (*.gdr) under the **Results Report** tab.



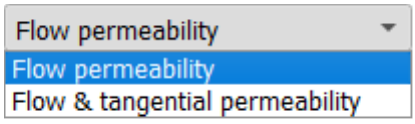
Error Bound (LIR and SimpleFFT)

The stopping criterion **Error Bound** uses the result of previous iterations, and predicts the final solution based on linear and quadratic extrapolation. The solver stops if the relative difference between computed and predicted solution is smaller than the specified error bound. The stopping criterion recognizes oscillations in the convergence behavior and prevents premature stopping at local minima or maxima. A damped convergence curve is fit through the oscillating curve and the solver stops then regarding the damped convergence curve.



If the Krylov method (under LIR - Advanced Options) is activated, the definition of **Error Bound** is somewhat different. Here, no prediction is made, instead the continuity between neighboring cells and the conservation of mass is checked. The maximal value is normalized by the mean flow and if this value is smaller than the **Error Bound** the simulation stops.

It is possible to choose if the stopping criterion should consider the permeability only in flow direction or in both flow and tangential directions. For flow simulations where the permeability in flow direction converges much faster than the permeability in tangential directions, it is recommended to change from **Flow permeability** to **Flow & tangential permeability**.



Tolerance (LIR and SimpleFFT)

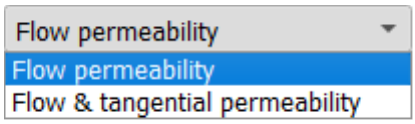
The **Tolerance** stopping criterion looks for stagnation of the method when the process becomes stationary, i.e., the improvement in the permeability value becomes extremely small from iteration to iteration.

In each iteration, the solver checks for the current computed value x_c against the values x_{c-i} of the last 100 iterations if there are any changes.

$$\max_{i \in [1,100]} \frac{x_c - x_{c-i}}{x_{c-i}} < \text{tolerance} \quad (37)$$

The computation is stopped if the maximal relative change is smaller than the value entered for **Tolerance**. When there is doubt about the quality of the solution, decrease the **Tolerance** value by a factor of ten. The drawback of this criterion is that the solver sometimes might stop too early in case of slow convergence rate or at local extrema of oscillatory convergence curves.

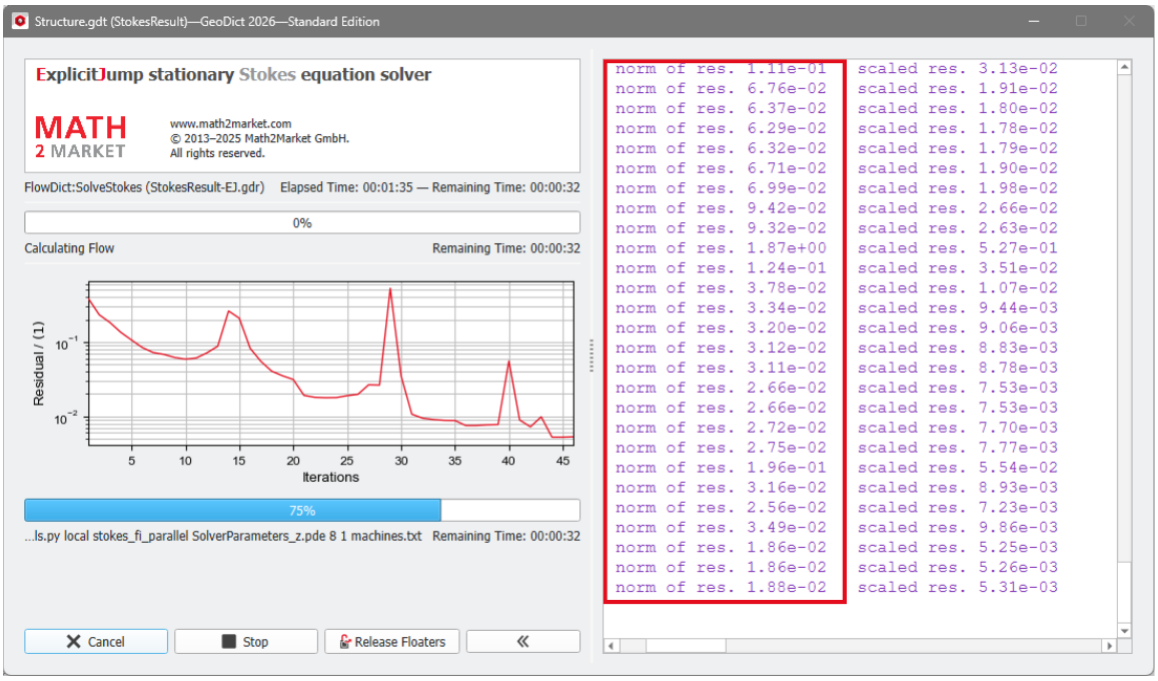
It is possible to choose if the stopping criterion should consider the permeability only in flow direction or in both flow and tangential directions. For flow simulations where the permeability in flow direction converges much faster than the permeability in tangential directions, it is recommended to change from **Flow permeability** to **Flow & tangential permeability**.



Residual (SimpleFFT)

When the **Residual** stopping criterion is used, the iteration is stopped if the solution satisfies the equation up to the required accuracy.

By setting the stopping criterion to **Residual**, the computations terminate as soon as the relative norm drops below the selected residual threshold. The relative norm of the Schur Complement residual is computed and displayed in the console window during the calculations. The console is visible by clicking the double arrow button on the lower right corner in the progress dialog.



The recommendation to choose **Tolerance** or **Residual** for the **EJ** solver is based on the structure's porosity. Both give similar results for highly porous structures. For dense structures, if using the Schur Complement **Residual**, the relative norm of the residual may be small even though the correct permeability has not been reached. So, when in doubt, use the **Tolerance** criteria – the default option.

Maximum Iterations and Maximum Run Time (all solvers)

Use the **Maximum Iterations** value or the **Maximum Run Time (h)** stopping criteria causes the solver to stop if the maximal number of iterations and/or the maximal runtime (in hours) is exceeded.

When the solver stops because one of these criteria has been reached, no guarantee on the quality of solution can be given. In this case, a warning is printed into the report. The following possibilities might help:

- Check the corresponding .log file to see how large the residual values and the permeability values are. If the permeability values are in reasonable range and they do not have changed during the last iterations, you may decide to use the current result.
- Double-check the structure and parameter values. Unphysical parameters or too rough resolution of the structure (leading, e.g., to artificial unconnected components) can cause an iterative solver to fail.

Know how! We recommend to use the **Error Bound** criterion for the **LIR** and **SimpleFFT** solver. This stopping criterion approximates the "relative deviation from the final solution".

Restart Save Interval

The calculations run by the solvers can be restarted from saved intermediate result files and the interval between auto-saves can be configured from the value entered in **Restart Save Interval (h)**.

Parallelization

The simulation process can be parallelized to speed-up the computation process. It depends on your license, how many parallel processes you may use. The solver benchmarks illustrate the speed-up that you can achieve.

The **Parallelization Options** dialog box opens when clicking the **Edit...** button, to choose between **Sequential**, **Parallel (Shared Memory)**, **Automatic Maximum of Threads** and (if available) **Cluster**.

For details on how to set up and run parallel computations, consult the High Performance Computing chapter.

Restart from a *.gdr File

In some situations, it may be useful to re-use previously computed results and save a great amount of time by restarting the process from an existing GeoDict result file (*.gdr). Typical examples would be

- non-sufficient accuracy of some computation, when it is suspected that more iterations may improve the quality of the result,
- trying another boundary condition (e.g., Dirichlet instead of Periodic or other way around),
- continuing an iteration process,
- or using similar parameter values for the computation.

To restart and initialize the solver from a result file (*.gdr), check **Restart from .gdr File** and enter the name of a file, or **Browse** for it in the project folder.

When using **Restart From .gdr File** only the volume field (*.vap file) associated to the loaded *.gdr file will be used to continue the simulation.

Important! The *.vap file is saved in the result folder of a previous run. Make sure that this file was not deleted from the results folder. If the *.vap file is not available the restart option cannot be used.

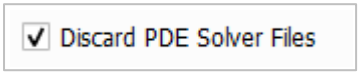
No other parameters from the options dialog will be loaded from the *.gdr file, but it is possible to load them by clicking **Load Parameters** in the Result Viewer. This way the

simulation can be continued while using the same settings as in the initial simulation.

Important! The structure from the run of which you want to use the result file must be loaded into GeoDict. If this is not the case, an error message is displayed.

Discard PDE Solver Files

Checking **Discard PDE Solver Files** causes the deletion of all intermediate computation files, such as log files and flow field files.



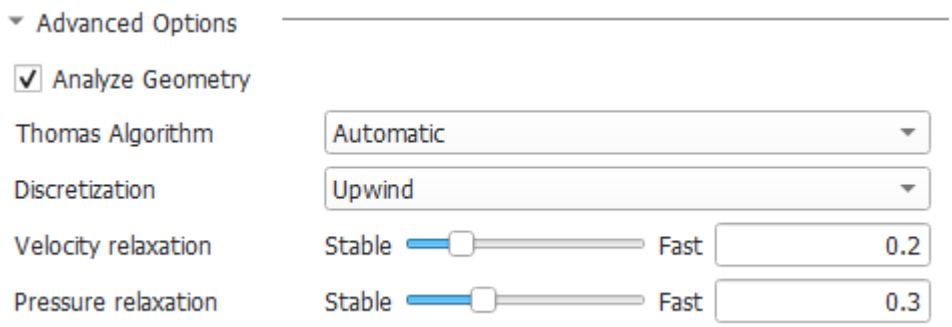
Only the content of the final GeoDict result file (*.gdr) is stored.

While having the benefit of saving hard disk storage place, discarding intermediate solver files has also the effect of disabling the 3D visualization of the results.

Advanced Options

Unfold **Advanced Options** to gain access to several advanced solver options.

Advanced Options for SimpleFFT



Analyze Geometry

Checking **Analyze Geometry** performs an analysis of the geometry before the solver computations.

If no percolation path through the structure is found, the partial differential equation does not need to be solved, and a permeability of 0.0 is directly reported as the solution.

Furthermore, the solver operates on the whole structure, regardless of which parts are

connected or unconnected. However, unconnected components are not transport relevant, thus the effort of solving in these parts is not necessary.

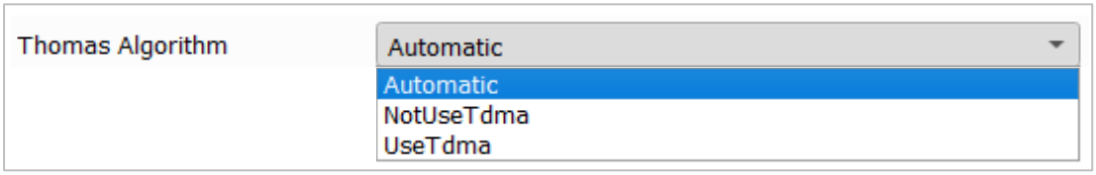
For these reasons, a geometrical analysis is routinely run to determine whether a through path exists. After the analysis is finished, unconnected components are removed from the computational grid.

This may speed up the computations but requires some time for the geometrical analysis, especially for very large structures. So if you know beforehand that there are no unrelevant parts or that the structure has been processed to eliminate them, the geometry analysis can be switched off by unchecking **Analyze Geometry**.

Thomas Algorithm

The tridiagonal matrix algorithm (Tdma), also known as the **Thomas Algorithm**, may help to improve the convergence for high porosity structure, especially for Navier-Stokes. The algorithm transforms the 3D flow problem to 2D. Thus, it leads to a shorter runtime and increases the convergence speed.

Three options can be chosen from the pull-down menu: **Automatic**, **NotUse Tdma** or **Use Tdma**.



If **Automatic** is selected, the SimpleFFT solver decides if **Tdma** will be used, regarding the porosity of the structure. Thus, **Tdma** is used if the porosity is high. For most cases it is recommended to choose **Automatic**.

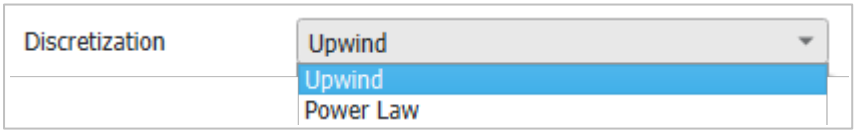
If a message dialog appears, displaying that NaN is detected in iterations, **NotUse Tdma** should be tried.

If the structures porosity is very high, it might decrease the runtime a little bit to explicitly choose **Use Tdma**. But for most cases **Automatic** will work well.

For more information about the Thomas algorithm see here.

Discretization

Choose the **Discretization** scheme of the inertia term when solving the Navier-Stokes (-Brinkman) equation.



The two available options are **Upwind** and **Power Law**, but we recommend to stay

with the default Upwind.

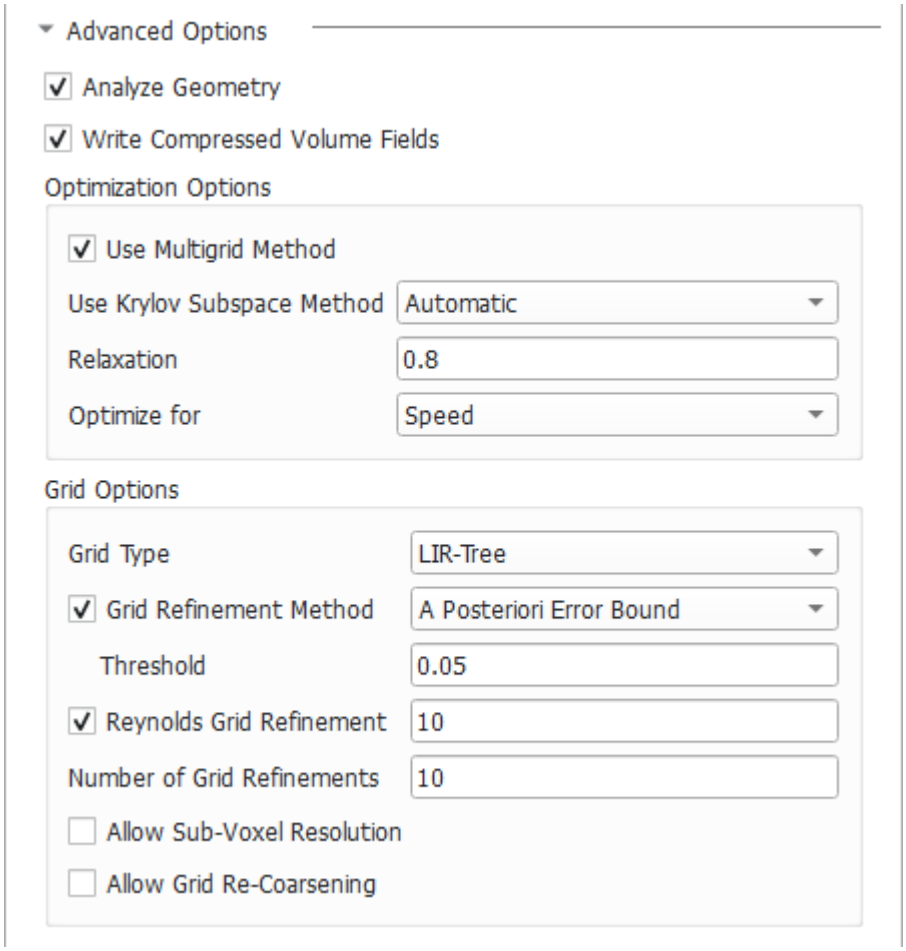
Relaxation

Depending on the material parameters and geometrical structure of the structure, the underlying mathematical problem can vary in complexity, thus influencing the behavior of the solver. This is directly related to the Reynolds number, an indication of the complexity of the flow solver computations. The higher the Reynolds number, the more **Stable** the flow solver settings should be, resulting in higher number of iterations, slower time stepping, and longer flow solver run times. However, making the solver run less iterations and, thus, faster (**Fast**), implies the risk that the solver does not converge.

For the SimpleFFT solver, the management of this balance is done through the **Velocity relaxation: Stable ↔ Fast** and **Pressure relaxation: Stable ↔ Fast** slide bars.

Setting the balance of **Stable** versus **Fast** is a trial-and-error process. Although there is no general rule to optimize it, the log files and the visualization of the structure might help finding the balance. Non-sense structure visualization results indicate that a more stable, slower solver is needed.

Advanced Options for LIR 12:



Analyze Geometry

Checking **Analyze Geometry** performs an analysis of the geometry before the solver computations.

If no percolation path through the structure is found, the partial differential equation does not need to be solved, and a permeability of 0.0 is directly reported as the solution.

Furthermore, the solver operates on the whole structure, regardless of which parts are connected or unconnected. However, unconnected components are not transport relevant, thus the effort of solving in these parts is not necessary.

For these reasons, a geometrical analysis is routinely run to determine whether a through path exists. After the analysis is finished, unconnected components are removed from the computational grid.

This may speed up the computations but requires some time for the geometrical analysis, especially for very large structures. So if you know beforehand that there are

no unrelevant parts or that the structure has been processed to eliminate them, the geometry analysis can be switched off by unchecking **Analyze Geometry**.

Write Compressed Volume Fields

The LIR solver uses a very memory efficient adaptive grid structure for the simulations.

If the option **Write Compressed Volume Fields** is checked, then the adaptive grid is used as compression method for writing out *.vap files. This option allows to save 80-90% space on hard drive. The runtime for writing *.vap files is also reduced significantly. But the runtime for loading and uncompressing of compressed *.vap is increased by the amount of runtime that was saved for writing out compressed *.vap files.

If the option **Write Compressed Volume Fields** is not checked, then a usual regular grid is used for writing out *.vap files.

Optimization Options

The **Multigrid Method** (see, e.g., Wesseling, 2004) was introduced to speed-up the computation and reduce the runtime significantly. The main idea of **Multigrid** is the usage of multiple coarser adaptive grids to speed up convergence behavior but requires only little more memory.

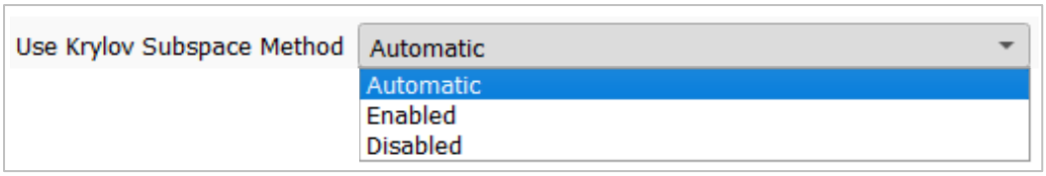
The method is available to solve the Stokes and Stokes–Brinkman equations as well as for solving mechanics, diffusion, thermal, and electrical conduction and is enabled by default.

Another speed-up option to accelerate the convergence behavior of the LIR solver is called **Krylov Subspace Method**.

The runtime of the LIR solver depends on many different properties of the structure and the simulation parameters. The BiCGStab algorithm is used, which can reduce the runtime for challenging simulation very drastically.

Note! Using the BiCGStab method approximately doubles the amount of RAM needed for the computation.

Unfortunately, the Krylov method is not always faster than a simulation without the Krylov method and therefore we introduced an **Automatic** mode which uses some heuristics to choose the most efficient method based on structure, material parameters, and boundary conditions automatically. Of course, it is possible to explicitly enable (**Enabled**) or disable (**Disabled**) the method.

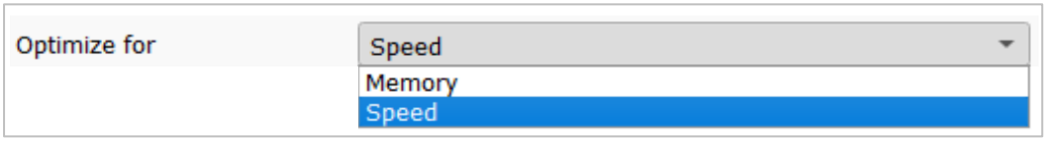


Note! In case that the Krylov subspace method (BICGStab) is used, the **Relaxation** may also be adapted automatically.

Depending on the material parameters and geometry of the structure, the underlying mathematical problem can vary in complexity, thus influencing the behavior of the solver. The more complex the problem is, the more stable the solver settings should be.

With the **Relaxation** number, the solver is adjusted from **Stable** (which results in higher number of iterations, slower time stepping, and longer solver run times), to **Fast**, which makes the solver run less iterations but implies the risk that the solver does not converge. The **Relaxation** is a parameter of the SOR method and must be between 0 and 2 to ensure convergence. For relaxation values smaller than one (<1.0), the simulation is more stable. For relaxation values larger than one (>1.0), the simulation converges faster.

The **LIR** solver can **Optimize for Speed** or **Memory**.



- If **Speed** is chosen, the solver constructs additional optimization structures. The runtime is decreased by up to 30% but requires up to 50% more memory compared to the other option.
- If **Memory** is chosen, the runtime is increased by up to 40% but the solver requires up to 50% less memory.

Grid Options

The **Grid Type** decides what kind of tree structure is used for the simulation.

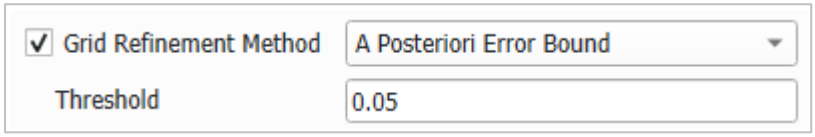
The default option is **LIR-Tree** and should always be used. The solver uses an adaptive grid structure called LIR-tree and needs up to 10 times less runtime and memory compared to the **Regular Grid** option.

The solver can analyze the result field during the computation and improves the adaptive grid in places where more accuracy is needed. The LIR solver splits cells where a high gradient occurs.

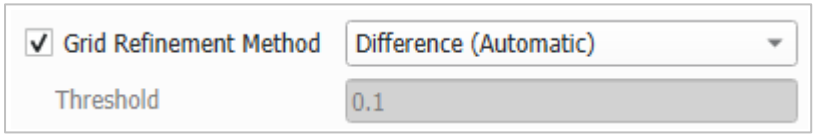
The solver can analyze the computed field and refines the adaptive grid during the computation at locations where more accuracy is needed. The LIR solver splits cells where a high gradients occur when the **Grid Refinement Method** is enabled. In this case, the additional parameters **Number of Grid Refinements**, **Allow Sub-Voxel Resolution**, and **Allow Grid Re-Coarsening** become available.

Select one of the three available refinement methods from the drop-down menu.

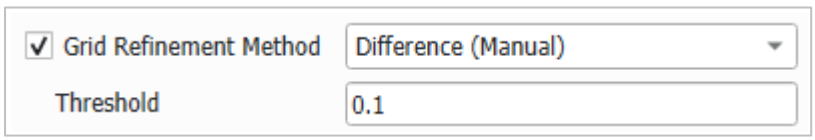
When **A Posteriori Error Bound** is selected, the solver targets the specified accuracy **Threshold**. While the **Error Bound** (set as Simulation Stopping Criterion) determines the relative error in the solution of the linear system, **A Posteriori Error Bound** refers to the relative error estimated by comparing high-order and low-order discrete solutions. The accuracy **Threshold** refers to the relative error to the analytical solution and the value must be between 0.0 and 1.0.



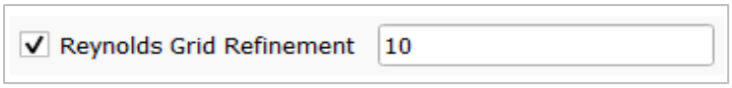
Choosing **Difference (Automatic)** leads to computational cells being split when the difference in values between neighboring cells exceeds a certain **Threshold**. The solver automatically chooses all internal parameters based on the structure and simulation settings. For this option, cells are split where the current gradient is greater than the **Threshold** multiplied by the maximum gradient, where the threshold is determined automatically.



Selecting **Difference (Manual)** also causes the computational cells to be split when the difference in values between neighboring cells exceeds the specified **Threshold**. You can specify all parameters (Threshold and Number of Grid Refinements) for the grid refinement manually. For this option, cells are split where the current gradient is greater than the **Threshold** multiplied by the maximum gradient. The **Threshold** value must be between 0.0 and 1.0. The recommended value range is between 0.05 and 0.1.



Activate **Reynolds Grid Refinement** to use the local Reynolds number (Re) for adaptive grid refinement.



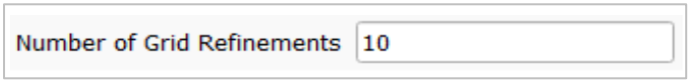
A grid cell is split if the local Reynolds number, i.e., the ratio between inertial and viscous forces exceeds the given threshold. The Reynolds number is defined as:

$$\mathrm{Re} = \frac{\rho \mathrm{u} \mathrm{L}}{\mu} \quad (38)$$

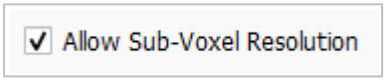
Reynolds Number

where ρ is the density of the fluid (SI units: kg/m^3), u is the velocity of the fluid (m/s), μ is the dynamic viscosity of the fluid (Pa·s or N·s/m^2 or kg/m·s), L is a characteristic length (m). For the local Reynolds number, the size of the grid cell is used for the characteristic length.

The **Number of Grid Refinements** controls the maximum number of grid refinements that the solver can perform during the simulation. The value should be set between 0 and 10, where a value of 0 means that no grid refinements will be made. Grid refinements may increase the number of iterations, runtime, and memory requirements.



When **Allow Sub-Voxel Resolution** is enabled, the solver is allowed to split computational cells to sizes smaller than the voxel length.



This feature is beneficial for low-porosity structures for which the pore throats require finer resolution or for modeling fast Navier-Stokes[7] flows with strong vortices.

Enabling this feature may increase the number of iterations, runtime, and memory requirements.

Check **Allow Grid Re-Coarsening** to allow the solver to automatically revert the grid refinement and coarsen the computational cells.

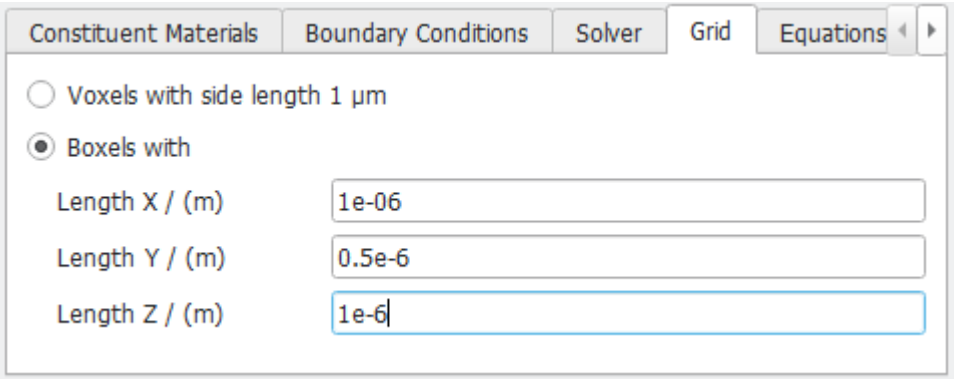
This means, grid refinement is done temporarily. Afterwards, the cells are merged as soon as accuracy is not needed anymore. This reduces the memory and runtime requirements.

Grid

For the the SimpleFFT[12] solver, the computational grid can be adjusted under the **Grid** tab if necessary.

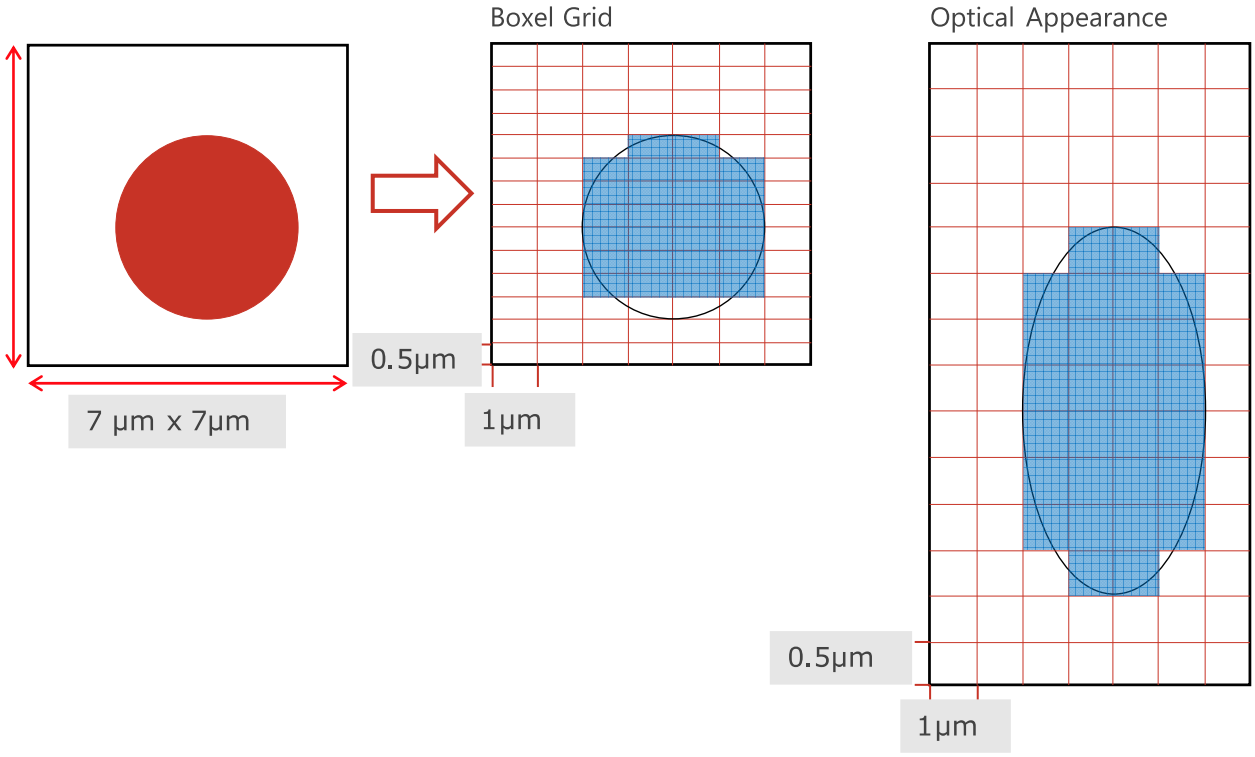
If **Boxels** (cuboid) instead of **Voxels** (cube) are used to resolve the structure, the boxel lengths

can be set in the **Grid** tab.



Sometimes, grid lengths may be different in each direction, e.g., for Focused Ion Beam (FIB) images, the grid size in Z-direction is often different from the size in X- and Y-directions.

For example, a structure is sampled in a computational grid with different lengths in X- and Y-direction (e.g., LX = 2*LY). When the structure is loaded in GeoDict, each grid cell gets assigned 1 voxel and the visual representation of the structure looks stretched. To correct this effect, boxels set as, e.g., LX = 1 µm and LY = 0.5 µm, need to be chosen in the **Grid Size** panel.



1.3.2 Results

After the solver has finished, the result file (.gdr) is saved in the project folder and the **Result Viewer** automatically opens. Find detailed information on the **Result Viewer** options in the Result Viewer user guide.

Results of the Navier-Stokes-Brinkman command

- Report 85
  Shows the computed results.
- Plots 87
  View result plots.
- Flow Visualization 90
  Load results for a graphical visualization.
- Create Videos 91
  Use a predefined macro to create a video.

Report

The **Report** tab under the **Results** tab shows the computed results. For each selected Computational Direction 62, the computed results are shown:

- Pressure drop / flow velocity: the determined result (pressure drop or flow velocity) is always shown in bold face.
- Volume flow rate: either prescribed as boundary condition or computed in the simulation.
- Reynolds number: estimated with the given characteristic length. You can change the char. length in the post-processing options.
- Asymptotic estimate of pressure: the asymptotic result predicted by the Error Bound 72 stopping criterion

Z-direction

Pressure drop is **356.721 Pa** at average (superficial) flow velocity: 1 m/s.
Volume flow rate: 1000 l/(s m²)
Reynolds number 2.10179 for characteristic length 3.20156e-05 m.
Asymptotic estimate of pressure: 359.769 Pa.

Warning Messages

The following warnings may appear in the result report:

> **Warning:** The tangential component of the velocity is more than 5% of the normal component at the outlet. The outflow may not be developed. Please consider adding more outlet length.

Given a sufficiently long outlet, it is expected that the applied Velocity In - Pressure Out boundary conditions lead to outlet velocities that point strictly outwards in the flow direction. If this warning appears, this is not the case. If possible, repeat the computation with a longer outlet.

> **Warning:** The pressure loss in the implicitly added inlet / outlet is more than 1%. It is recommended to add an inlet/outlet with ProcessGeo explicitly to see the pressure loss in the visualization and in the pressure curve.

In this case, a significant pressure loss is observed in the outflow region, caused by stationary vortices. As the outflow region was only added implicitly, these vortices and the pressure loss are not visible in the result plots and flow field visualization. Consider

> to add the inflow and outflow region explicitly using ProcessGeo Embed and then run the computation again.

Information about the convergence of the iterative solver is also displayed for each selected computation direction: Iterations, runtime, memory usage, and applied stopping criterion.

Flow solver information, Z-direction

Solver	LIR
Threads	12
Iterations	2650
Number of cells	818803
Runtime	1.96335 min
Memory usage	511.294 MiB
Stopping criterion	error bound

Total runtime: 00:02:01, Total memory usage: 1.372 GiB

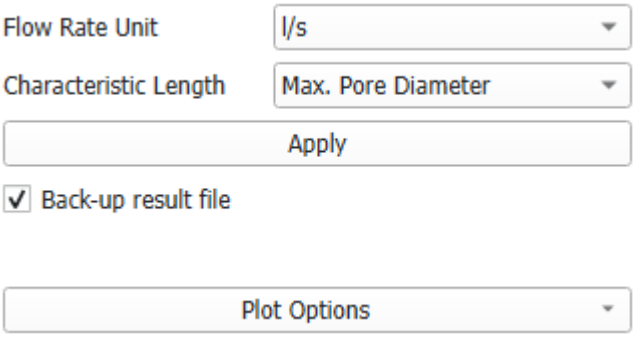
In the last line the total runtime and total memory usage are given. Here not only the resources used by the solver are taken into account but also writing volume fields and structure files or already loaded volume fields from previous simulations. Thus, even if only one flow direction is computed, the total runtime and total memory values are usually higher than the solver specific values.

Post-Processing

In the post-processing panel on the left of the report, you can change the **Flow Rate Unit** to **m³/s**, **l/s**, **l/min**, **m³/h**, or **cfm**.

You can also change how the **Characteristic Length** used in the calculation of the Reynolds number is determined. Select between **Numerical Length**, **Max. Pore Diameter**, **Max. Fiber/Particle Diameter**, and **Manually specified**. If **Manually specified** is selected, specify the characteristic length below. You can find further information on the **Characteristic Length** here[19].

Click **Apply** to change the **Report**.



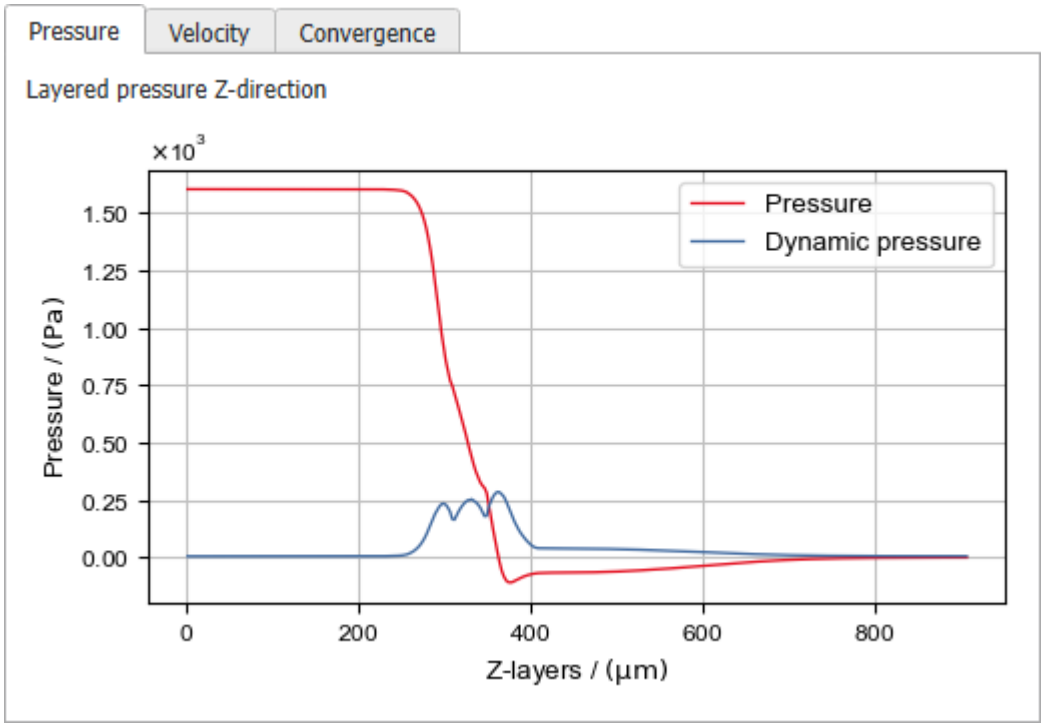
Important! Note that after clicking **Apply**, the result file changes and is not the original simulation result file anymore! If **Back-up result file** is checked, the original *.gdr file is saved in the result folder and can be restored at any time.

Plots

Under the **Plots** tab various result plots are available:

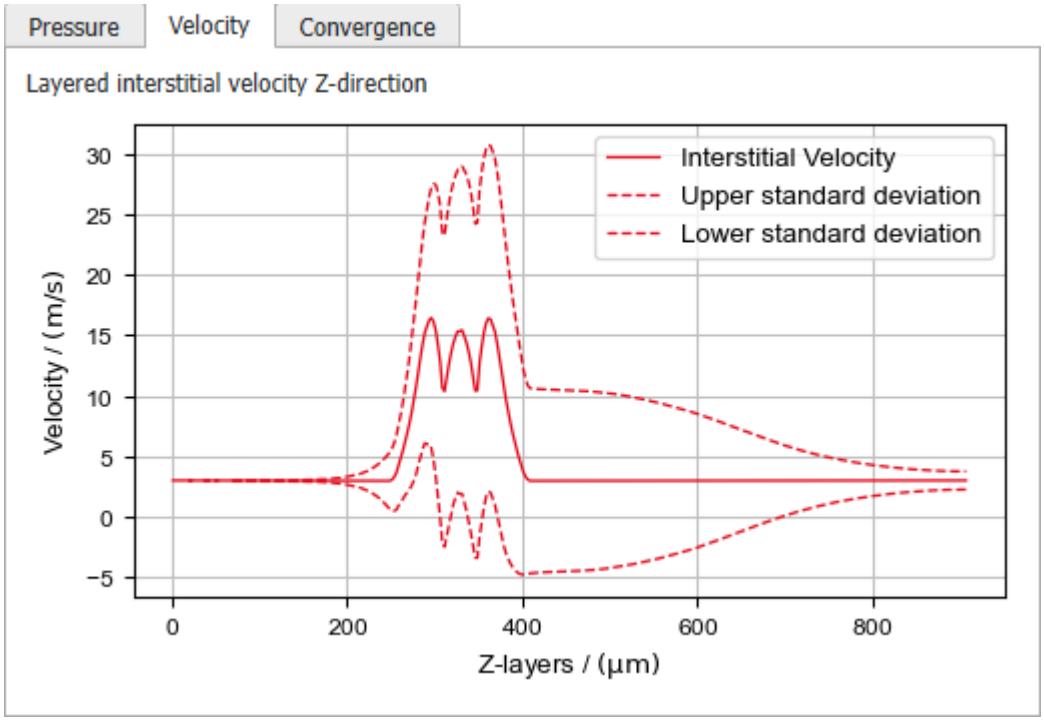
Pressure

Under the **Pressure** tab, a graph of the changes in **Pressure** across the layers is shown. There is one plot for each selected Computational Direction. Here, only the Z-direction had been chosen for computations.



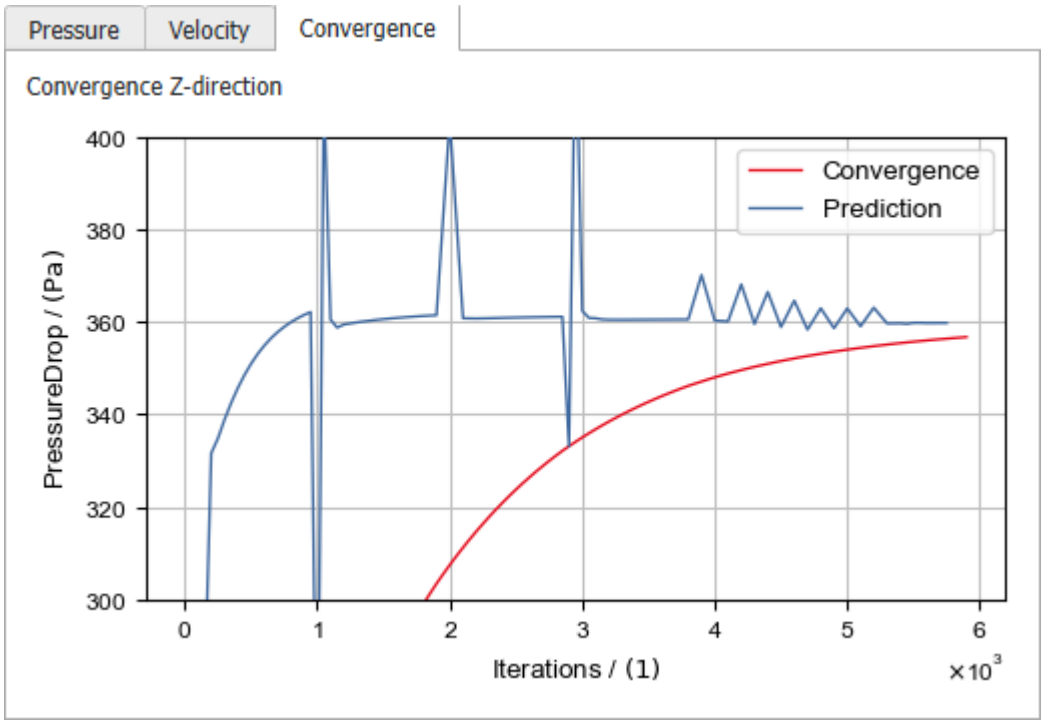
Velocity

Under the **Velocity** tab, a graph of the changes in **Velocity** across the layers is shown. The velocity component in the chosen direction is averaged over the pore voxels for each layer in this direction. Therefore, the velocity matches the average velocity in the empty inflow and outflow region.



Convergence

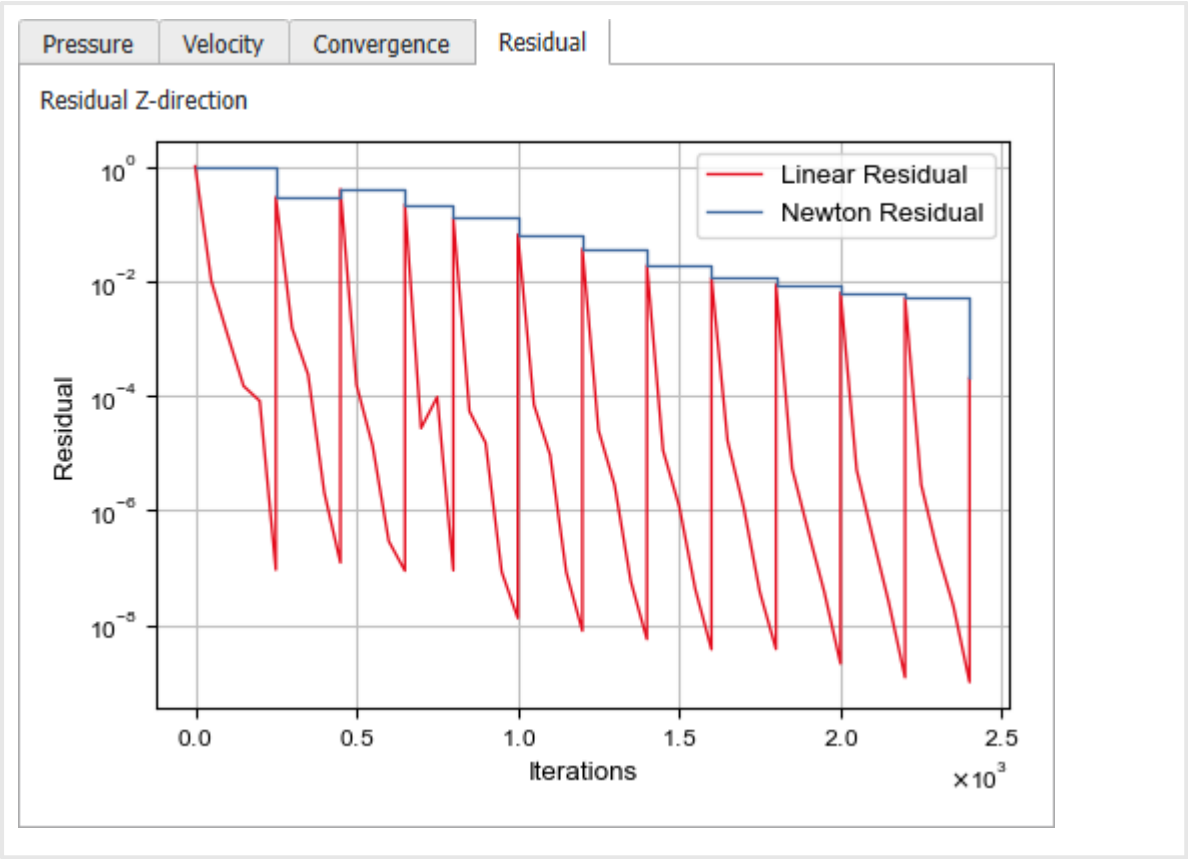
The **Convergence** tab plots the computed mean velocity, flow rate or pressure drop at each iteration. For the **Error Bound** criterion, it also shows the predicted value. In this case, the solver stops when the computed solution and the prediction are within the selected relative error.



Residual

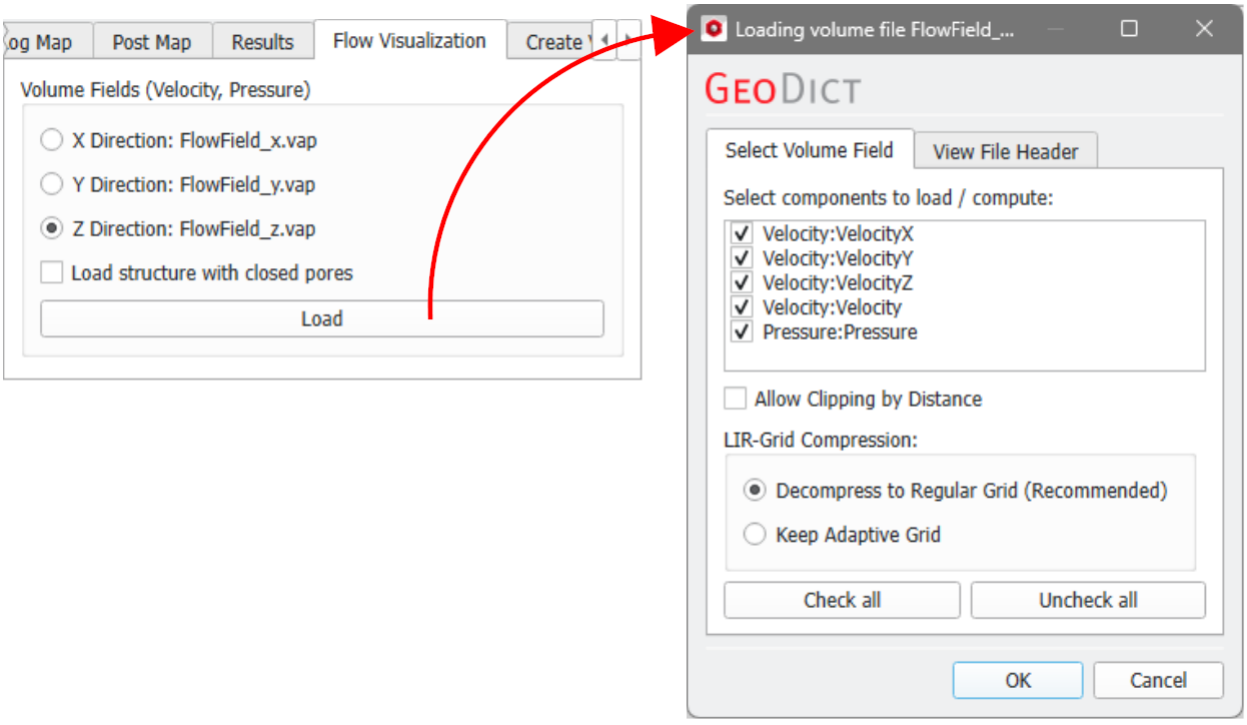
The **Residual** plots tab shows the **Linear Residual** and the **Newton Residual** at given **Iteration** values. The **Newton Residual** is defined as the residual while solving non-linear equations with Newton's method. The Newton residual curve changes, when the equation is solved, while the linear residual changes with every iteration while solving a linear equation system.

This plot is available only when the Krylov method is used.



Flow Visualization

Select the **Flow Visualization** tab to load the computed flow field (velocity and pressure, *.vap). Dependent on the selected **Computational Direction(s)**, result fields can be available for flow in **X-**, **Y-**, and/or **Z-Direction**.



Select the flow direction, then click **Load**. A dialog pops up, that let's you select which of the stored field components **VelocityX**, **VelocityY**, **VelocityZ**, **Velocity**, and/or **Pressure** you want to load and visualize.

Check **Allow Clipping by Distance** if you want to use this feature in the visualization. If selected, an additional data field is computed and loaded that contains the Euclidean distance to the solid structure. You can use this data field, e.g., to print arrows only with a certain distance from the solids. If you do not need this feature, do not select it, as it requires additional RAM to store the distance field.

Result files computed with the LIR solver are stored using the adaptive grid of the LIR solver (**LIR-Grid Compression**). Select if you want to interpolate the data during import to the regular voxel grid or if you want to keep the adaptive grid:

- **Decompress to Regular Grid:** More RAM is needed to load the data, and you cannot visualize the adaptive grid. This option enables the visualization with Streamlines, Arrows, and Schlieren.
- **Keep Adaptive Grid**: Less RAM is needed to load the data. You can visualize the adaptive grid. You cannot use Streamlines, Arrows, or Schlieren to visualize the results.

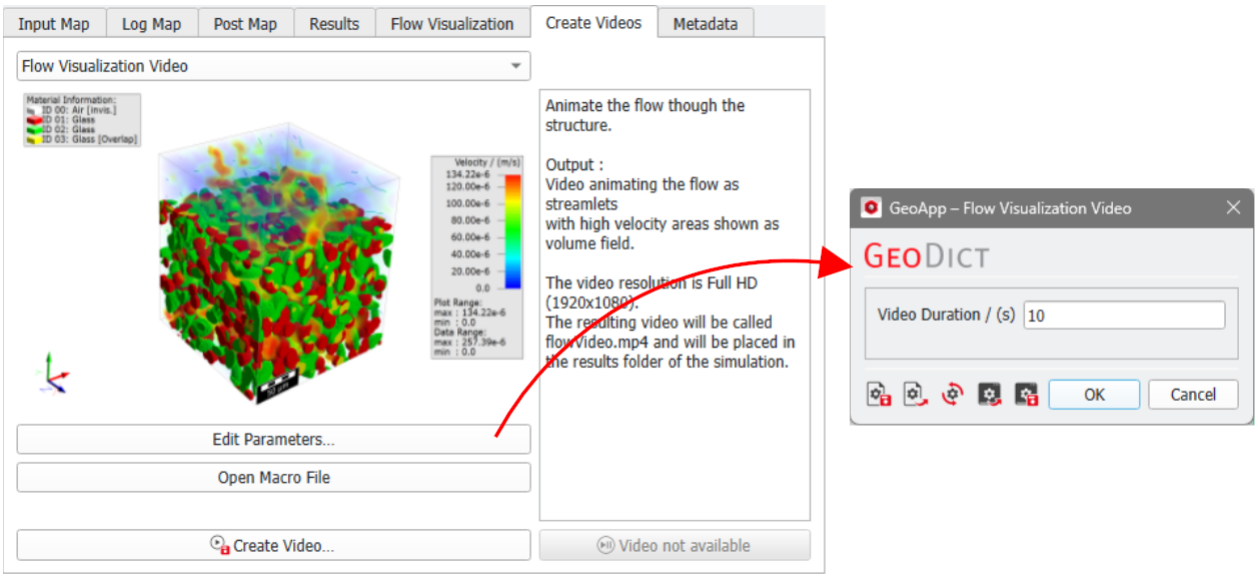
The structure that the solver used internally is loaded if **Load structure with closed pores** is checked. This structure shows which pores have been closed due to prior geometry analysis.

Visualization Options

- Volume Field: Pressure drop or velocity are visualized by a 3D color field - you can clip by threshold values or use transparency to view inside.
- Streamlines: Visualize the flow through the structure with streamlines or animated streamlets.
- Arrows: Visualize the velocity vector classically with arrows.
- Schlieren: This option uses Line Integral Convolution (LIC) fields to visualize vortices.

Create Videos

The **Create Videos** tab provides a predefined Python macro to create a video of the structure with animated flow streamlets with high velocity areas shown as volume field. Click **Edit Parameters** to change the **Video Duration** in the **Flow Visualization Video Parameters** dialog.

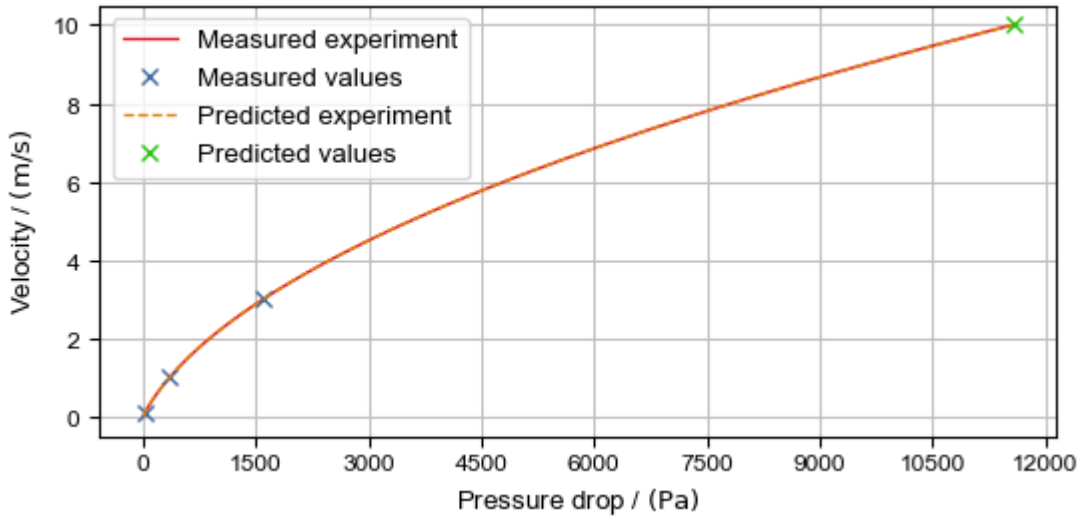


Click **Create Video** to start the video generation. This generates a video (*.mp4), GeoDict result file (*.gdr), and a results folder inside the corresponding result folder of the FlowDict simulation.

1.4 Forchheimer Approximation

The **Forchheimer Approximation** command allows to estimate pressure drop and mean velocities for high Reynolds numbers from measured or simulated results for small Reynolds numbers.

At high Reynolds numbers, the flow becomes turbulent and no stationary solution exists. In this situation, FlowDict cannot compute a stationary solution and the Navier-Stokes[58] solver will diverge or oscillate. Nevertheless, flow simulation results achieved for smaller Reynolds numbers, i.e., at lower flow velocities, can be used to predict the outcome at higher velocities. It is also possible to change the fluid properties, i.e., viscosity and density, and estimate pressure drop or mean velocities for other fluids with lower viscisities. In short, the formula allows to extrapolate the results of higher Reynolds numbers.



Forchheimer's equation describes the pressure drop of a fluid flow (in a porous media). The equation extends Darcy's law[20] and considers the pressure drop caused by turbulence in addition to the pressure drop caused by dynamic viscosity:

$$-\nabla p = \frac{\mu}{K}\vec{u} + \frac{\rho}{k_1}\vec{u}^2 \qquad (39)$$

Forchheimer's equation

where $\vec{u}$ (m/s) is the fluid flow velocity, K (m^2) is the permeability, μ (Pa · s) is the dynamic viscosity, p (Pa) is the pressure, and ρ (kg/m^3) is the density. The new symbol k_1 (m) is the non-Darcy permeability coefficient and depends on the porous media only. It does not depend on the flowing fluid.

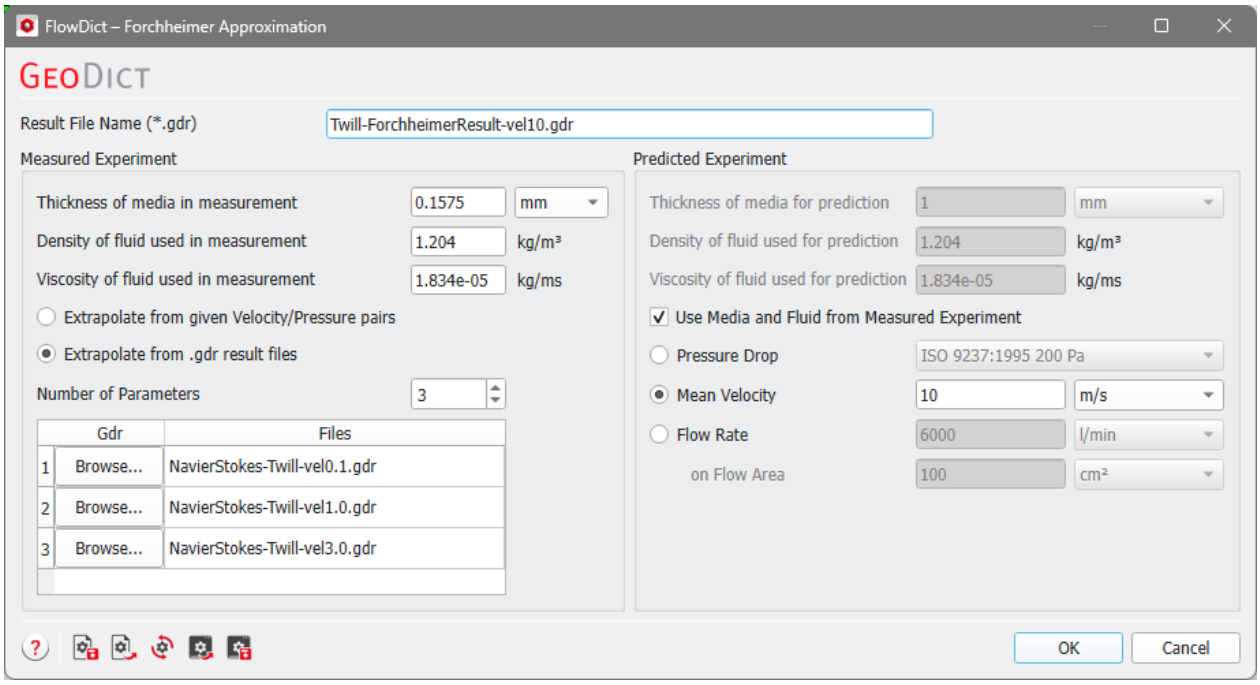
The Forchheimer Approximation command

- Options[94]
  How to set the input parameters of the command.
- Results[96]
  View the simulation results.

1.4.1 Options

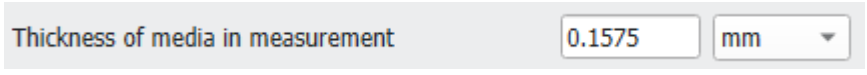
At the top of the dialog, enter the **Result File Name (*.gdr)**.

The **Forchheimer Approximation Options** dialog is organized in two panels. In the **Measured Experiment** panel, enter the measurement or simulation results and the corresponding fluid parameters. In the **Predicted Experiment** panel, enter the setup for which you want to predict pressure drop or mean velocity.

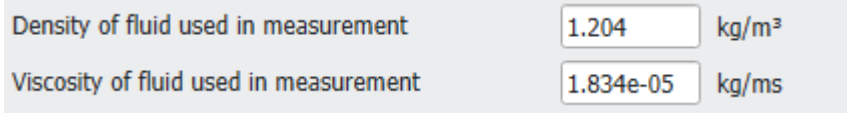


Measured Experiment

The **Thickness of media in measurement** must correspond to the velocity/pressure pairs or simulated results entered below. If you extrapolate from simulated results, enter the used domain size in flow direction without inflow or outflow region.



Below, enter the fluid density and viscosity. If you extrapolate from simulated results, enter the parameters used in the simulation.

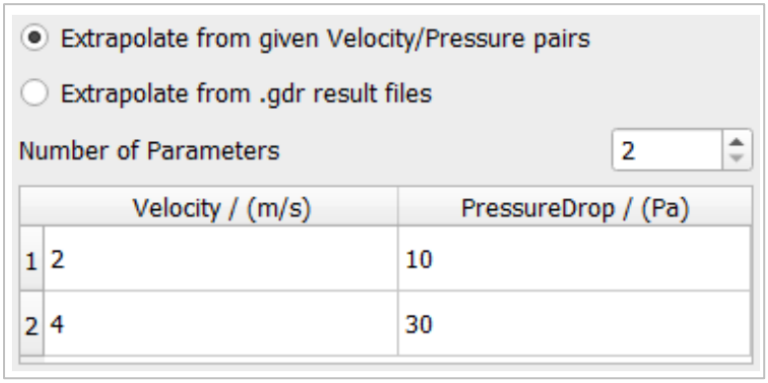


Extrapolate from given Velocity/Pressure pairs

Enter **Velocity / Pressure Drop** pairs from real measurements or simulated experiments. From these values the **Forchheimer Approximation** extrapolates the values for the desired **Predicted Experiment**. A quadratic polynomial is fit to the given

velocity / pressure drop pairs with quadratic regression by least squares analysis. At least two of these pairs are required, and it is necessary that the entered pairs show a nonlinear behavior (it is not possible to extrapolate the quadratic behavior if the given results are both from very small Reynolds numbers).

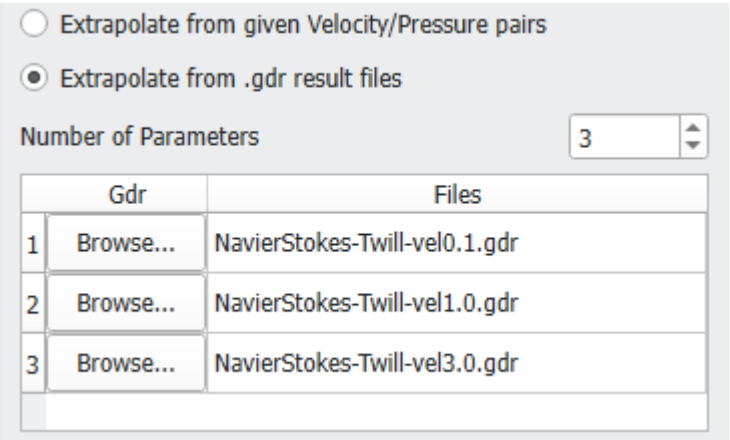
A plain table for velocity and pressure drop is displayed below. Adjust the **Number of Parameters** according to the number of given pairs and enter the velocity / pressure drop pairs into the table.



Extrapolate from .gdr result files

If **Extrapolate from .gdr result files** is chosen, a table for result files in .gdr format appears. Adjust the **Number of Parameters** according to the number of given .gdr files.

Browse to the given .gdr files until the table is filled. These .gdr files must be result files from previous FlowDict flow simulations. The velocity / pressure drop pairs are then extracted from the given files. Make sure that the results were achieved with the Navier-Stokes command and that not all results were achieved for very small Reynolds numbers.

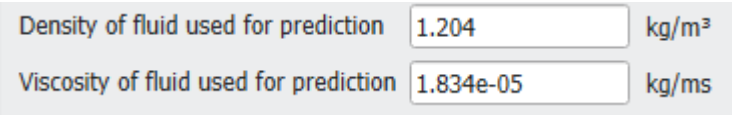


Predicted Experiment

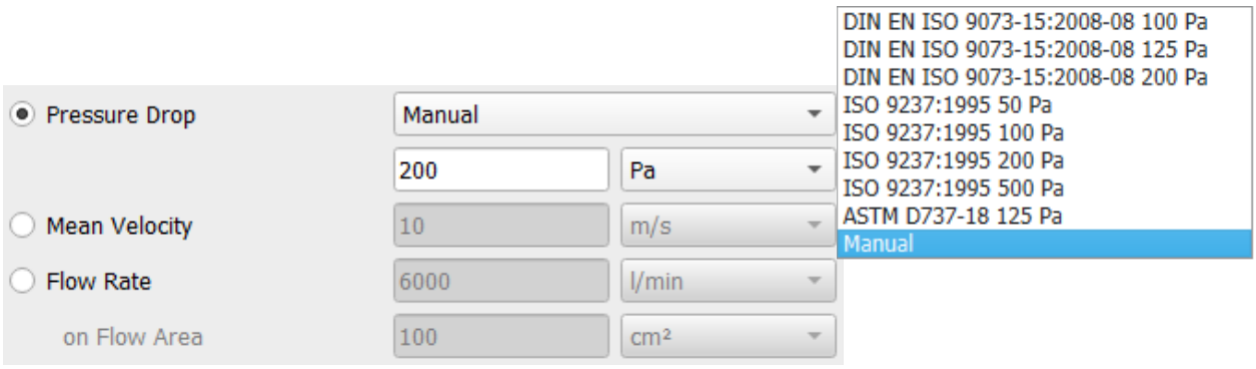
If you select **Use Media and Fluid from Measured Experiment**, the values entered in the Measured Experiment 94 tab will be used.

Otherwise, you can enter a different media thickness. Be aware that this only makes sense if it is a thicker or thinner layer of the same porous material. The extrapolation assumes that the material in the predicted experiment has the same permeability (Darcy and Forchheimer) as in the measured experiment.

You can predict experimental results for a different fluid as used in the measurement by entering the density and viscosity here.



Enter either the pressure drop, the mean velocity, or the flow rate of the predicted experiment. If a **Pressure Drop** is prescribed, then the predicted property is the mean velocity. If a **Mean Velocity** or a **Flow Rate** is prescribed, then the pressure drop is predicted.



In addition to the manual definition of the **Pressure Drop**, there are some presets based on **DIN**, **ASTM**, and **ISO** norms.

1.4.2 Results

After clicking **Run**, the Result Viewer opens directly. No partial differential equation is solved, and thus no computational resources are needed for the **Forchheimer Approximation**.

Report

The **Results - Report** tab shows the approximation of Forchheimer's equation that fits the entered velocity / pressure drop pairs. The report also shows the determined parameters for the **Darcy permeability** κ and the **Inertial permeability** k_1 as defined in (39) Forchheimer's equation 93.

Forchheimer equation for measured Experiment:

Measured velocity list in (m/s): (0.1, 1, 3).
Measured pressure list in (Pa) : (28.682, 356.721, 1604.04).

$\Delta P = 267.961\ u + 88.9047\ u^2$

For $-dp/dx = \mu u/\kappa + \rho u^2/\kappa_1$
where κ is Darcy permeability, κ_1 is inertial permeability, μ is dynamic viscosity, ρ is fluid density, and L is medium thickness.

Darcy permeability $\kappa = \mu * L / f1 = 1.834e\text{-}05 * 0.0001575 / 267.961\ m^2 = 1.07797e\text{-}11\ m^2$.
Inertial permeability $\kappa_1 = \rho * L / f2 = 1.204 * 0.0001575 / 88.9047\ m = 2.13296e\text{-}06\ m$.

Below, the approximation of Forchheimer's equation for the predicted experiment is shown. While the **Darcy permeability** κ and the **Inertial permeability** k_1 will always be identical to the measured experiment (we assume it is the same porous material), the formula given for ΔP will differ if a different media thickness or different fluid parameters were used for the prediction:

Forchheimer equation for predicted Experiment:
$\Delta P = 267.961\ u + 88.9047\ u^2$

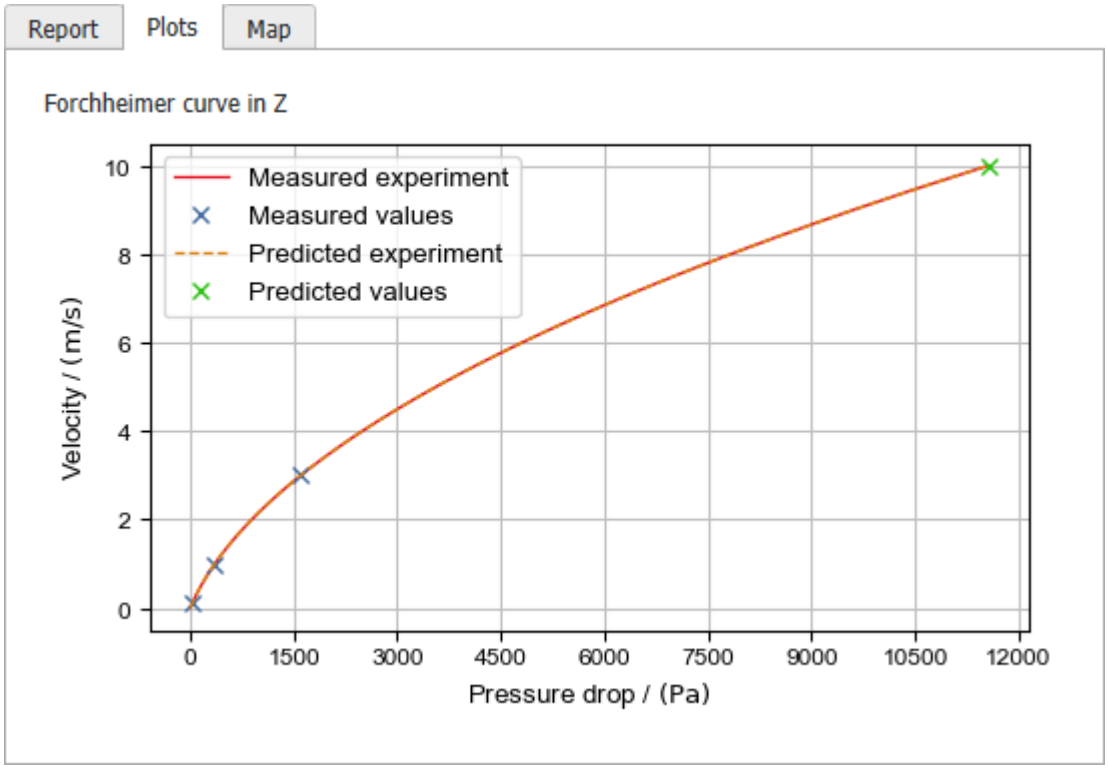
For $-dp/dx = \mu u/\kappa + \rho u^2/\kappa_1$
where κ is Darcy permeability, κ_1 is inertial permeability, μ is dynamic viscosity, ρ is fluid density, and L is medium thickness.

Darcy permeability $\kappa = \mu * L / f1 = 1.834e\text{-}05 * 0.0001575 / 267.961\ m^2 = 1.07797e\text{-}11\ m^2$.
Inertial permeability $\kappa_1 = \rho * L / f2 = 1.204 * 0.0001575 / 88.9047\ m = 2.13296e\text{-}06\ m$.

The estimation is 11570.1 Pa for 10 m/s.

Plots

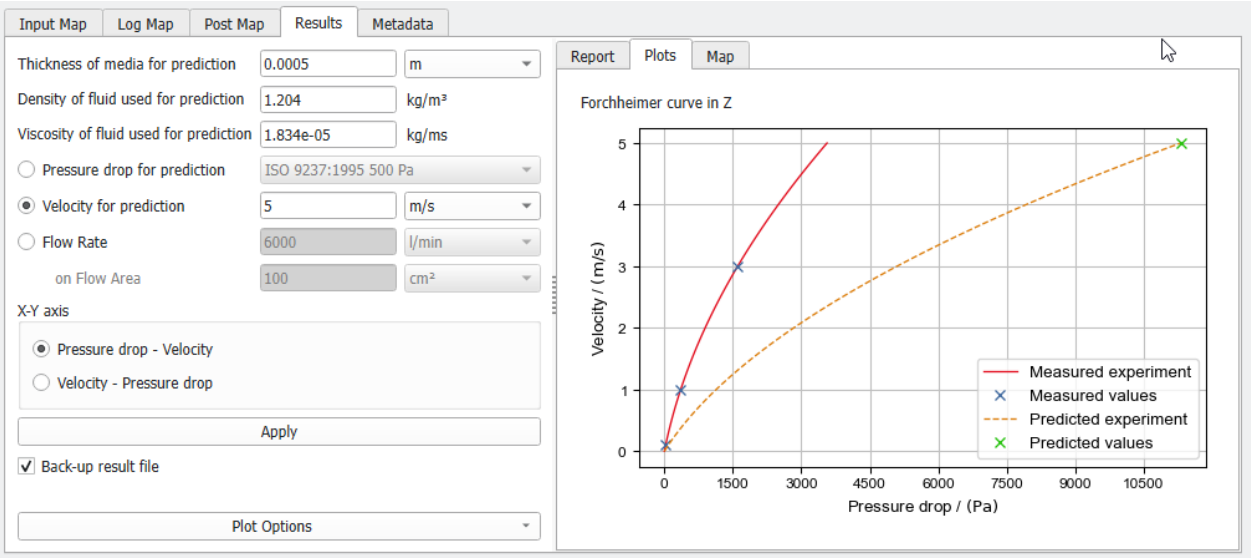
The **Results - Plots** subtab shows the computed quadratic polynomial in a graph. The velocity / pressure drop pairs entered in the **Forchheimer Approximation Options** dialog are highlighted on the curve with a blue **X**, while the green **X** shows the predicted velocity / pressure drop pair. The plot can be further customized by right-clicking into the plot area. See the Result Viewer user guide for details.



Post-Processing

In the post-processing area on the left, you can change all parameters entered in the Predicted Experiment 95 panel. Click **Apply** to change the plot and report of the result file. If you choose the media thickness or fluid density and viscosity different from the Measured Experiment, the **Predicted Experiment** curve will differ from the **Measured Experiment** curve.

Important! Note that after clicking **Apply**, the result file changes and is not the original simulation result file anymore! If **Back-up result file** is checked, the original *.gdr file is saved in the result folder and can be restored at any time.



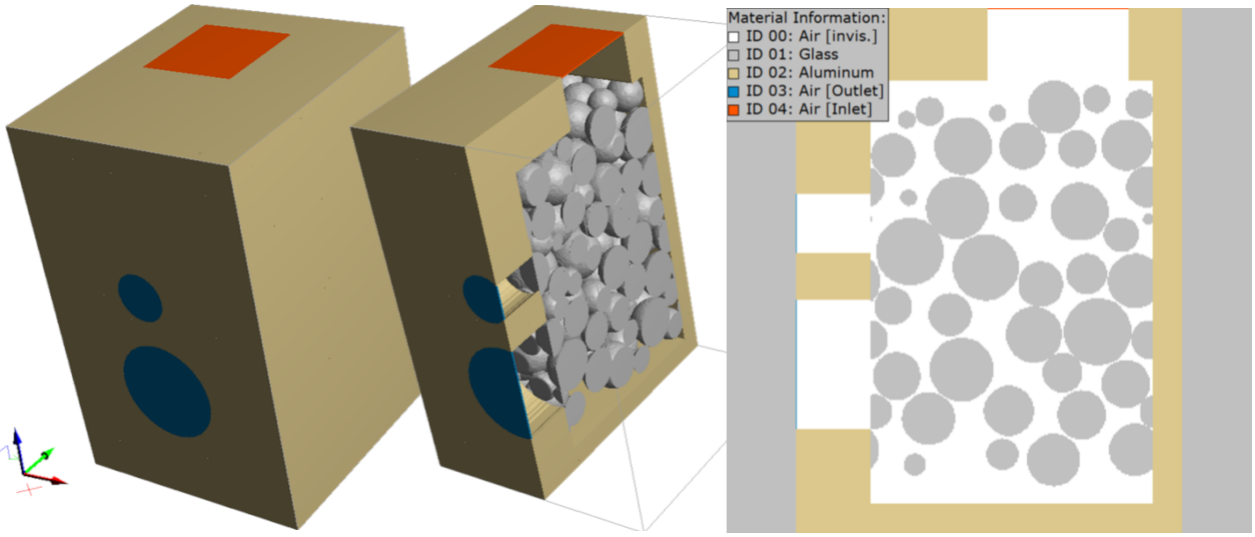
The graph can also be plotted with interchanged axes by checking **Velocity – Pressure drop** and clicking **Apply**.

1.5 Simulate Flow Experiment

Use **Simulate Flow Experiment** if the inlet and outlet are not axis-aligned in the same flow direction, for example if the flow enters and leaves a component through tubes. Define where the flow enters and leaves the structure by setting up one inlet and one or more outlets. Usually, the complete structure is encased by solid voxels except for the inlet and outlet voxels. For symmetric structures, it is also possible to model only one half or one quarter of the structure and use symmetric boundary conditions instead. The inlet and outlet are defined by certain material IDs.

The inlet and outlet materials must be defined in the structure with certain conditions:

- While they must have different material IDs, inlet and outlet should have the same material (i.e., fluid) as the pore space inside the structure.
- While several IDs can be defined as outlet, only one ID can be set to inlet.
- There can be several inlet and outlet areas. They can even be located at different sides of the structure.
- Since GeoDict 2024 inlet and outlet can be thicker than 1 voxel and do not need to be axis-aligned. Also, pore voxels at the boundary are allowed.
- Only one connected pore system is allowed.
  - It must contain all fluids and porous materials as well as inflow and outflow areas.
  - Inlet and outlet are not allowed to be connected via an outer fluid.

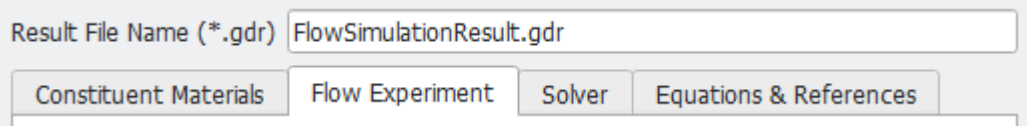


The Simulate Flow Experiment command

- Options 101
  How to set the input parameters of the command.
- Results 116
  View the simulation results.

1.5.1 Options

Select **Simulate Flow Experiment** from the pull-down menu and click the **Option**'s **Edit...** button to open a dialog to enter or edit the simulation options. Choose a **Result File Name** for the result file (*.gdr) fitting to your current project.



The parameters are organized under the tabs: **Constituent Materials**, **Flow Experiment**, and **Solver**. The **Equations & References** tab gives a summary of the used equations and symbols which are explained in more detail here 7.

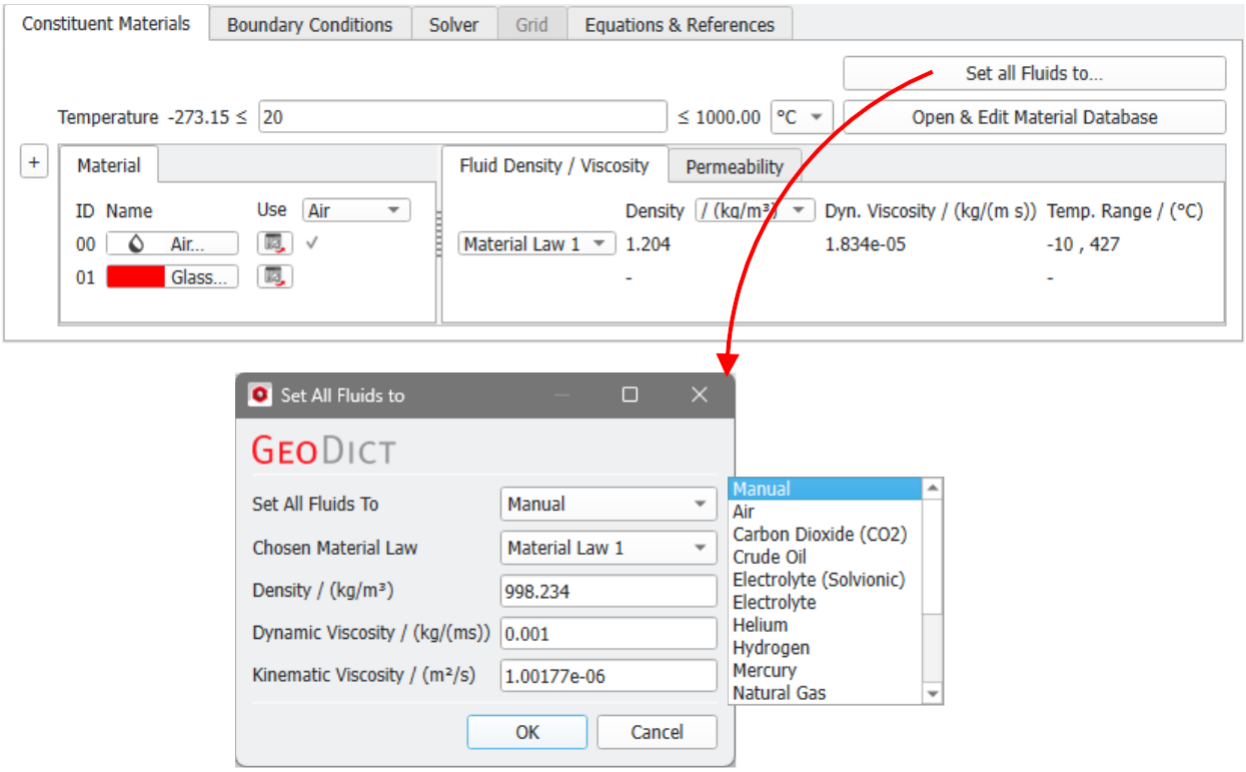
Tabs of the Simulate Flow Experiment dialog

- Constituent Materials 101
  Define the materials in the structure.
- Flow Experiment 104
  Define flow motion and boundary conditions.
- Solver 107
  Set up the flow solver.

Constituent Materials

For the flow solver's computation, the materials in the structure and the fluids flowing through it need to be defined under the **Constituent Materials** tab, so that their physical properties parameters are taken into account.

To set the fluid flowing through all pore and porous materials, click **Set all Fluids to** and select it from the pull-down menu.

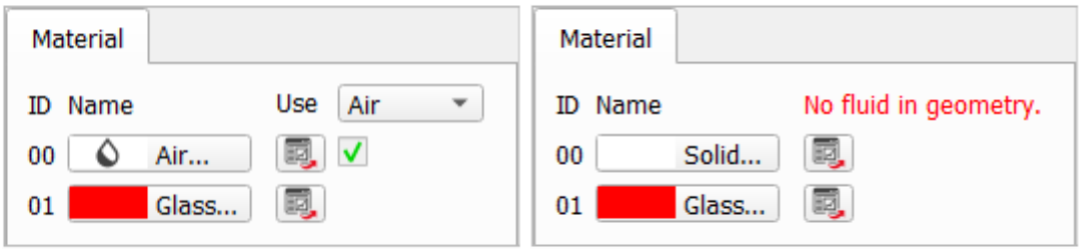


If one of the predefined fluids is used, the values for **Density**, **Dynamic Viscosity**, and **Kinematic Viscosity** are taken from the GeoDict Material Database. They are dependent on the given **Temperature** value and the **Chosen Material Law**. If **Manual** is selected, these values can be entered manually.

Any values entered manually, e.g., for a special type of oil, can also be added to the material database through the **Open & Edit Material Database** button. Additionally, different properties for the same material can be saved as different **Material Laws** and can then be selected from the pull-down menu in the **Set all Fluids to** dialog. **Material Law 1** is the default shipped with GeoDict. Click here for more information on editing, expanding, and using the GeoDict Material Database.

The **Temperature** is selectable in Kelvin (K), Celsius (°C), and Fahrenheit (F), and has a default value of 293.15 K (20.0°C, 68 F).

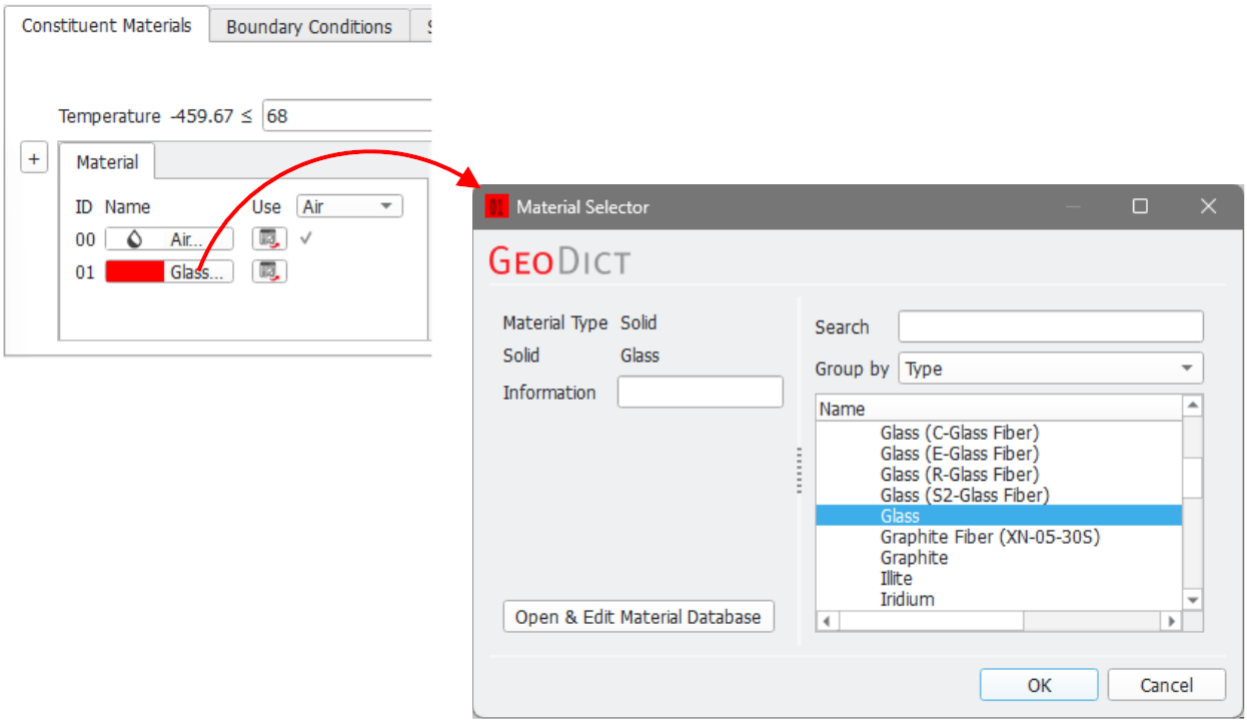
Under the **Material** subtab, select the fluid used in the simulation from the **Use** pull-down menu. If other fluids are contained in the structure, they will be treated as immobile (solid) during the simulation. If the structure contains no fluid material, **No fluid in geometry** is displayed in the top of the **Material** tab. Select a material for the pore phase by clicking on the corresponding material button.



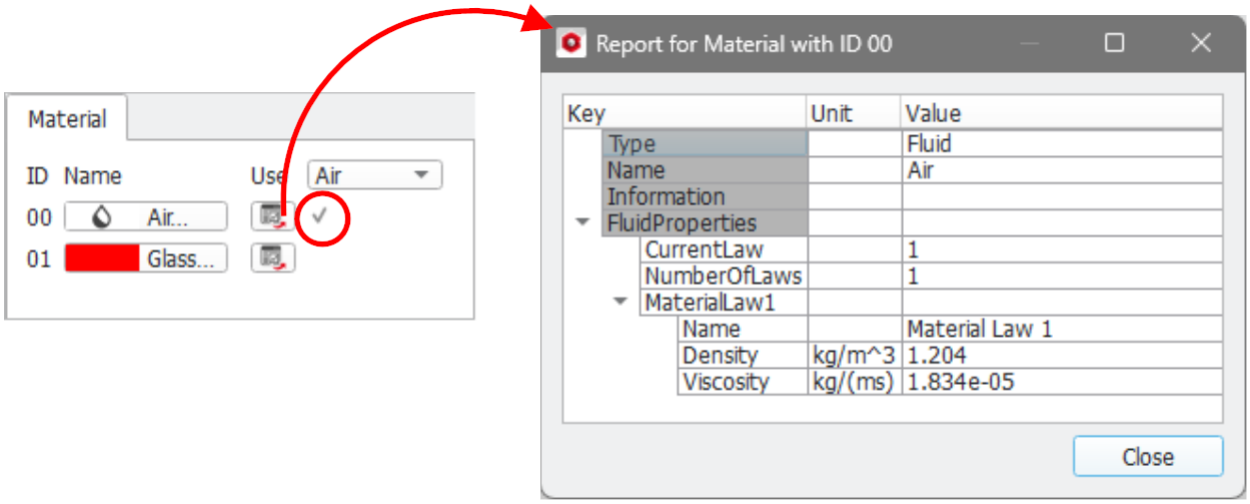
The density and the dynamic viscosity of this fluid, at the selected temperature, is shown under

the **Fluid Density/Viscosity** subtab.

The **Material** subtab shows also the material **IDs** assigned to the materials (00, 01, 02, etc.). For each material ID, you can open the **Material Selector** dialog and select a material by clicking on the corresponding material buttons.

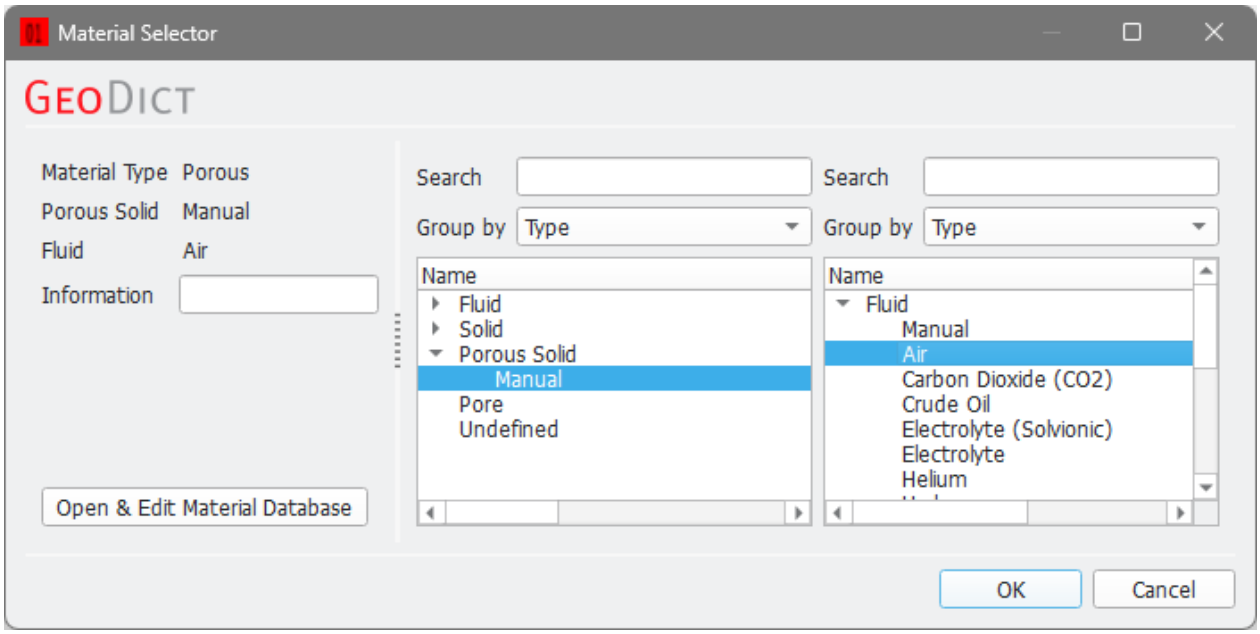


Click the **Material Report** button to see the relevant parameters for the material with a selected ID.

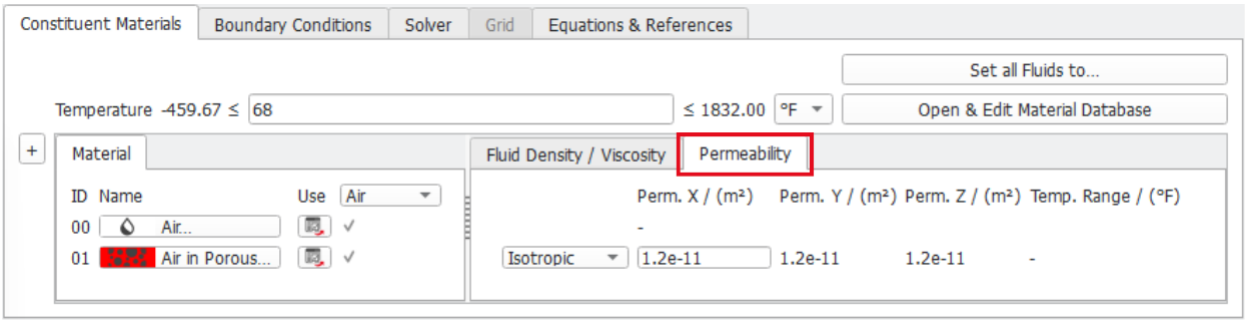


Fluid flow can happen in pores and porous materials, but not in solid materials. The check sign next to the **Material Report** icon marks all materials in which flow happens.

For porous voxels, set the **Material Type** (in the Material Selector dialog) to **Porous** and choose the fluid that occupies the pores.



The permeability of porous voxels is set under the **Permeability** subtab. You can define if the material is isotropic (same permeability in all flow directions) or anisotropic (permeability depends on flow direction). When the material is isotropic, the permeability entered in the X-direction is used in all directions. For an anisotropic material, the permeabilities in the three Cartesian directions are needed.



The local orientation of porous materials with anisotropic properties is not considered. You must select the **SimpleFFT** solver if anisotropic permeabilities are entered.

Flow Experiment

On the **Flow Experiment** tab, define where and how fast the flow motion is directed and set the boundary conditions.

Flow Settings

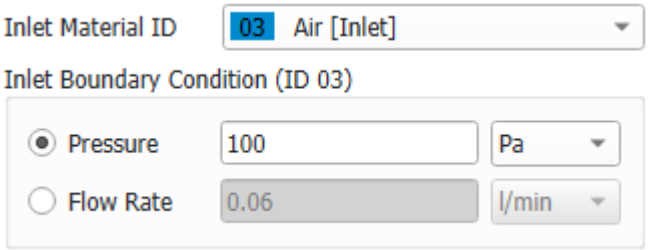
The **Flow Direction** inside the structure will always be **From Inlet to Outlet**.

The **Flow Motion** can either be creeping or fast flow. In the creeping flow case, the Stokes[8] equations are solved to determine the flow field. If fast flow is chosen, the Navier-Stokes[7] equations are solved to determine the flow field.

As the Stokes equation is a simplification of the Navier-Stokes equation, choosing **Fast Flow (Navier-Stokes-Brinkman)** will give correct results also in the case of slow, creeping flow. Solving the Navier-Stokes equations is computationally more expensive than solving the Stokes equations, so it is advisable to choose **Creeping Flow (Stokes-Brinkman)** whenever justified.

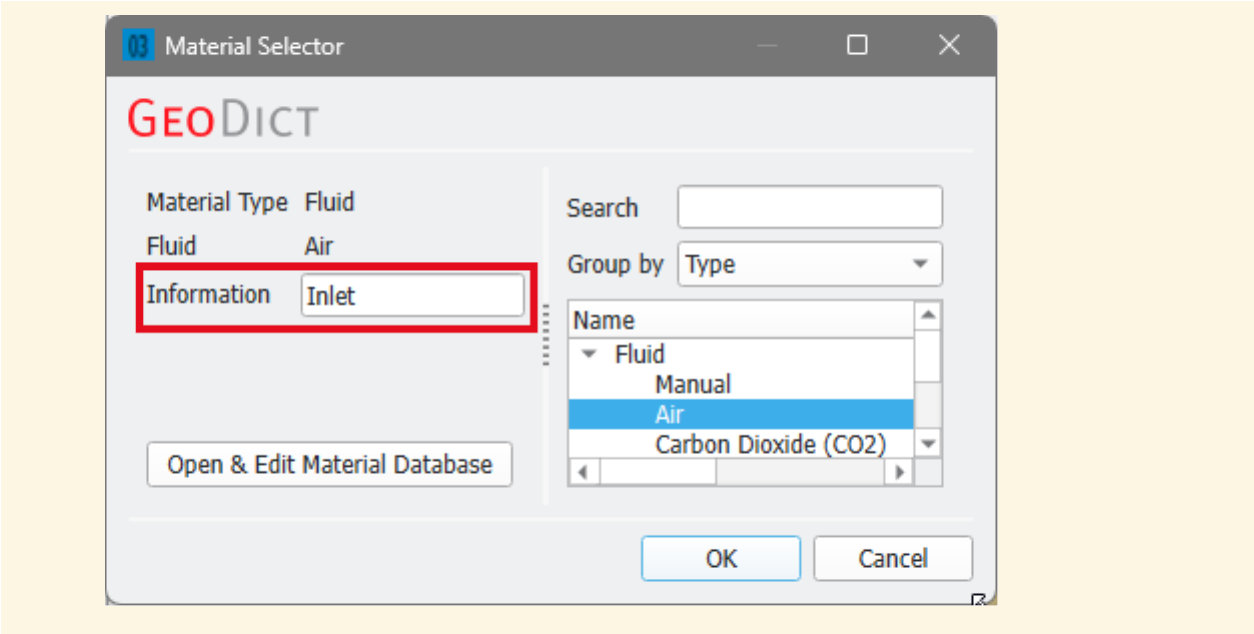
Inlet Material ID and Inlet Boundary Condition

Select the desired **Inlet Material ID** from the pull-down menu. If one of the material IDs additionally has the **Information** Inlet or Inflow, it is preselected by default.



Then, either define the average **Pressure** at the inlet or the **Flow Rate** at the inlet for the experiment.

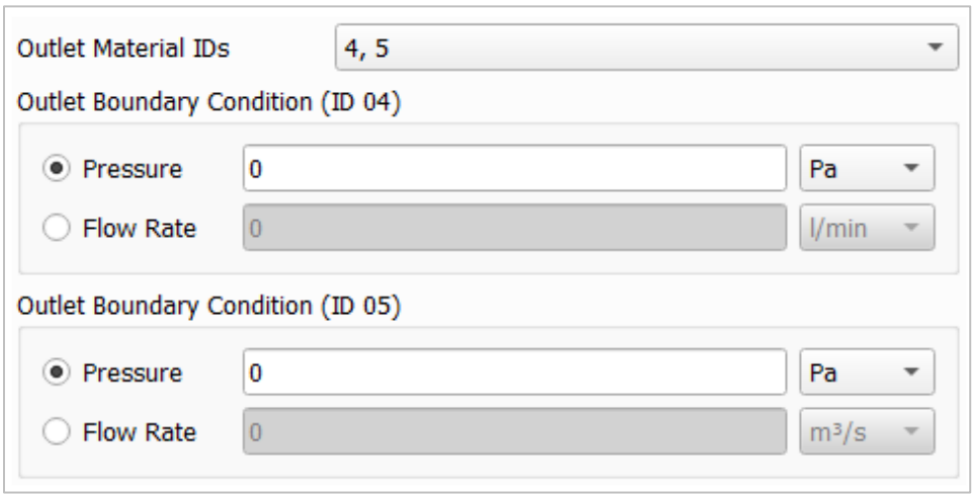
Know how! You can add additional information to a material ID while editing the materials either in the Constituent Materials tab or in the **Constituent Materials** dialog available by selecting **Settings** → **Select Constituent Materials** from the menu bar in GeoDict. Click on the material ID to open the **Material Selector** and enter the information on the left.



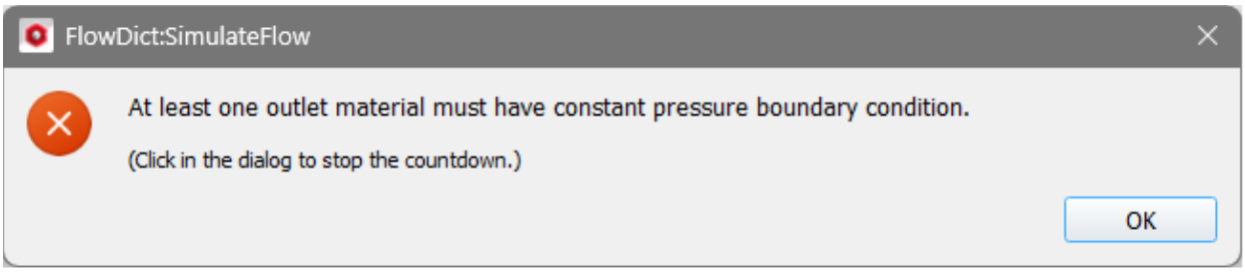
Outlet Material IDs and Outlet Boundary Conditions

As **Outlet Material IDs** more than one material ID can be selected. If a material ID additionally has the **Information** Outlet or Outflow, it is preselected by default.

Set the **Pressure** for at least one outlet ID. It must be smaller than at the inlet, since pressure decreases from inlet to outlet. From the pull-down menus different units can be selected. If more than one outlet ID is selected, it is possible to define the **Flow Rate** for all outlet IDs but one.



If all outlet IDs are defined by **Flow Rate**, an error message appears.



Conditions for Inlet and Outlet Definition.

- While inlet and outlet materials must have different material IDs, they must have the same material (i.e., fluid) as the pore space inside the structure.
- Inlet and outlet can be thicker than 1 voxel and do not need to be axis-aligned. Pore voxels at the domain boundary are allowed.
- Only one connected pore system is allowed:
  - It must contain all fluids and porous materials as well as inflow and outflow areas.
  - Inlet and outlet are not allowed to be connected via an outer fluid.

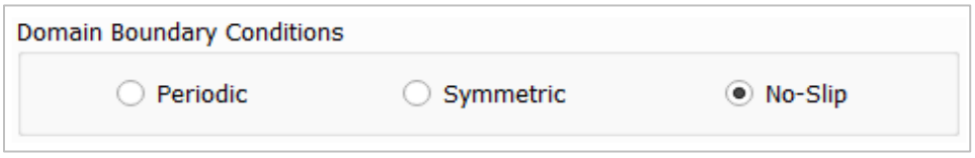
If the inlet or outlet are not set correctly, then an error message appears when starting the simulation

Domain Boundary Conditions

Set the boundary conditions in all directions to be **Periodic**, **Symmetric**, or **No-Slip**. More details on these settings can be found here 32.

The domain boundary conditions do not apply to the selected inlet and outlet areas. They apply to all fluid voxels at the domain boundaries that are neither inlet nor outlet. In cases where the 3D structure model contains the complete casing or housing, no such fluid voxels at the domain boundary exist, and the selection has no effect.

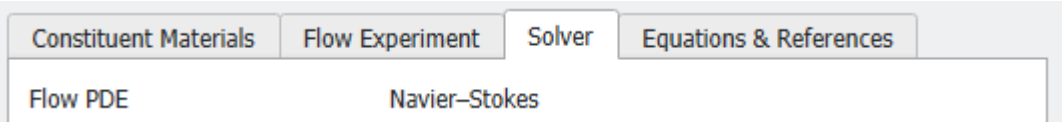
If the simulation is done on one half or one quarter of a symmetric 3D structure, **Symmetric** boundary conditions must be used.



Solver

In the first line of the **Solver** tab the **Flow PDE** to solve is shown. It changes automatically with the **Flow Motion** set in the Flow Experiment 104 tab and with the materials of the structure, i.e., if porous materials are found, the Brinkman 9 term is added.

Here, the LIR 12 solver is always used.



Parameters of the Solver tab

- Simulation Stopping Criterion 108
  Define when the computation is stopped.
- Restart Save Interval 110
  Configure an auto-save interval.
- Parallelization 110
  Parallelize the FlowDict computations.
- Restart from a *.gdr File 110
  Continue a previous FlowDict computation.
- Discard PDE Solver Files 111
  Delete intermediate files.
- Advanced Options 111
  Additional solver options.

Simulation Stopping Criterion

The equations are solved by an iterative approach.

Iterative Solver

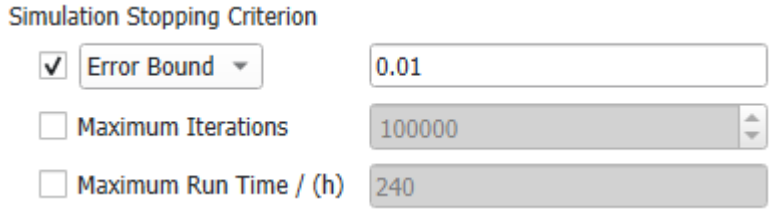
The basic idea of an iterative method is to

1. Start with some initial guess for the unknown values
2. Improve the current values in each iterative step
3. Repeat the iterative process until one of the stopping criteria is reached.

The iterative process is controlled by setting the values and activation for **Error Bound, Tolerance**, **Maximal Iterations**, and **Maximal Run Time (h**). The stopping criteria apply for each computational direction individually.

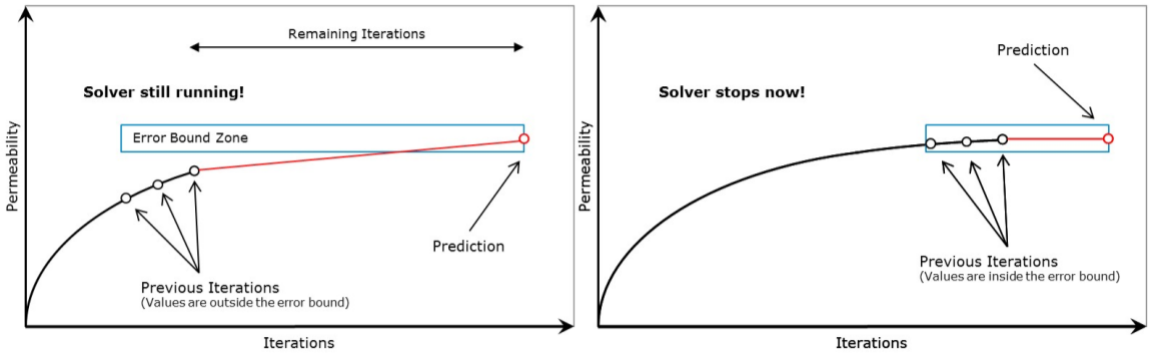
If multiple stopping criteria are selected, the first criterion that is reached causes the solver to stop.

The stopping criterion that has been reached can be viewed in the Result Viewer of the GeoDict result file (*.gdr) under the **Results Report** tab.



Error Bound

The stopping criterion **Error Bound** uses the result of previous iterations, and predicts the final solution based on linear and quadratic extrapolation. The solver stops if the relative difference between computed and predicted solution is smaller than the specified error bound. The stopping criterion recognizes oscillations in the convergence behavior and prevents premature stopping at local minima or maxima. A damped convergence curve is fit through the oscillating curve and the solver stops then regarding the damped convergence curve.



If the Krylov method (under LIR - Advanced Options) is activated, the definition of **Error Bound** is somewhat different. Here, no prediction is made, instead the continuity between neighboring cells and the conservation of mass is checked. The maximal value is normalized by the mean flow and if this value is smaller than the **Error Bound** the simulation stops.

Tolerance

The **Tolerance** stopping criterion looks for stagnation of the method when the process becomes stationary, i.e., the improvement in the permeability value becomes extremely small from iteration to iteration.

In each iteration, the solver checks for the current computed value x_c against the values x_{c-i} of the last 100 iterations if there are any changes.

$$\max_{i\in[1,100]} \frac{x_c - x_{c-i}}{x_{c-i}} < \text{tolerance} \tag{40}$$

The computation is stopped if the maximal relative change is smaller than the value entered for **Tolerance**. When there is doubt about the quality of the solution, decrease the **Tolerance** value by a factor of ten. The drawback of this criterion is that the solver sometimes might stop too early in case of slow convergence rate or at local extrema of oscillatory convergence curves.

Maximum Iterations and Maximum Run Time

Use the **Maximum Iterations** value or the **Maximum Run Time (h)** stopping criteria causes the solver to stop if the maximal number of iterations and/or the maximal runtime (in hours) is exceeded.

When the solver stops because one of these criteria has been reached, no guarantee on the quality of solution can be given. In this case, a warning is printed into the report. The following possibilities might help:

- Check the corresponding .log file to see how large the residual values and the permeability values are. If the permeability values are in reasonable range and they do not have changed during the last iterations, you may decide to use the current result.
- Double-check the structure and parameter values. Unphysical parameters or too rough resolution of the structure (leading, e.g., to artificial unconnected components) can cause an iterative solver to fail.

Know how! We recommend to use the **Error Bound** criterion. This stopping criterion approximates the "relative deviation from the final solution".

Restart Save Interval

The calculations run by the solvers can be restarted from saved intermediate result files and the interval between auto-saves can be configured from the value entered in **Restart Save Interval (h)**.

Parallelization

The simulation process can be parallelized to speed-up the computation process. It depends on your license, how many parallel processes you may use. The solver benchmarks 13 illustrate the speed-up that you can achieve.

The **Parallelization Options** dialog box opens when clicking the **Edit...** button, to choose between **Sequential**, **Parallel (Shared Memory)**, and **Automatic Maximum of Threads**.

For details on how to set up and run parallel computations, consult the High Performance Computing chapter.

Restart from a *.gdr File

In some situations, it may be useful to re-use previously computed results and save a great amount of time by restarting the process from an existing GeoDict result file (*.gdr). Typical examples would be

- non-sufficient accuracy of some computation, when it is suspected that more iterations may

improve the quality of the result,

- trying another boundary condition (e.g., Dirichlet instead of Periodic or other way around),
- continuing an iteration process,
- or using similar parameter values for the computation.

To restart and initialize the solver from a result file (*.gdr), check **Restart from .gdr File** and enter the name of a file, or **Browse** for it in the project folder.

When using **Restart From .gdr File** only the volume field (*.vap file) associated to the loaded *.gdr file will be used to continue the simulation.

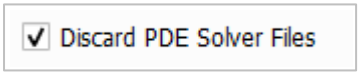
Important! The *.vap file is saved in the result folder of a previous run. Make sure that this file was not deleted from the results folder. If the *.vap file is not available the restart option cannot be used.

No other parameters from the options dialog will be loaded from the *.gdr file, but it is possible to load them by clicking **Load Parameters** in the Result Viewer. This way the simulation can be continued while using the same settings as in the initial simulation.

Important! The structure from the run of which you want to use the result file must be loaded into GeoDict. If this is not the case, an error message is displayed.

Discard PDE Solver Files

Checking **Discard PDE Solver Files** causes the deletion of all intermediate computation files, such as log files and flow field files.

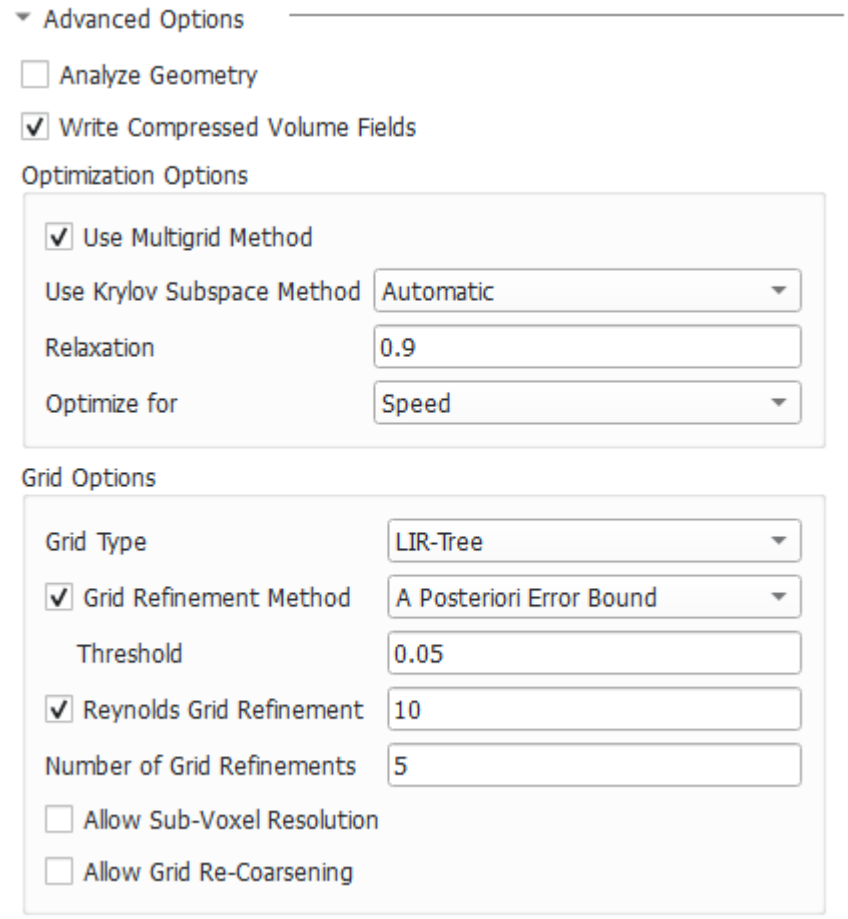


Only the content of the final GeoDict result file (*.gdr) is stored.

While having the benefit of saving hard disk storage place, discarding intermediate solver files has also the effect of disabling the 3D visualization of the results.

Advanced Options

Unfold **Advanced Options** to gain access to several advanced solver options.



Analyze Geometry

Checking **Analyze Geometry** performs an analysis of the geometry before the solver computations.

If no percolation path through the structure is found, the partial differential equation does not need to be solved, and a permeability of 0.0 is directly reported as the solution.

Furthermore, the solver operates on the whole structure, regardless of which parts are connected or unconnected. However, unconnected components are not transport relevant, thus the effort of solving in these parts is not necessary.

For these reasons, a geometrical analysis is routinely run to determine whether a through path exists. After the analysis is finished, unconnected components are removed from the computational grid.

This may speed up the computations but requires some time for the geometrical analysis, especially for very large structures. So if you know beforehand that there are

no unrelevant parts or that the structure has been processed to eliminate them, the geometry analysis can be switched off by unchecking **Analyze Geometry**.

Write Compressed Volume Fields

The LIR solver uses a very memory efficient adaptive grid structure for the simulations.

If the option **Write Compressed Volume Fields** is checked, then the adaptive grid is used as compression method for writing out *.vap files. This option allows to save 80-90% space on hard drive. The runtime for writing *.vap files is also reduced significantly. But the runtime for loading and uncompressing of compressed *.vap is increased by the amount of runtime that was saved for writing out compressed *.vap files.

If the option **Write Compressed Volume Fields** is not checked, then a usual regular grid is used for writing out *.vap files.

Optimization Options

The **Multigrid Method** (see, e.g., Wesseling, 2004) was introduced to speed-up the computation and reduce the runtime significantly. The main idea of **Multigrid** is the usage of multiple coarser adaptive grids to speed up convergence behavior but requires only little more memory.

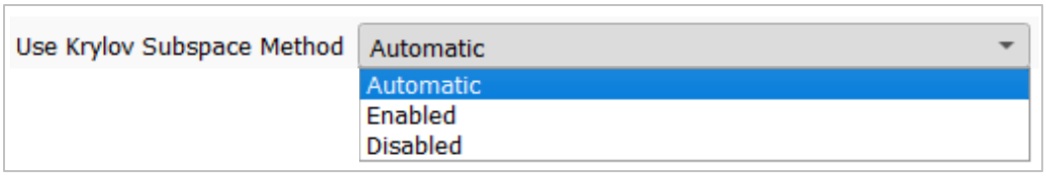
The method is available to solve the Stokes and Stokes–Brinkman equations as well as for solving mechanics, diffusion, thermal, and electrical conduction and is enabled by default.

Another speed-up option to accelerate the convergence behavior of the LIR solver is called **Krylov Subspace Method**.

The runtime of the LIR solver depends on many different properties of the structure and the simulation parameters. The BiCGStab algorithm is used, which can reduce the runtime for challenging simulation very drastically.

Note! Using the BiCGStab method approximately doubles the amount of RAM needed for the computation.

Unfortunately, the Krylov method is not always faster than a simulation without the Krylov method and therefore we introduced an **Automatic** mode which uses some heuristics to choose the most efficient method based on structure, material parameters, and boundary conditions automatically. Of course, it is possible to explicitly enable (**Enabled**) or disable (**Disabled**) the method.

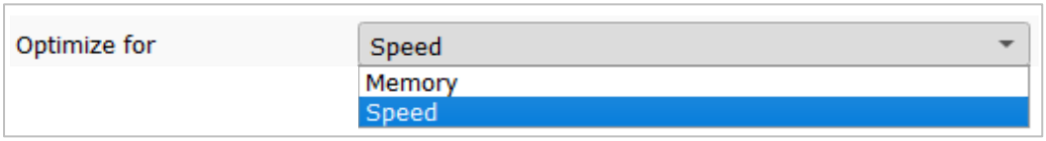


Note! In case that the Krylov subspace method (BICGStab) is used, the **Relaxation** may also be adapted automatically.

Depending on the material parameters and geometry of the structure, the underlying mathematical problem can vary in complexity, thus influencing the behavior of the solver. The more complex the problem is, the more stable the solver settings should be.

With the **Relaxation** number, the solver is adjusted from **Stable** (which results in higher number of iterations, slower time stepping, and longer solver run times), to **Fast**, which makes the solver run less iterations but implies the risk that the solver does not converge. The **Relaxation** is a parameter of the SOR method and must be between 0 and 2 to ensure convergence. For relaxation values smaller than one (<1.0), the simulation is more stable. For relaxation values larger than one (>1.0), the simulation converges faster.

The **LIR** solver can **Optimize for Speed** or **Memory**.



- If **Speed** is chosen, the solver constructs additional optimization structures. The runtime is decreased by up to 30% but requires up to 50% more memory compared to the other option.
- If **Memory** is chosen, the runtime is increased by up to 40% but the solver requires up to 50% less memory.

Grid Options

The **Grid Type** decides what kind of tree structure is used for the simulation.

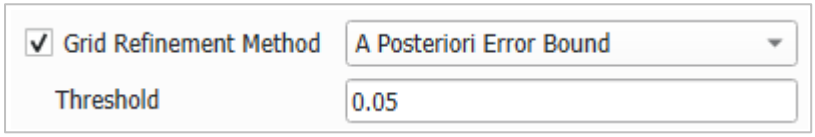
The default option is **LIR-Tree** and should always be used. The solver uses an adaptive grid structure called LIR-tree and needs up to 10 times less runtime and memory compared to the **Regular Grid** option.

The solver can analyze the result field during the computation and improves the adaptive grid in places where more accuracy is needed. The LIR solver splits cells where a high gradient occurs.

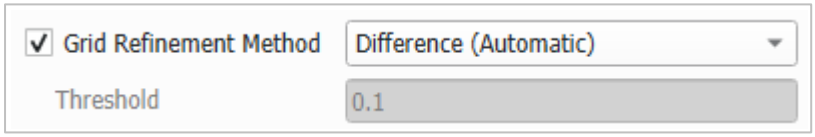
The solver can analyze the computed field and refines the adaptive grid during the computation at locations where more accuracy is needed. The LIR solver splits cells where a high gradients occur when the **Grid Refinement Method** is enabled. In this case, the additional parameters **Number of Grid Refinements**, **Allow Sub-Voxel Resolution**, and **Allow Grid Re-Coarsening** become available.

Select one of the three available refinement methods from the drop-down menu.

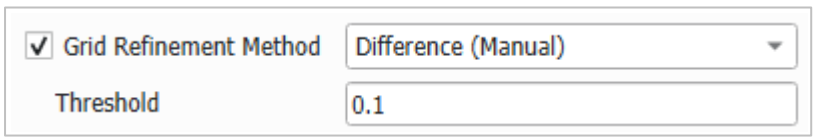
When **A Posteriori Error Bound** is selected, the solver targets the specified accuracy **Threshold**. While the **Error Bound** (set as Simulation Stopping Criterion) determines the relative error in the solution of the linear system, **A Posteriori Error Bound** refers to the relative error estimated by comparing high-order and low-order discrete solutions. The accuracy **Threshold** refers to the relative error to the analytical solution and the value must be between 0.0 and 1.0.



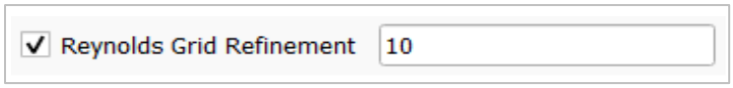
Choosing **Difference (Automatic)** leads to computational cells being split when the difference in values between neighboring cells exceeds a certain **Threshold**. The solver automatically chooses all internal parameters based on the structure and simulation settings. For this option, cells are split where the current gradient is greater than the **Threshold** multiplied by the maximum gradient, where the threshold is determined automatically.



Selecting **Difference (Manual)** also causes the computational cells to be split when the difference in values between neighboring cells exceeds the specified **Threshold**. You can specify all parameters (Threshold and Number of Grid Refinements) for the grid refinement manually. For this option, cells are split where the current gradient is greater than the **Threshold** multiplied by the maximum gradient. The **Threshold** value must be between 0.0 and 1.0. The recommended value range is between 0.05 and 0.1.



Activate **Reynolds Grid Refinement** to use the local Reynolds number (Re) for adaptive grid refinement.



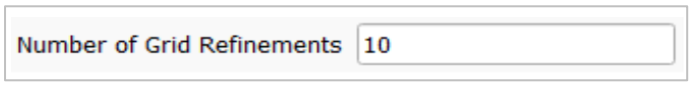
A grid cell is split if the local Reynolds number, i.e., the ratio between inertial and viscous forces exceeds the given threshold. The Reynolds number is defined as:

$$\mathrm{Re} = \frac{\rho \mathrm{u} \mathrm{L}}{\mu} \quad (41)$$

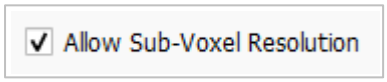
Reynolds Number

where ρ is the density of the fluid (SI units: kg/m^3), u is the velocity of the fluid (m/s), μ is the dynamic viscosity of the fluid (Pa·s or N·s/m^2 or kg/m·s), L is a characteristic length (m). For the local Reynolds number, the size of the grid cell is used for the characteristic length.

The **Number of Grid Refinements** controls the maximum number of grid refinements that the solver can perform during the simulation. The value should be set between 0 and 10, where a value of 0 means that no grid refinements will be made. Grid refinements may increase the number of iterations, runtime, and memory requirements.



When **Allow Sub-Voxel Resolution** is enabled, the solver is allowed to split computational cells to sizes smaller than the voxel length.



This feature is beneficial for low-porosity structures for which the pore throats require finer resolution or for modeling fast Navier-Stokes flows with strong vortices.

Enabling this feature may increase the number of iterations, runtime, and memory requirements.

Check **Allow Grid Re-Coarsening** to allow the solver to automatically revert the grid refinement and coarsen the computational cells.

This means, grid refinement is done temporarily. Afterwards, the cells are merged as soon as accuracy is not needed anymore. This reduces the memory and runtime requirements.

1.5.2 Results

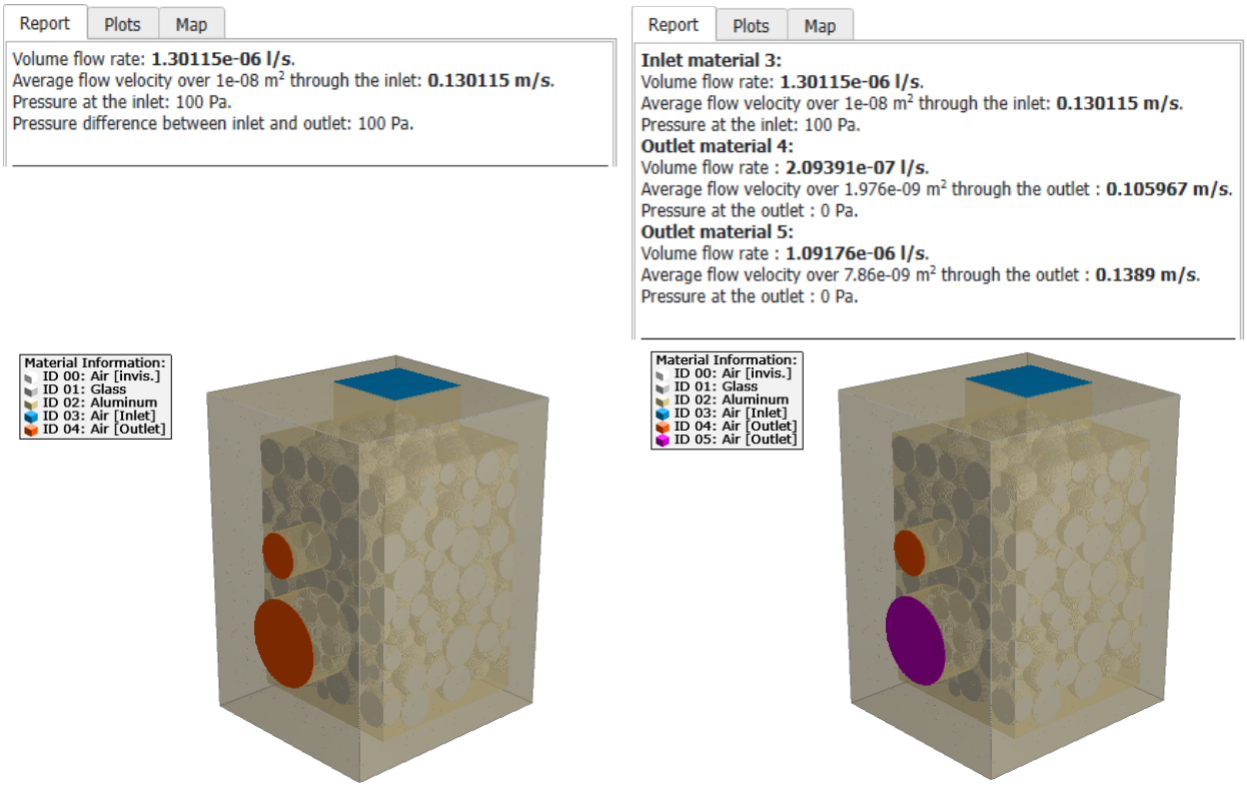
After the solver has finished, the result file (.gdr) is saved in the project folder and the **Result Viewer** automatically opens. Find detailed information on **Result Viewer** options in the Result Viewer user guide.

Results of the Simulate Flow Experiment command

- Report 117
  Shows the computed results.
- Plots 118
  View result plots.
- Flow Visualization 119
  Load results for a graphical visualization.

Report

The **Report** tab under the **Results** tab shows the computed results. Here the **Volume flow rate** (you can change the unit in the post-processing panel on the left), the **Average flow velocity**, and the **Pressure at inlet/outlet** is given. When only one outlet material ID is defined the **Pressure difference between inlet and outlet** is also provided. If several IDs are defined as outlet, the results are shown for each outlet material ID.



Information about the convergence of the iterative solver is also displayed: **Threads**, **Iterations**, **Number of cells**, **Runtime**, **Memory usage**, and the applied **Stopping criterion**.

Flow solver information

Threads	12
Iterations	1750
Number of cells	2653749
Runtime	9.79937 min
Memory usage	1.688 GiB
Stopping criterion	error bound

Total runtime: 00:09:48, Total memory usage: 2.645 GiB

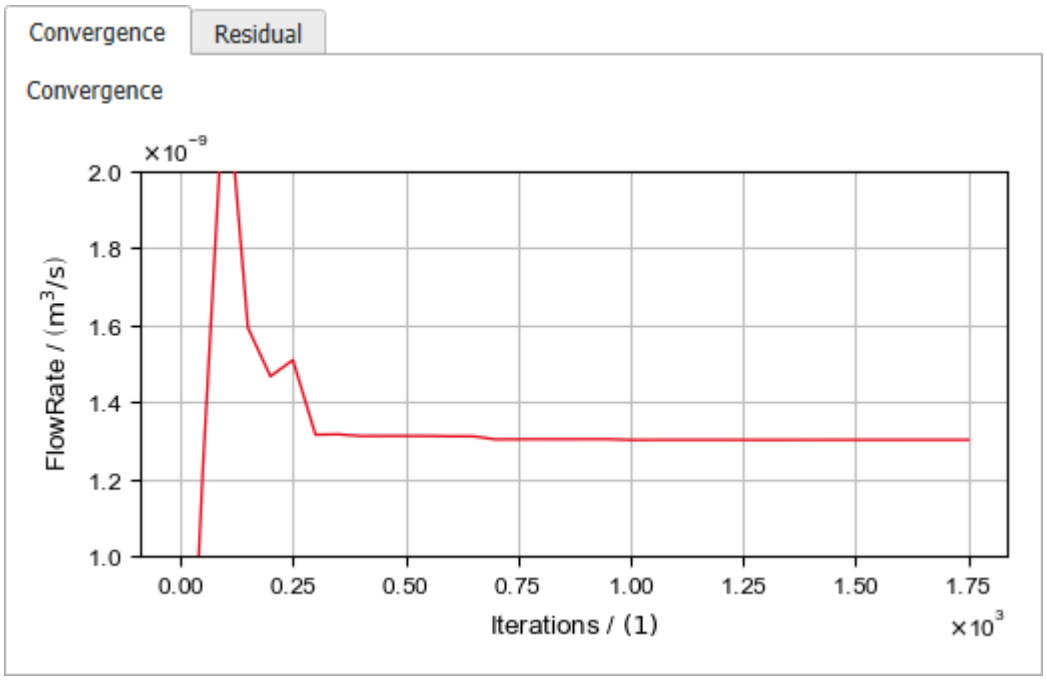
In the last line the **Total runtime** and **Total memory usage** are given. Here not only the resources used by the solver are taken into account but also writing volume fields and structure files or already loaded volume fields from previous simulations.

Plots

Under the **Plots** tab, only the **Convergence** and **Residual** plots are available:

Convergence

The **Convergence** tab plots the computed flow rate at each iteration.

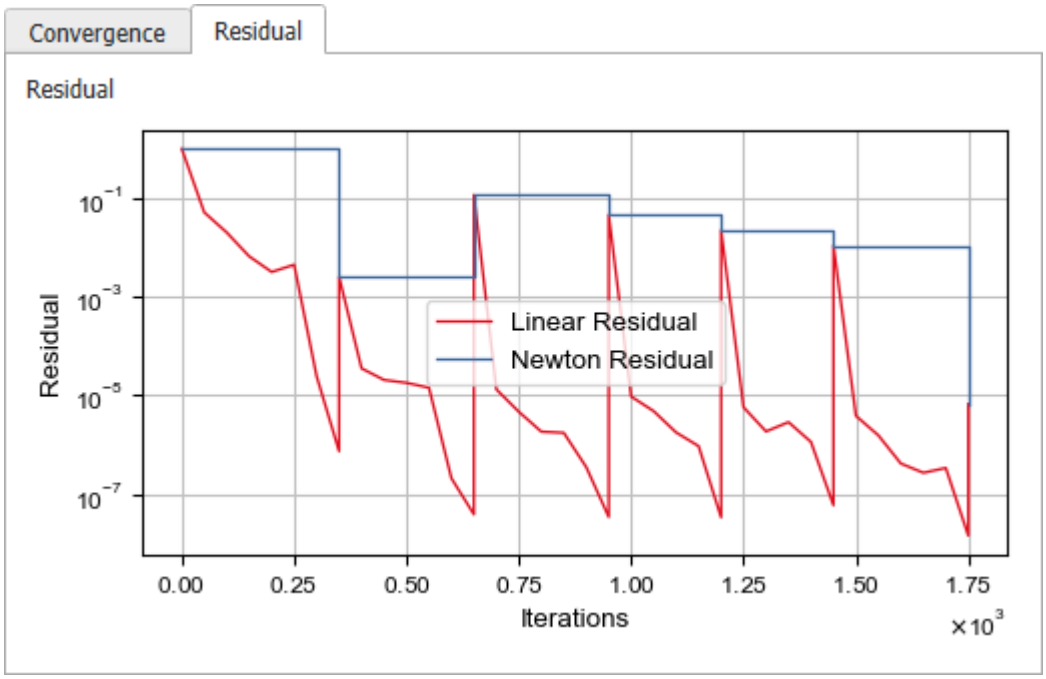


Residual

The **Residual** plots tab shows the **Linear Residual** and the **Newton Residual** at given **Iteration** values. The **Newton Residual** is defined as the residual while solving non-

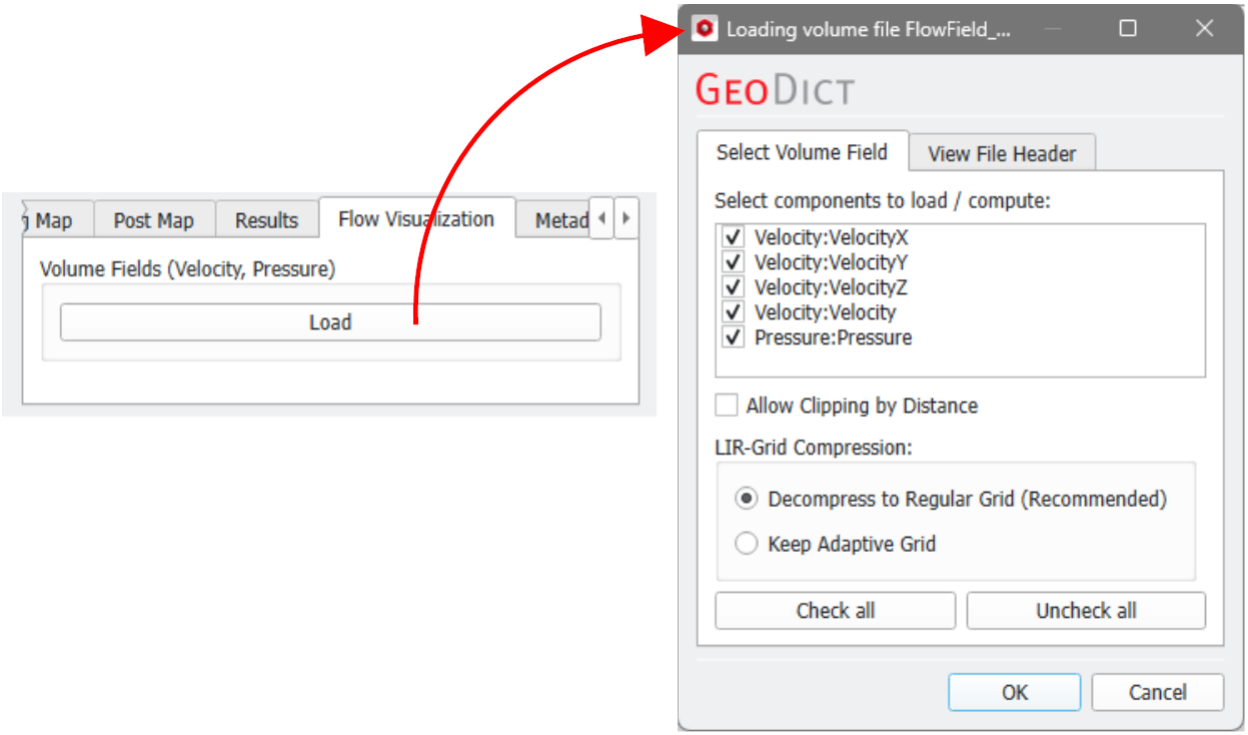
linear equations with Newton's method. The Newton residual curve changes, when the equation is solved, while the linear residual changes with every iteration while solving a linear equation system.

This plot is available only when the Krylov method is used.



Flow Visualization

Select the **Flow Visualization** tab to load the computed flow field (velocity and pressure, *.vap). A dialog pops up, that let's you select which of the stored field components **VelocityX**, **VelocityY**, **VelocityZ**, **Velocity**, and/or **Pressure** you want to load and visualize.



Check **Allow Clipping by Distance** if you want to use this feature in the visualization. If selected, an additional data field is computed and loaded that contains the Euclidean distance to the solid structure. You can use this data field, e.g., to print arrows only with a certain distance from the solids. If you do not need this feature, do not select it, as it requires additional RAM to store the distance field.

Result files computed with the LIR solver are stored using the adaptive grid of the LIR solver (**LIR-Grid Compression**). Select if you want to interpolate the data during import to the regular voxel grid or if you want to keep the adaptive grid:

- **Decompress to Regular Grid:** More RAM is needed to load the data, and you cannot visualize the adaptive grid. This option enables the visualization with Streamlines, Arrows and Schlieren.
- **Keep Adaptive Grid**: Less RAM is needed to load the data. You can visualize the adaptive grid. You cannot use Streamlines, Arrows or Schlieren to visualize the results.

The structure that the solver used internally is loaded if **Load structure with closed pores** is checked. This structure shows which pores have been closed due to prior geometry analysis.

Visualization Options

- Volume Field: Pressure drop or velocity are visualized by a 3D color field - you can clip by threshold values or use transparency to view inside.
- Streamlines: Visualize the flow through the structure with streamlines or animated streamlets. Be aware, that you must change the **Placement** method away from the default Inflow Plane.
- Arrows: Visualize the velocity vector classically with arrows.

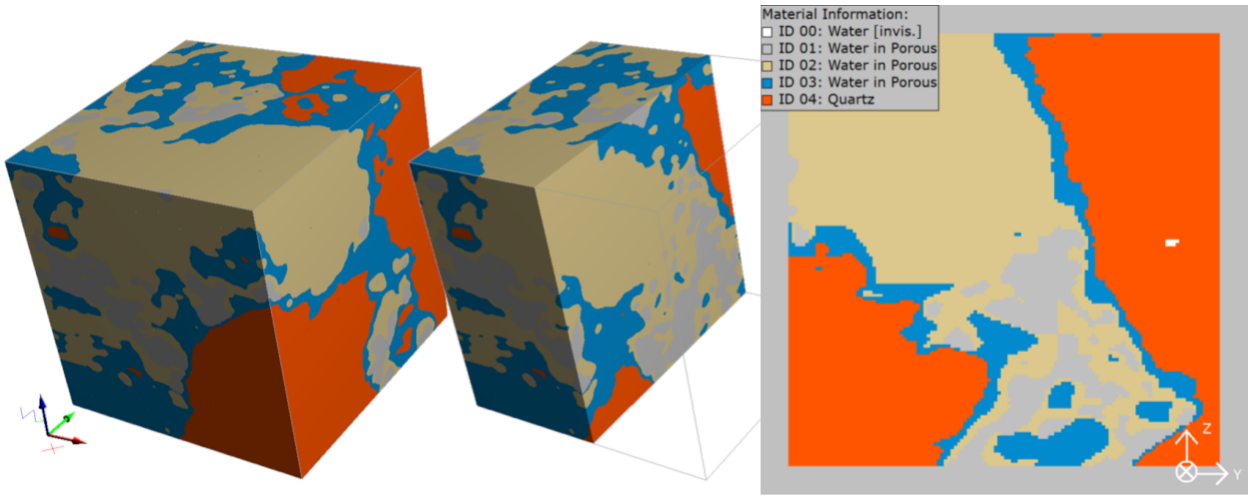
- Schlieren: This option uses Line Integral Convolution (LIC) fields to visualize vortices.

1.6 Darcy Flow

The **Darcy Flow** command determines the permeability of a 3D structure where the majority of pores is unresolved by the voxel grid.

You can use **Darcy Flow** if no through-path of fluid voxels exists and the flow has to pass through porous voxels. These porous voxels represent materials with small permeabilities where the viscous term can be neglected, e.g., rocks. In these cases, Darcy's law 20 can be applied.

The structure in this example has a voxel length of 400 µm and permeabilities of 10^{-12} m^2 and smaller.

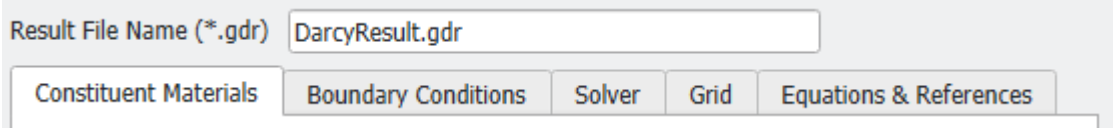


The Darcy Flow command

- Options 122
  How to set the input parameters of the command.
- Results 140
  View the simulation results.

1.6.1 Options

Select **Darcy Flow** from the pull-down menu and click the **Option**'s **Edit...** button to open a dialog to enter or edit the simulation options. Choose a **Result File Name** for the result file (*.gdr) fitting to your current project.



The parameters are organized under the tabs: **Constituent Materials**, **Boundary Conditions**, **Solver**, and **Grid**. The **Equations & References** tab gives a summary of the used equations and symbols which are explained in more detail here 10.

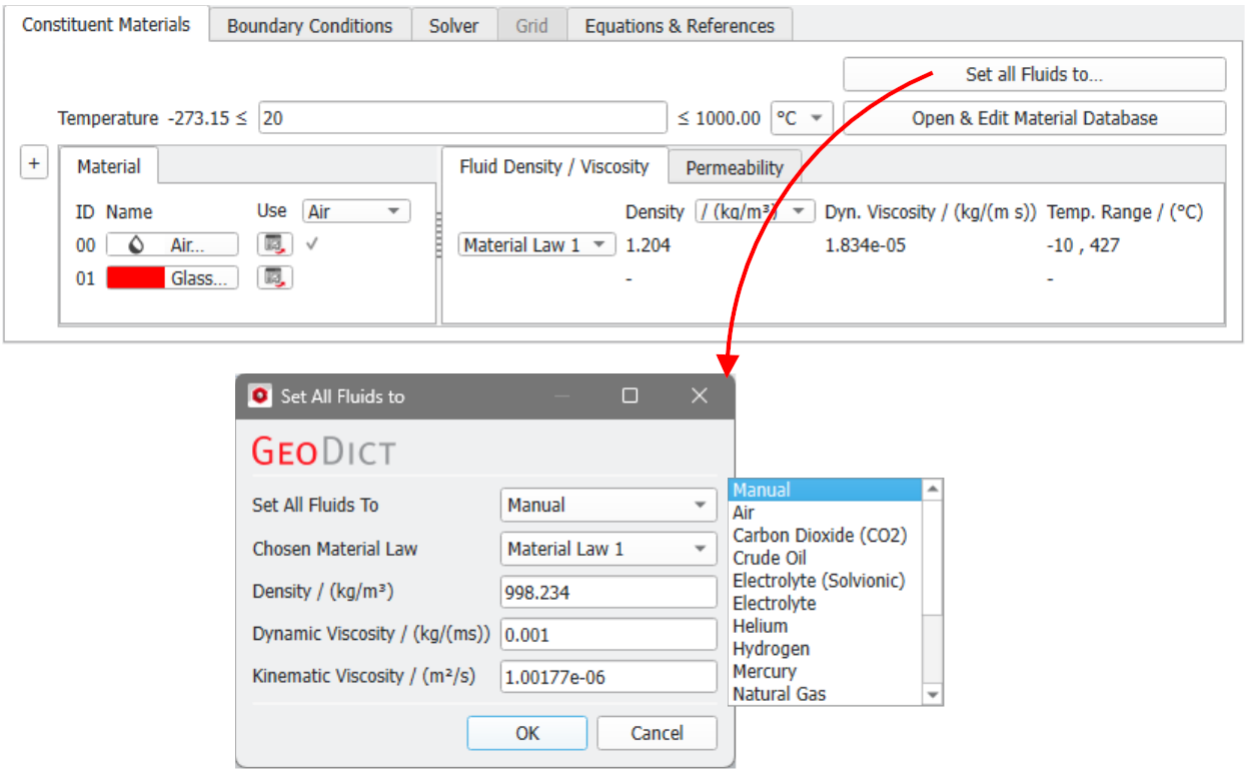
Tabs of the Darcy Flow dialog

- Constituent Materials 123
  Define the materials in the structure.
- Boundary Conditions 126
  Set up the boundary value problem that describes your experiment.
- Solver 129
  Select a flow solver.
- Grid 139
  Adjust the computational grid.

Constituent Materials

For the flow solver's computation, the materials in the structure and the fluids flowing through it need to be defined under the **Constituent Materials** tab, so that their physical properties parameters are taken into account.

To set the fluid flowing through all pore and porous materials, click **Set all Fluids to** and select it from the pull-down menu.



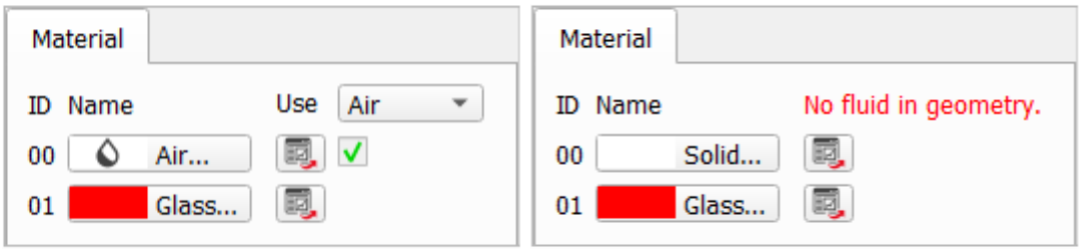
If one of the predefined fluids is used, the values for **Density**, **Dynamic Viscosity**, and **Kinematic Viscosity** are taken from the GeoDict Material Database. They are dependent on the given **Temperature** value and the **Chosen Material Law**. If **Manual** is selected, these values can be entered manually.

Any values entered manually, e.g., for a special type of oil, can also be added to the material database through the **Open & Edit Material Database** button. Additionally, different

properties for the same material can be saved as different **Material Laws** and can then be selected from the pull-down menu in the **Set all Fluids to** dialog. **Material Law 1** is the default shipped with GeoDict. Click here for more information on editing, expanding, and using the GeoDict Material Database.

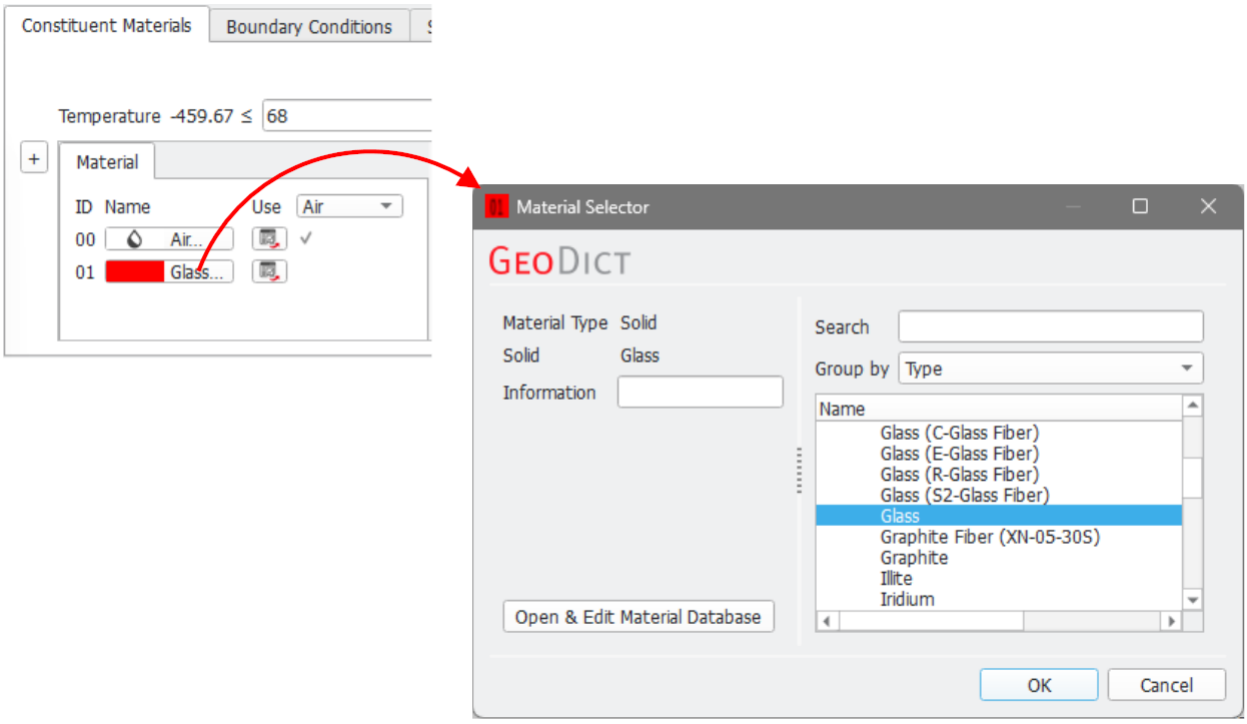
The **Temperature** is selectable in Kelvin (K), Celsius (°C), and Fahrenheit (F), and has a default value of 293.15 K (20.0°C, 68 F).

Under the **Material** subtab, select the fluid used in the simulation from the **Use** pull-down menu. If other fluids are contained in the structure, they will be treated as immobile (solid) during the simulation. If the structure contains no fluid material, **No fluid in geometry** is displayed in the top of the **Material** tab. Select a material for the pore phase by clicking on the corresponding material button.

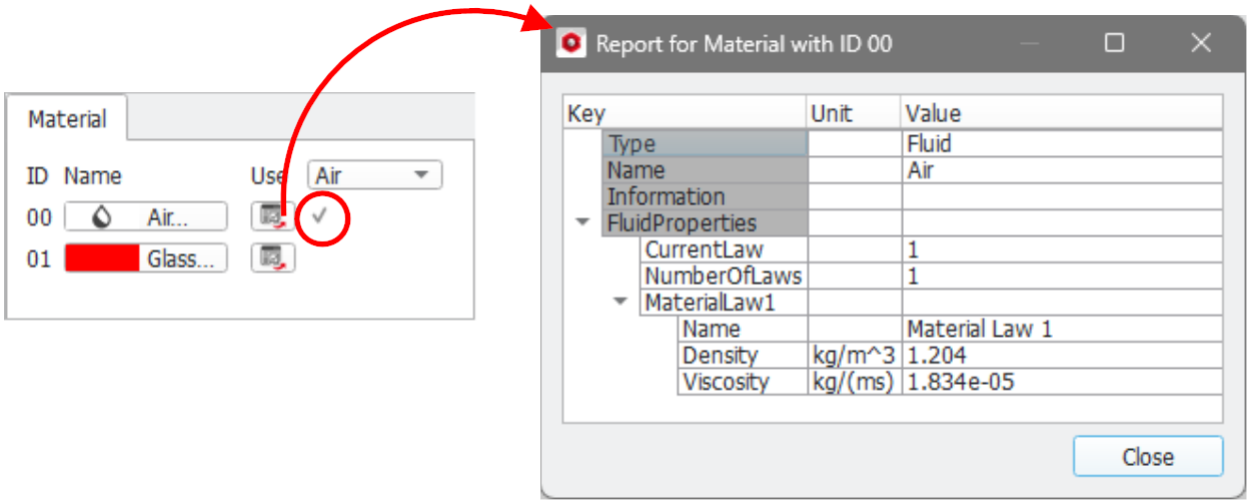


The density and the dynamic viscosity of this fluid, at the selected temperature, is shown under the **Fluid Density/Viscosity** subtab.

The **Material** subtab shows also the material **IDs** assigned to the materials (00, 01, 02, etc.). For each material ID, you can open the **Material Selector** dialog and select a material by clicking on the corresponding material buttons.

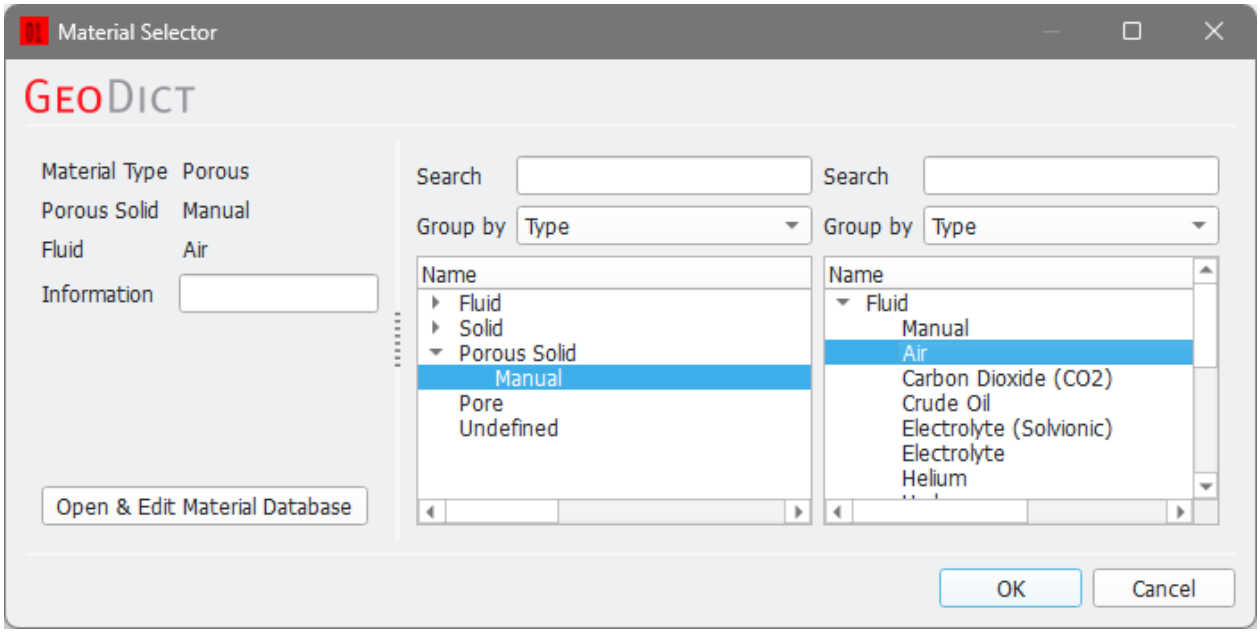


Click the **Material Report** button to see the relevant parameters for the material with a selected ID.

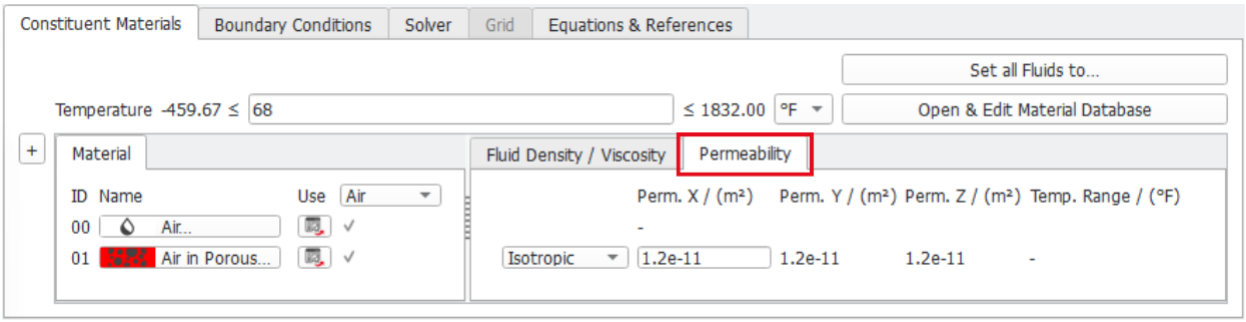


Fluid flow can happen in pores and porous materials, but not in solid materials. The check sign next to the **Material Report** icon marks all materials in which flow happens.

For porous voxels, set the **Material Type** (in the Material Selector dialog) to **Porous** and choose the fluid that occupies the pores.



The permeability of porous voxels is set under the **Permeability** subtab. You can define if the material is isotropic (same permeability in all flow directions) or anisotropic (permeability depends on flow direction). When the material is isotropic, the permeability entered in the X-direction is used in all directions. For an anisotropic material, the permeabilities in the three Cartesian directions are needed.



The local orientation of porous materials with anisotropic properties is not considered. You must select the **SimpleFFT** solver if anisotropic permeabilities are entered.

Boundary Conditions

To simulate the flow through the domain, GeoDict solves a boundary-value problem. In the fluid-filled part of the domain, the flow is governed by a differential equation. The boundary values describe what happens at the boundaries, and are a necessary and important part of the problem description. Boundaries are the six faces of the cuboid (X-, X+, Y-, Y+, Z-, Z+), and the pore-solid walls within the structure.

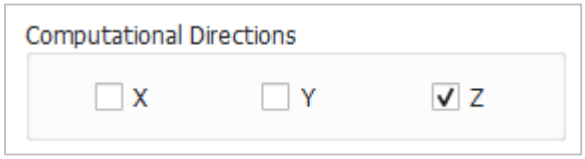
For the six cuboid faces, FlowDict distinguishes between the domain faces in the computational direction (also called flow direction or longitudinal direction) and the domain faces in the two other space directions (called transversal or tangential directions). The computational direction is the direction in which pressure drop and flow rates are measured.

Computational Directions

As **Computational Directions**, select the directions for which you want to calculate the flow. To obtain the full 3x3 permeability matrix and a pressure drop value for each direction in the results file, select all three directions.

For every selected direction, a boundary value problem is solved in which the **Domain Boundary Conditions in Longitudinal Direction** and the corresponding **Experiment Input/Output** are applied in the computational direction, and the **Domain Boundary Conditions in Transverse Directions** are applied on the other two space directions.

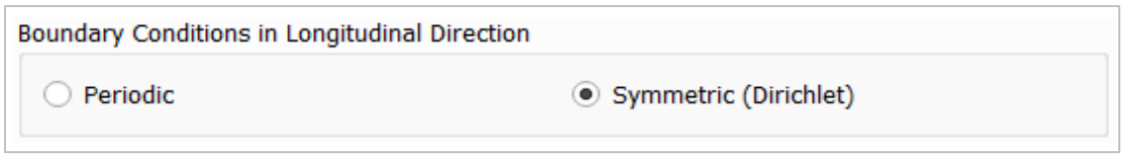
Any directions not selected will not be calculated and will be marked as 'unknown' in the permeability matrix in the **Report** tab of the result file.



Boundary Conditions in Longitudinal Direction

For Darcy Flow, the **Boundary Conditions in Longitudinal Direction** can be **Periodic** or **Symmetric (Dirichlet)**. As Darcy Flow is used for structures with unresolved pores and low permeabilities, no inflow or outflow region should be given. Use symmetric

boundary conditions for non-periodic structures if the percentage of solid voxels is high at the inlet. Otherwise, use periodic boundary conditions, as the simulation runtime is much faster.

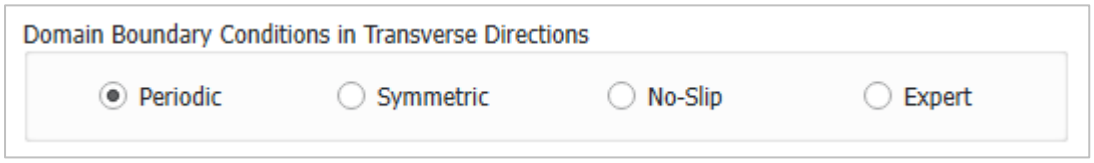


Experiment Input

In Darcy Flow simulations, the average **Pressure at** the **Inlet** surface and the average **Pressure at** the **Outlet** surface can be defined, where the pressure at outlet must be smaller than the pressure at inlet, since the pressure decreases from inlet to outlet. Select the desired pressure unit from the pull-down menu.

Boundary Conditions in Transverse Directions

The **Domain Boundary Conditions in Transverse Directions** can be **Periodic**, **Symmetric**, **No-Slip**, or **Expert**.



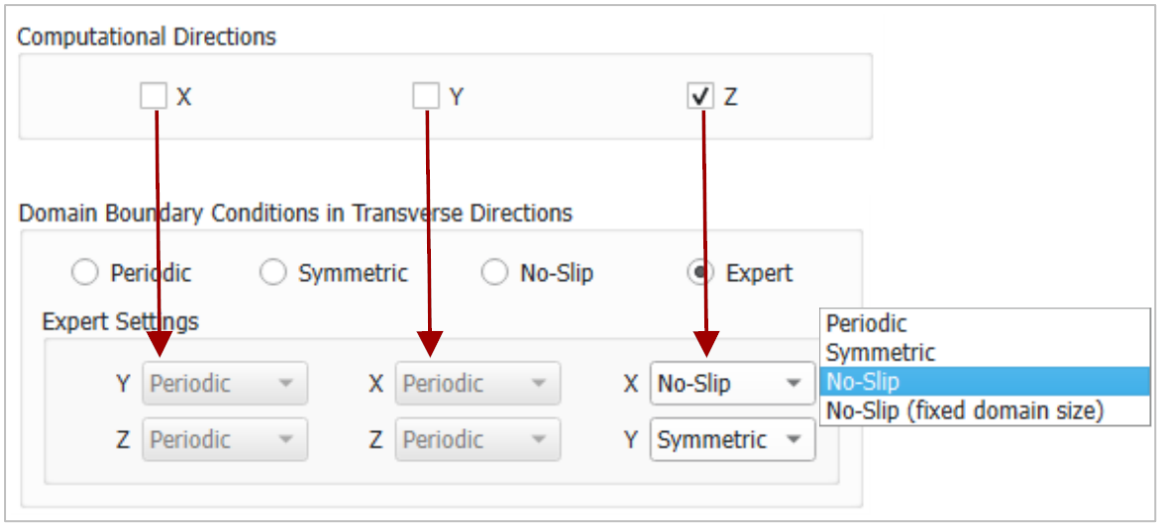
With **Periodic** boundary conditions, the structure is repeated in the transverse directions.

With **Symmetric** boundary conditions, the structure is mirrored at the domain boundary. The LIR and EJ solver solve the symmetric boundary condition by using Neumann (or zero flux) condition at the domain boundary.

With **No-Slip** (or Encase) boundary conditions, the domain boundary is treated as closed by solid material. When **No-Slip** is used, the solver internally adds a one-voxel layer in the required direction and solves with periodic boundary conditions. This is effectively equivalent to solving the structure with casing at both sides in the transverse direction. So, the structure size in the corresponding transverse direction becomes enlarged by one voxel internally.

With **Expert**, it is possible to define different boundary conditions for each side in each computation:

The boundary conditions in the two directions transverse to the flow can also be set to be different by checking **Expert** boundary conditions. For example, when the fluid is chosen to flow in the Z-direction, the boundary conditions could be chosen to be **No-slip** in X-direction and **Symmetric** in the Y-direction.

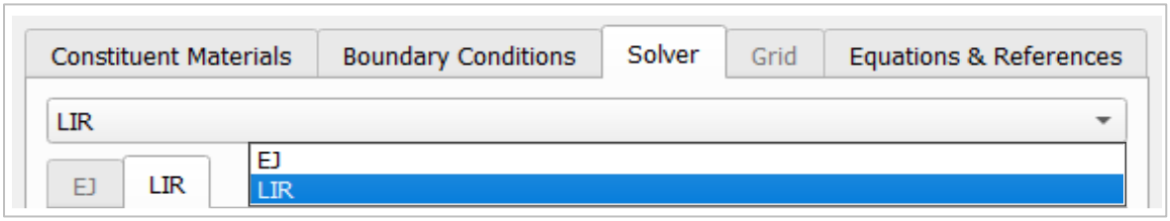


For the Darcy Flow command, additionally to **Periodic**, **Symmetric** and **No-Slip** also **No-Slip (fixed domain size)** is available.

When **No-Slip** is used, the solver internally adds a one-voxel layer in the required direction and solves with periodic boundary conditions. That effectively is equivalent to solve the structure with casing in two ends in the direction of interest. So, the size of computation in this direction becomes n+1. However, if changing the computational size is not preferable, using **No-Slip (fixed domain size)** can avoid that, then the first layer of the structure is replaced by an impermeable material.

Solver

Two different flow solvers 12 are available for solving the equation. Select the solver in the pull-down menu to activate the corresponding tab below.



The LIR solver requires the least numerical resources (RAM) and is therefore selected by default. Often, the LIR is also the fastest solver, but that may depend on the parameter selection and the 3D structure model.

Parameters of the Solver tab

- Simulation Stopping Criterion 130
  Define when the computation is stopped.
- Restart Save Interval 133
  Configure an auto-save interval.
- Parallelization 133

Parameters of the Solver tab

Parallelize the FlowDict computations.

- Restart from a *.gdr File 133
  Continue a previous FlowDict computation.
- Discard PDE Solver Files 134
  Delete intermediate files.
- Orientation Mode (for LIR) 134
  Define how to determine the local orientation.
- Advanced Options 135
  Additional solver options.

Simulation Stopping Criterion

The equations are solved by an iterative approach.

Iterative Solver

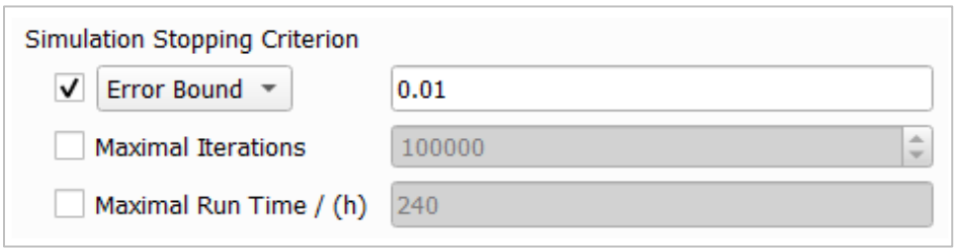
The basic idea of an iterative method is to

1. Start with some initial guess for the unknown values
2. Improve the current values in each iterative step
3. Repeat the iterative process until one of the stopping criteria is reached.

The iterative process is controlled by setting the values and activation for **Error Bound, Tolerance, Residual**, **Maximal Iterations**, and **Maximal Run Time (h**). The stopping criteria apply for each computational direction individually.

If multiple stopping criteria are selected, the first criterion that is reached causes the solver to stop.

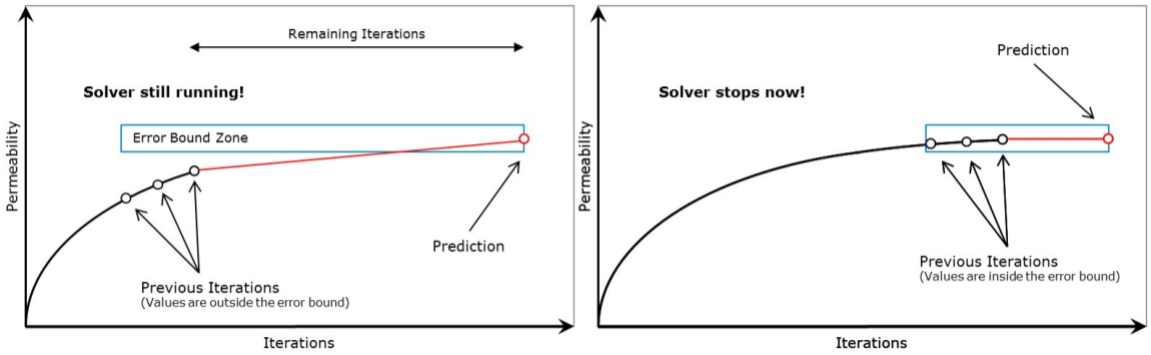
The stopping criterion that has been reached can be viewed in the Result Viewer of the GeoDict result file (*.gdr) under the **Results Report** tab.



Error Bound (LIR)

The stopping criterion **Error Bound** uses the result of previous iterations, and predicts the final solution based on linear and quadratic extrapolation. The solver stops if the relative difference between computed and predicted solution is smaller than the specified error bound. The stopping criterion recognizes oscillations in the

convergence behavior and prevents premature stopping at local minima or maxima. A damped convergence curve is fit through the oscillating curve and the solver stops then regarding the damped convergence curve.



If the Krylov method (under LIR - Advanced Options) is activated, the definition of **Error Bound** is somewhat different. Here, no prediction is made, instead the continuity between neighboring cells and the conservation of mass is checked. The maximal value is normalized by the mean flow and if this value is smaller than the **Error Bound** the simulation stops.

Tolerance (LIR and EJ)

The **Tolerance** stopping criterion looks for stagnation of the method when the process becomes stationary, i.e., the improvement in the permeability value becomes extremely small from iteration to iteration.

In each iteration, the solver checks for the current computed value x_c against the values x_{c-i} of the last 100 iterations if there are any changes.

$$\max_{i \in [1,100]} \frac{x_c - x_{c-i}}{x_{c-i}} < \text{tolerance} \tag{42}$$

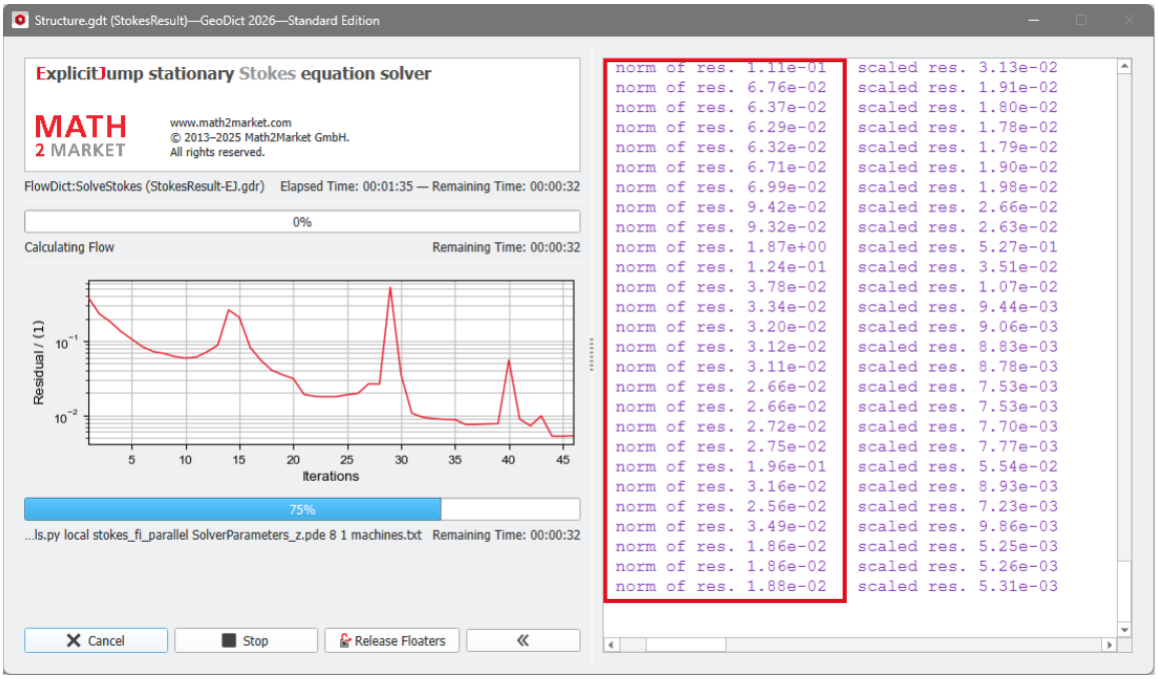
The computation is stopped if the maximal relative change is smaller than the value entered for **Tolerance**. When there is doubt about the quality of the solution, decrease the **Tolerance** value by a factor of ten. The drawback of this criterion is that the solver sometimes might stop too early in case of slow convergence rate or at local extrema of oscillatory convergence curves.

Residual (EJ)

When the **Residual** stopping criterion is used, the iteration is stopped if the solution satisfies the equation up to the required accuracy.

By setting the stopping criterion to **Residual**, the computations terminate as soon as the relative norm drops below the selected residual threshold. The relative norm of the

Schur Complement residual is computed and displayed in the console window during the calculations. The console is visible by clicking the double arrow button on the lower right corner in the progress dialog.



The recommendation to choose **Tolerance** or **Residual** for the **EJ** solver is based on the structure's porosity. Both give similar results for highly porous structures. For dense structures, if using the Schur Complement **Residual**, the relative norm of the residual may be small even though the correct permeability has not been reached. So, when in doubt, use the **Tolerance** criteria – the default option.

Maximum Iterations and Maximum Run Time (LIR and EJ)

Use the **Maximum Iterations** value or the **Maximum Run Time (h)** stopping criteria causes the solver to stop if the maximal number of iterations and/or the maximal runtime (in hours) is exceeded.

When the solver stops because one of these criteria has been reached, no guarantee on the quality of solution can be given. In this case, a warning is printed into the report. The following possibilities might help:

- Check the corresponding .log file to see how large the residual values and the permeability values are. If the permeability values are in reasonable range and they do not have changed during the last iterations, you may decide to use the current result.
- Double-check the structure and parameter values. Unphysical parameters or too rough resolution of the structure (leading, e.g., to artificial unconnected components) can cause an iterative solver to fail.

Know how! We recommend to use the **Error Bound** criterion for the **LIR** solver. This stopping criterion approximates the "relative deviation from the final solution".

Restart Save Interval

The calculations run by the solvers can be restarted from saved intermediate result files and the interval between auto-saves can be configured from the value entered in **Restart Save Interval (h)**.

Parallelization

The simulation process can be parallelized to speed-up the computation process. It depends on your license, how many parallel processes you may use. The solver benchmarks illustrate the speed-up that you can achieve.

The **Parallelization Options** dialog box opens when clicking the **Edit...** button, to choose between **Sequential**, **Parallel (Shared Memory)**, **Automatic Maximum of Threads**, and (if available) **Cluster**.

For details on how to set up and run parallel computations, consult the High Performance Computing chapter.

Restart from a *.gdr File

In some situations, it may be useful to re-use previously computed results and save a great amount of time by restarting the process from an existing GeoDict result file (*.gdr). Typical examples would be

- non-sufficient accuracy of some computation, when it is suspected that more iterations may improve the quality of the result,
- trying another boundary condition (e.g., Dirichlet instead of Periodic or other way around),
- continuing an iteration process,
- or using similar parameter values for the computation.

To restart and initialize the solver from a result file (*.gdr), check **Restart from .gdr File** and enter the name of a file, or **Browse** for it in the project folder.

When using **Restart From .gdr File** only the volume field (*.vap file) associated to the loaded *.gdr file will be used to continue the simulation.

Important! The *.vap file is saved in the result folder of a previous run. Make sure that this file was not deleted from the results folder. If the *.vap file is not available the

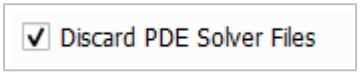
restart option cannot be used.

No other parameters from the options dialog will be loaded from the *.gdr file, but it is possible to load them by clicking **Load Parameters** in the Result Viewer. This way the simulation can be continued while using the same settings as in the initial simulation.

Important! The structure from the run of which you want to use the result file must be loaded into GeoDict. If this is not the case, an error message is displayed.

Discard PDE Solver Files

Checking **Discard PDE Solver Files** causes the deletion of all intermediate computation files, such as log files and flow field files.



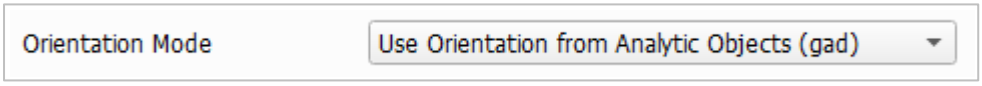
Only the content of the final GeoDict result file (*.gdr) is stored.

While having the benefit of saving hard disk storage place, discarding intermediate solver files has also the effect of disabling the 3D visualization of the results.

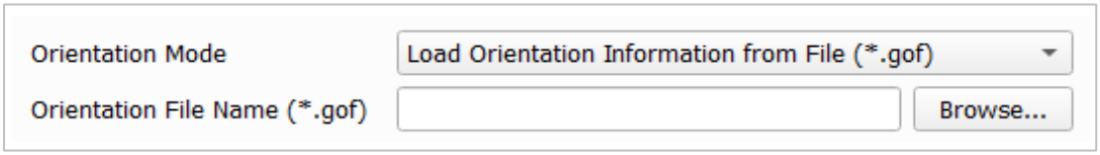
Orientation Mode (for LIR)

If one of the constituent materials has a non-isotropic material law, a local orientation is needed to compute the permeability. There are three different choices how to determine the local orientation.

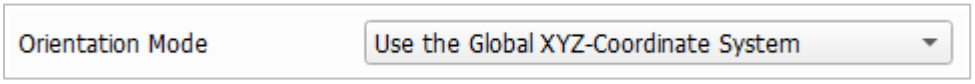
The standard case is to use the orientation defined by the local orientation of the GAD objects, e.g., the direction of a fiber (**Use Orientation from Analytic Objects (gad)**).



However, if the current structure was not generated using one of the structure generation modules, but imported from a 3D scan, GAD object information is not available. In such a case, the local orientation must be estimated from the image first, e.g., by using FiberFind or GrainFind. It is then possible to load the local orientation from a file generated by one of those modules, by selecting **Load Orientation Information from File (*.gof)**.

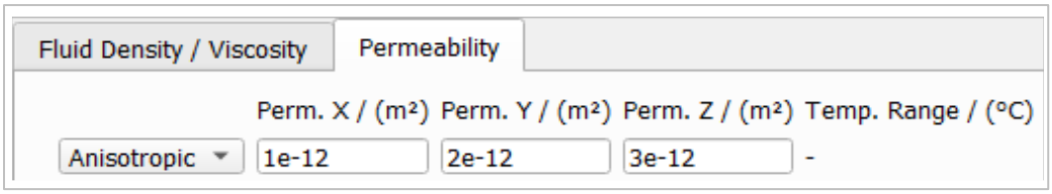


The last option is to simply **Use the Global XYZ-Coordinate System**.



In this case, the permeabilities entered in the Constituent Materials tab are the permeabilities

in the X-, Y-, and Z-directions.



Advanced Options

Unfold **Advanced Options** to gain access to several advanced solver options.

Advanced Options for EJ:

Analyze Geometry

Checking **Analyze Geometry** performs an analysis of the geometry before the solver computations.

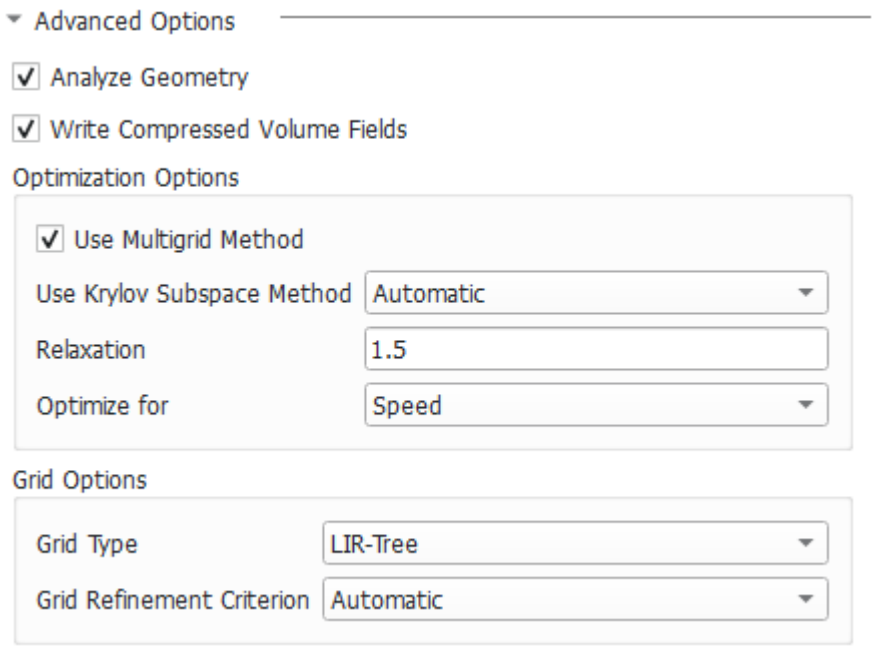
If no percolation path through the structure is found, the partial differential equation does not need to be solved, and a permeability of 0.0 is directly reported as the solution.

Furthermore, the solver operates on the whole structure, regardless of which parts are connected or unconnected. However, unconnected components are not transport relevant, thus the effort of solving in these parts is not necessary.

For these reasons, a geometrical analysis is routinely run to determine whether a through path exists. After the analysis is finished, unconnected components are removed from the computational grid.

This may speed up the computations but requires some time for the geometrical analysis, especially for very large structures. So if you know beforehand that there are no unrelevant parts or that the structure has been processed to eliminate them, the geometry analysis can be switched off by unchecking **Analyze Geometry**.

Advanced Options for LIR:



Analyze Geometry

Checking **Analyze Geometry** performs an analysis of the geometry before the solver computations.

If no percolation path through the structure is found, the partial differential equation does not need to be solved, and a permeability of 0.0 is directly reported as the solution.

Furthermore, the solver operates on the whole structure, regardless of which parts are connected or unconnected. However, unconnected components are not transport relevant, thus the effort of solving in these parts is not necessary.

For these reasons, a geometrical analysis is routinely run to determine whether a through path exists. After the analysis is finished, unconnected components are removed from the computational grid.

This may speed up the computations but requires some time for the geometrical analysis, especially for very large structures. So if you know beforehand that there are no unrelevant parts or that the structure has been processed to eliminate them, the geometry analysis can be switched off by unchecking **Analyze Geometry**.

Write Compressed Volume Fields

The LIR solver uses a very memory efficient adaptive grid structure for the simulations.

If the option **Write Compressed Volume Fields** is checked, then the adaptive grid is

used as compression method for writing out *.vap files. This option allows to save 80-90% space on hard drive. The runtime for writing *.vap files is also reduced significantly. But the runtime for loading and uncompressing of compressed *.vap is increased by the amount of runtime that was saved for writing out compressed *.vap files.

If the option **Write Compressed Volume Fields** is not checked, then a usual regular grid is used for writing out *.vap files.

Optimization Options

The **Multigrid Method** (see, e.g., Wesseling, 2004) was introduced to speed-up the computation and reduce the runtime significantly. The main idea of **Multigrid** is the usage of multiple coarser adaptive grids to speed up convergence behavior but requires only little more memory.

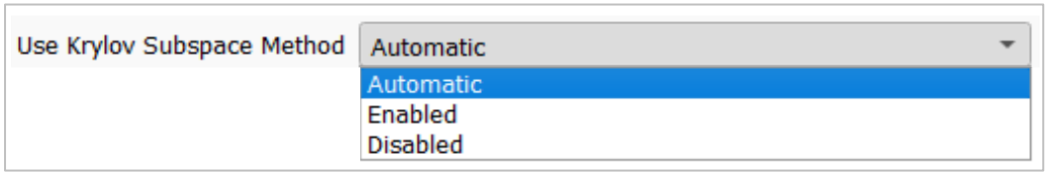
The method is available to solve the Stokes and Stokes–Brinkman equations as well as for solving mechanics, diffusion, thermal, and electrical conduction and is enabled by default.

Another speed-up option to accelerate the convergence behavior of the LIR solver is called **Krylov Subspace Method**.

The runtime of the LIR solver depends on many different properties of the structure and the simulation parameters. The BiCGStab algorithm is used, which can reduce the runtime for challenging simulation very drastically.

Note! Using the BiCGStab method approximately doubles the amount of RAM needed for the computation.

Unfortunately, the Krylov method is not always faster than a simulation without the Krylov method and therefore we introduced an **Automatic** mode which uses some heuristics to choose the most efficient method based on structure, material parameters, and boundary conditions automatically. Of course, it is possible to explicitly enable (**Enabled**) or disable (**Disabled**) the method.



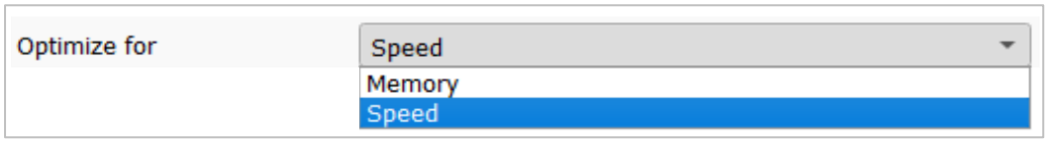
Note! In case that the Krylov subspace method (BICGStab) is used, the

Relaxation may also be adapted automatically.

Depending on the material parameters and geometry of the structure, the underlying mathematical problem can vary in complexity, thus influencing the behavior of the solver. The more complex the problem is, the more stable the solver settings should be.

With the **Relaxation** number, the solver is adjusted from **Stable** (which results in higher number of iterations, slower time stepping, and longer solver run times), to **Fast**, which makes the solver run less iterations but implies the risk that the solver does not converge. The **Relaxation** is a parameter of the SOR method and must be between 0 and 2 to ensure convergence. For relaxation values smaller than one (<1.0), the simulation is more stable. For relaxation values larger than one (>1.0), the simulation converges faster.

The **LIR** solver can **Optimize for Speed** or **Memory**.



- If **Speed** is chosen, the solver constructs additional optimization structures. The runtime is decreased by up to 30% but requires up to 50% more memory compared to the other option.
- If **Memory** is chosen, the runtime is increased by up to 40% but the solver requires up to 50% less memory.

Grid Options

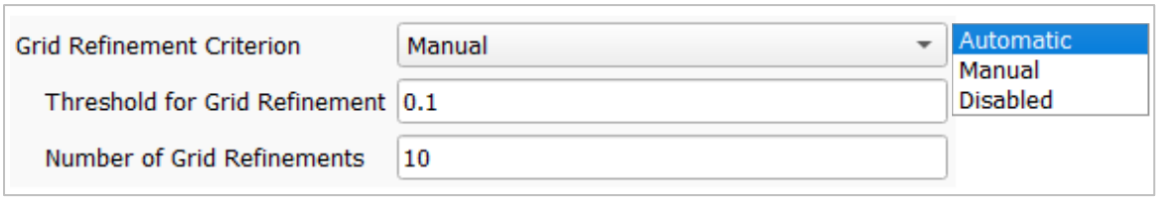
The **Grid Type** decides what kind of tree structure is used for the simulation.

The default option is **LIR-Tree** and should always be used. The solver uses an adaptive grid structure called LIR-tree and needs up to 10 times less runtime and memory compared to the **Regular Grid** option.

The solver can analyze the result field during the computation and improves the adaptive grid in places where more accuracy is needed. The LIR solver splits cells where a high gradient occurs.

The solver refines the adaptive grid based on the computed fields. New cells are introduced at locations where higher accuracy is needed. Three settings are available

here:



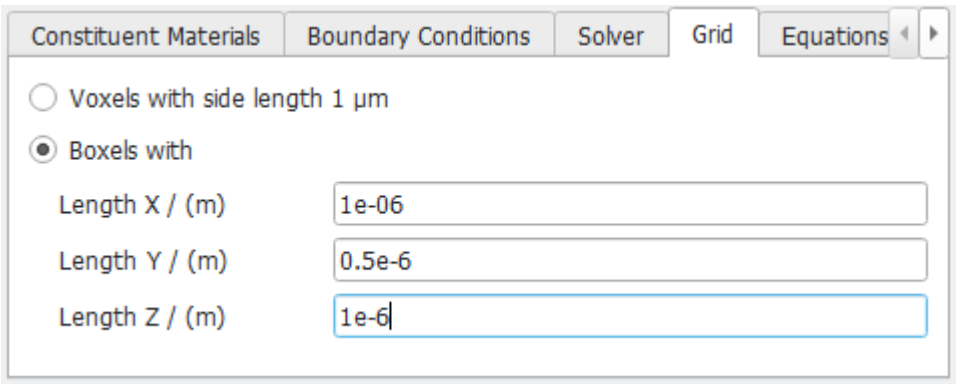
- **Automatic**: The computational cells are split when the difference in values between neighboring cells exceeds an automatically determined threshold. The solver automatically chooses all internal parameters based on the structure and simulation settings.
- **Manual**: The computational cells are split when the difference in values between neighboring cells exceeds the specified threshold. With this option you can specify the parameters **Threshold for Grid Refinement** and the **Number of Grid Refinements** manually. The **Threshold for Grid Refinement** must be between 0.0 and 1.0, while the recommended value range is between 0.05 and 0.1. The solver refines the grid at regions with high gradients and thus, the accuracy of the grid is enhanced in areas where it is necessary. Cells are split where the current gradient exceeds the **Threshold** multiplied by the maximum gradient. The **Number of Grid Refinements** is the maximum number of grid refinements that the solver can perform and should be between 1 and 10. The gradient-based grid refinement can increase the number of iterations, runtime, and memory requirements.
- **Disabled**: The grid refinement is turned off.

Know how! Grid Refinement can be neglected in most of the cases but should be used if a flow simulation is done on a structure with a very long inlet and outlet, for pleated filter structures, or for Navier-Stokes simulations.

Grid

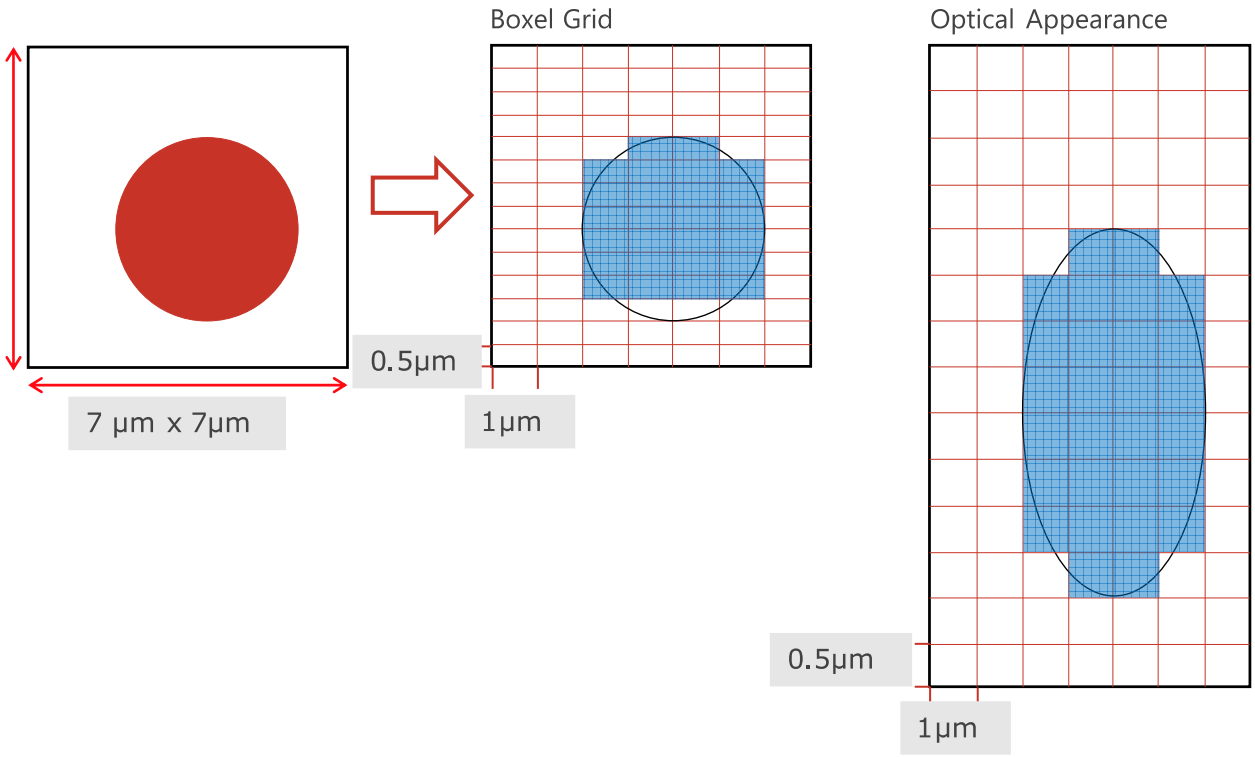
For the EJ13 solver, the computational grid can be adjusted under the **Grid** tab if necessary.

If **Boxels** (cuboid) instead of **Voxels** (cube) are used to resolve the structure, the boxel lengths can be set in the **Grid** tab.



Sometimes, grid lengths may be different in each direction, e.g., for Focused Ion Beam (FIB) images, the grid size in Z-direction is often different from the size in X- and Y-directions.

For example, a structure is sampled in a computational grid with different lengths in X- and Y-direction (e.g., LX = 2*LY). When the structure is loaded in GeoDict, each grid cell gets assigned 1 voxel and the visual representation of the structure looks stretched. To correct this effect, boxels set as, e.g., LX = 1 µm and LY = 0.5 µm, need to be chosen in the **Grid Size** panel.



1.6.2 Results

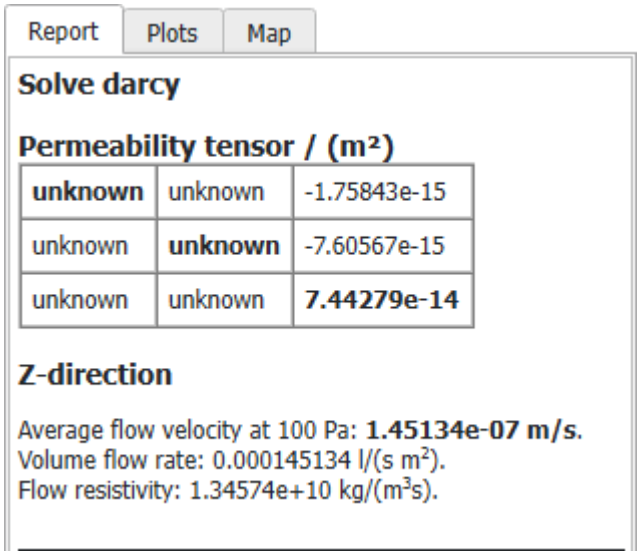
After the solver has finished, the result file (.gdr) is saved in the project folder and the **Result Viewer** automatically opens. Find detailed information on **Result Viewer** options in the Result Viewer user guide.

Results of the Darcy Flow command

- Report 140
  Shows the computed results.
- Plots 142
  View result plots.
- Flow Visualization 144
  Load results for a graphical visualization.

Report

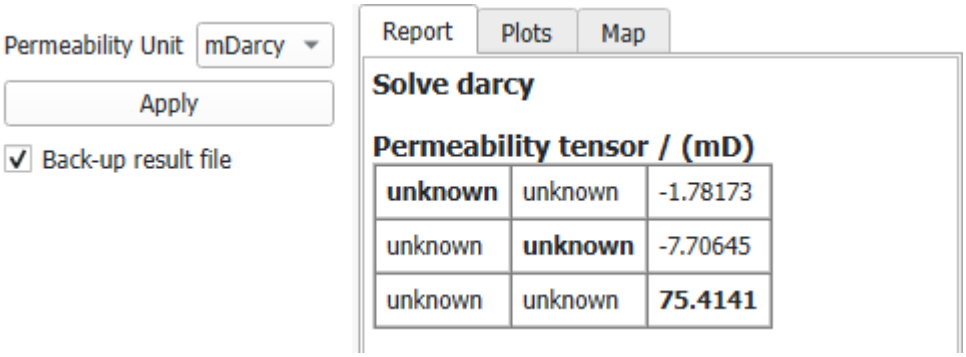
The **Report** tab under the **Results** tab shows the computed results.



It starts with the Permeability[20] (3x3) tensor. If not all space directions were selected as Computational Directions, the columns corresponding to the unselected directions are marked as **unknown**.

For each selected computational direction, the Average flow velocity[18] at a given pressure drop, the **Volume flow rate**, and the Flow resistivity[21] are reported.

In the post-processing panel on the left, you can change the unit of the permeability tensor.



Information about the convergence of the iterative solver is also displayed for each selected computation direction: **Threads**, **Iterations**, **Number of cells**, **Runtime**, **Memory usage**, and applied **Stopping criterion**.

Conduction solver data, Z-direction

Solver	EJ
Processes	12
Iterations	81
Runtime	7.641 s
Memory usage	100.959 MiB
Stopping criterion	tolerance

Total runtime: 00:00:08, Total memory usage: 1.209 GiB

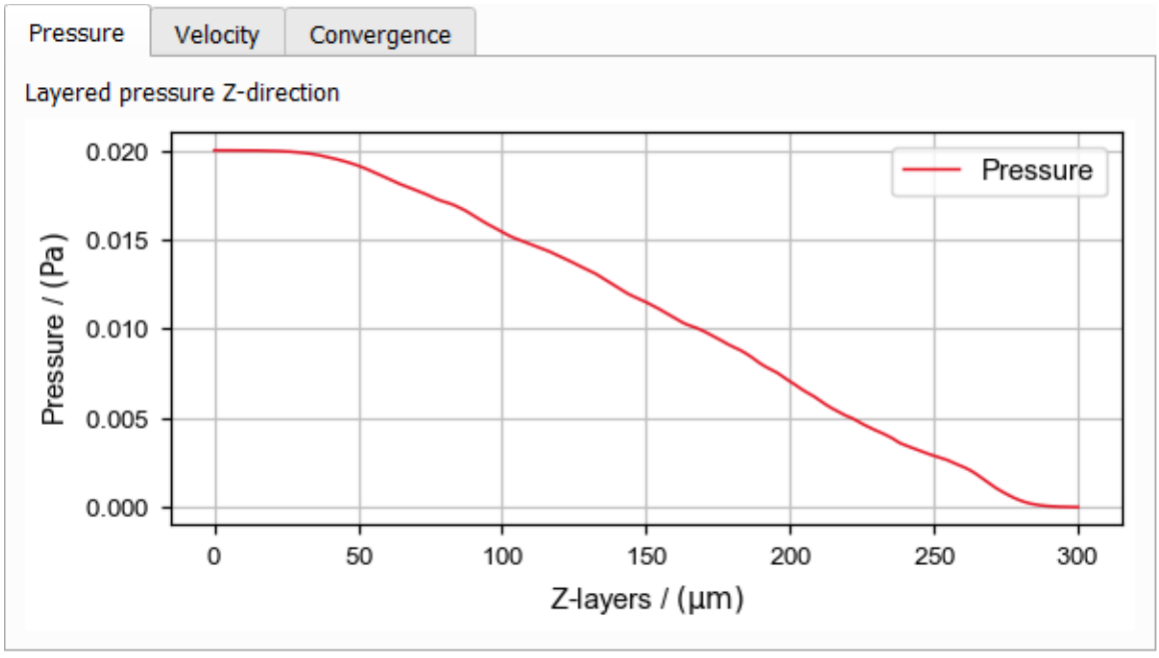
In the last line the **Total runtime** and **Total memory usage** are given. Here not only the resources used by the solver are taken into account but also writing volume fields and structure files or already loaded volume fields from previous simulations. Thus, even if only one flow direction is computed, the total runtime and total memory values are usually higher than the solver specific values.

Plots

Under the **Plots** tab various result plots are available:

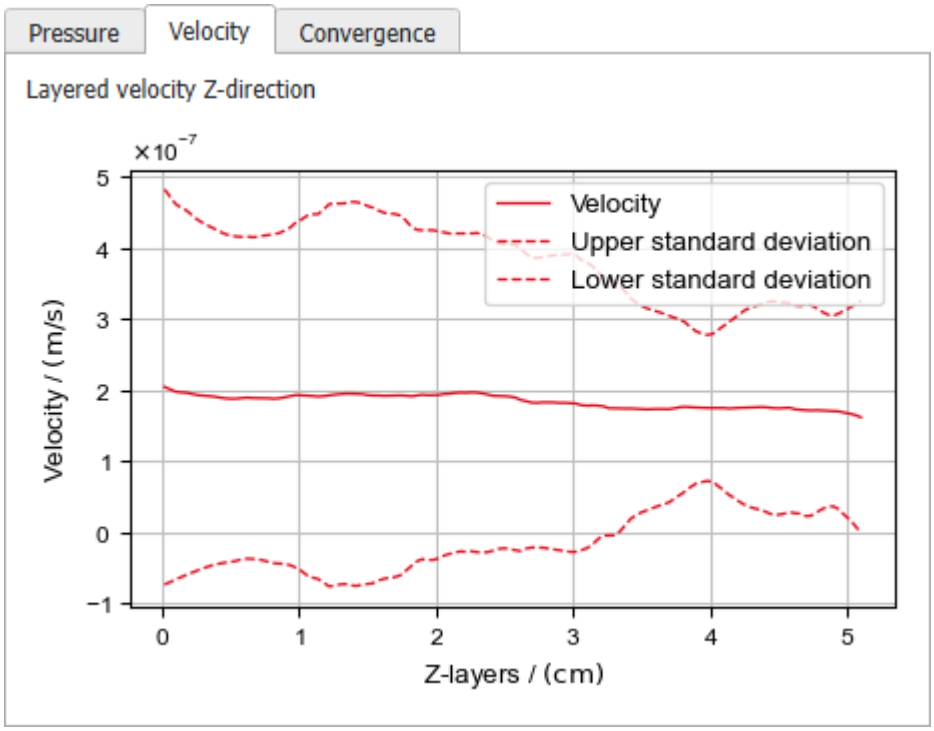
Pressure plot

Under the **Pressure** tab, a graph of the changes in **Pressure** across the layers is shown. There is one plot for each selected Computational Direction. Here, only the Z-direction was selected:



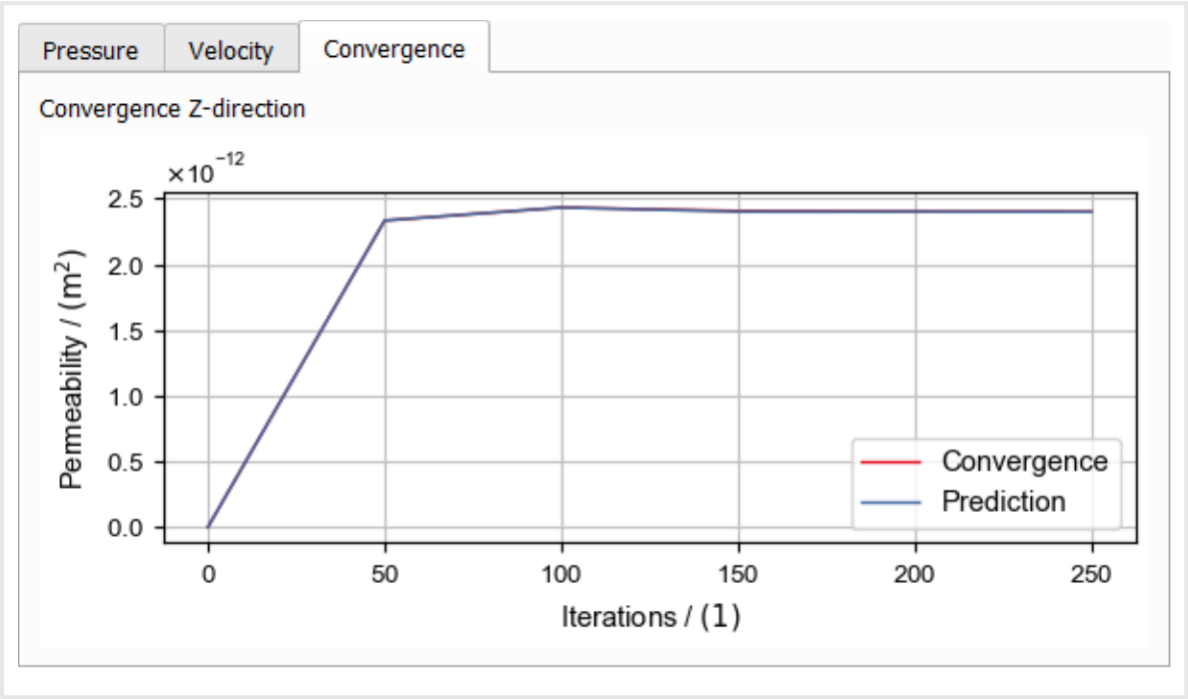
Velocity plot

Under the **Velocity** tab, a graph of the changes in **Velocity** across the layers is shown. Here, the velocity component in flow direction is shown. There is one plot for each selected Computational Direction. Here, only the Z-direction was selected: The velocity component in the chosen direction is averaged over the pore voxels for each layer in this direction.



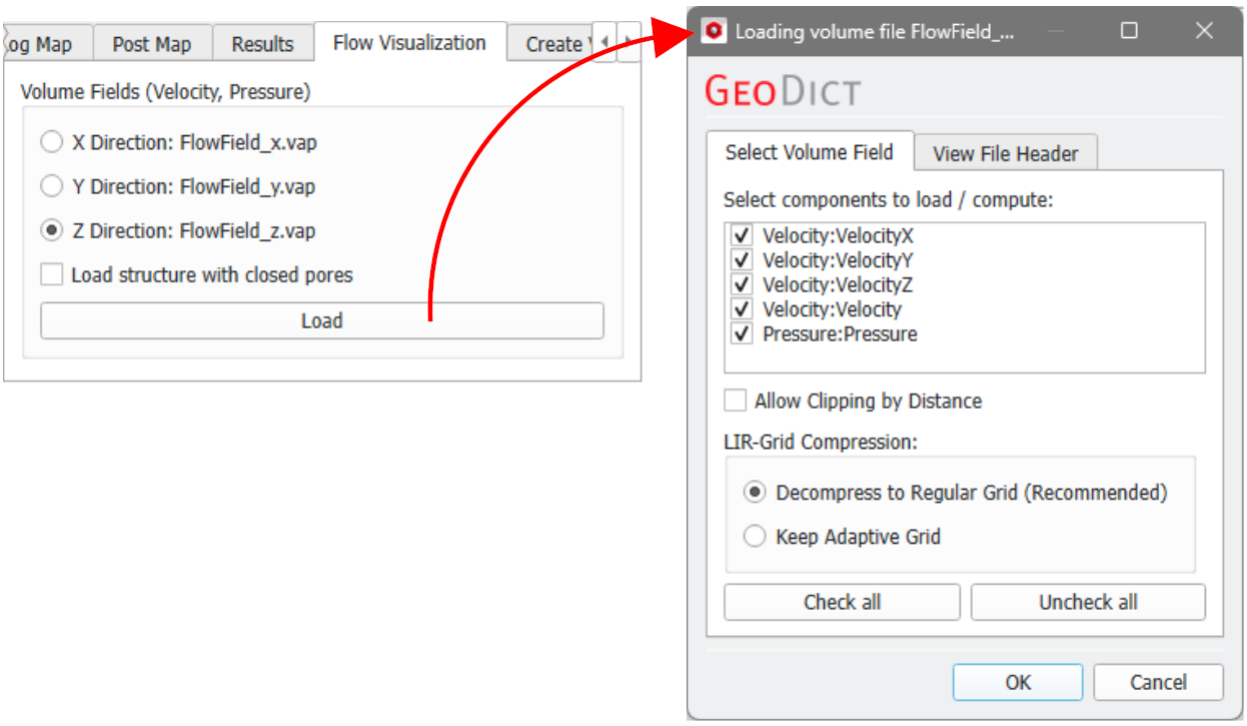
Convergence

The **Convergence** tab plots the computed permeability at each iteration. For the **Error Bound** criterion, it also shows the predicted value. In this case, the solver stops when the computed solution and the prediction are within the selected relative error.



Flow Visualization

Select the **Flow Visualization** tab to load the computed flow field (velocity and pressure, *.vap). Dependent on the selected **Computational Direction(s)**, result fields can be available for flow in **X-**, **Y-**, and/or **Z-Direction**.



Select the flow direction, then click **Load**. A dialog pops up, that let's you select which of the stored field components **VelocityX**, **VelocityY**, **VelocityZ**, **Velocity**, and/or **Pressure** you want to load and visualize.

Check **Allow Clipping by Distance** if you want to use this feature in the visualization. If selected, an additional data field is computed and loaded that contains the Euclidean distance to the solid structure. You can use this data field, e.g., to print arrows only with a certain distance from the solids. If you do not need this feature, do not select it, as it requires additional RAM to store the distance field.

Result files computed with the LIR solver are stored using the adaptive grid of the LIR solver (**LIR-Grid Compression**). Select if you want to interpolate the data during import to the regular voxel grid or if you want to keep the adaptive grid:

- **Decompress to Regular Grid:** More RAM is needed to load the data, and you cannot visualize the adaptive grid. This option enables the visualization with Streamlines, Arrows, and Schlieren.
- **Keep Adaptive Grid**: Less RAM is needed to load the data. You can visualize the adaptive grid. You cannot use Streamlines, Arrows, or Schlieren to visualize the results.

The structure that the solver used internally is loaded if **Load structure with closed pores** is checked. This structure shows which pores have been closed due to prior geometry analysis.

Visualization Options

- Volume Field: Pressure drop or velocity are visualized by a 3D color field - you can clip by threshold values or use transparency to view inside.
- Streamlines: Visualize the flow through the structure with streamlines or animated streamlets.
- Arrows: Visualize the velocity vector classically with arrows.
- Schlieren: This option uses Line Integral Convolution (LIC) fields to visualize vortices.

1.7 Single-Phase Flow GeoApps

Select **Single-Phase Flow GeoApps** to access pre-defined macros included in GeoDict:

Apps included in FlowDict

- Air Permeability Test 146
  This GeoApp performs an air permeability test simulation following norms like the ASTM D737 - 04. For fast flows, the Forchheimer approximation is used.

They can be edited with a standard text editor. To open the corresponding *.py file for a predefined macro, click the **Open Macro / Script File** button. After making changes to the script, you can either store the modified macro locally and run it from the main menu **Macro** → **Execute Macro Script** or as an administrator you may store it in its original place and make the modified version available to all users. Refer to the GeoApp User Guide.

To use a predefined macro, select it from the pull-down menu, click the **Edit...** button to set the parameters, and then the **Run** button to run the simulation.

1.7.1 Air Permeability Test

Air Permeability is an important factor in the performance of textile materials, e.g., gas filters or fabrics for air bags.

Modules needed to run this GeoApp:

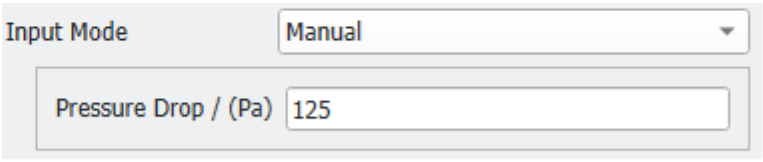
FlowDict

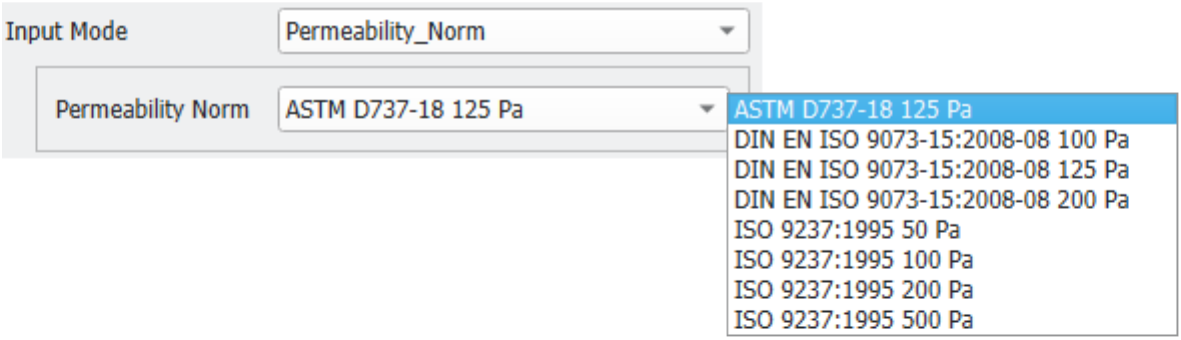
Know how! The *air permeability* is the air volume that passes through the media per time and surface area. It therefore has the unit $cm^3/min/cm^2$ or $ft^3/min/ft^2$ and thus describes an average flow velocity through the media at a standardized pressure drop. It is not to be confused with the material property *permeability* that is used in Darcy's law which has the unit m^2 or Darcy.

Select **Air Permeability Test** from the pull-down menu and click the **Option**'s **Edit...** button to open a dialog to enter or edit the simulation options. Choose a **Result File Name** for the result file (*.gdr) fitting to your current project.

Input Mode

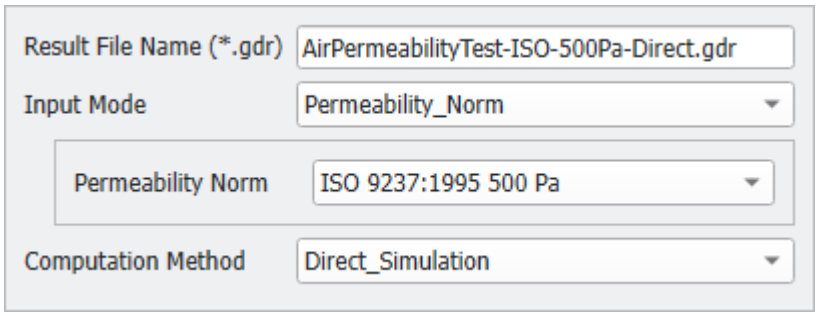
Two different **Input Mode** are available to prescribe the pressure drop for the simulation. Either enter a pressure drop manually or select one of the norms that define a pressure drop for the measurement. The pressure drop is applied in Z-direction on the current 3D structure.



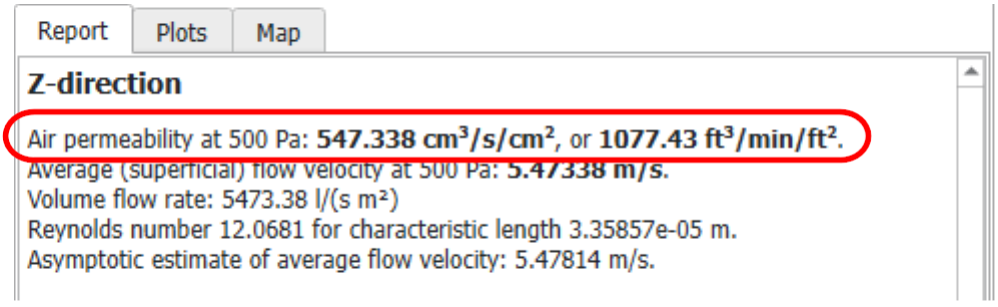


Computation Method: Direct Simulation

If you select **Direct Simulation** as **Computation Method**, the Navier-Stokes command 58 is used to simulate the air flow at the prescribed pressure drop. The boundary conditions and all solver parameters are automatically selected by the app.



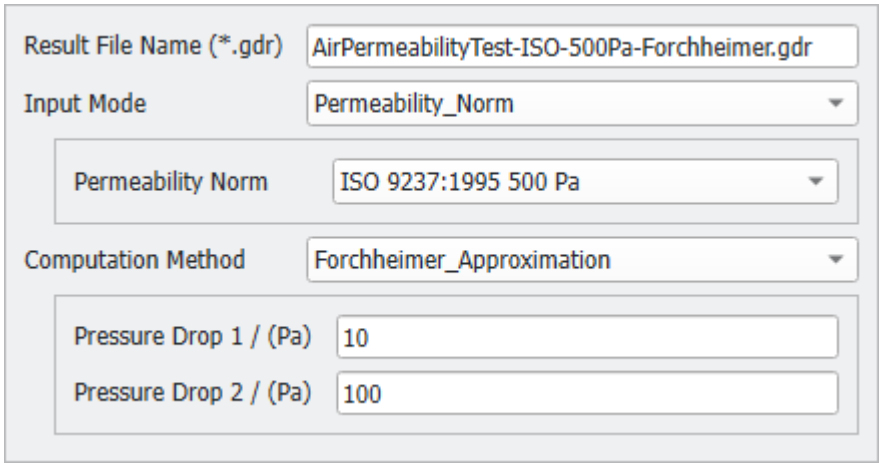
You can find the computed air permeability in the first line of the report.



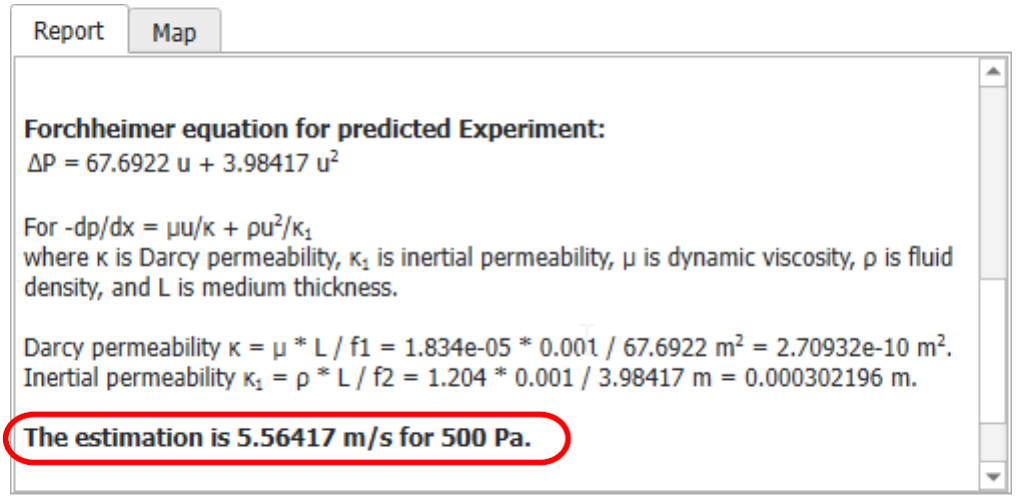
Computation Method: Forchheimer Approximation

If the media has a low flow resistivity, the air velocities may become very high at the prescribed pressure drop. In some cases, the flow might become turbulent, and the **Direct Simulation** might not converge. In these situations, select **Forchheimer Approximation** as **Computation Method**.

Select two lower pressure drops (**Pressure Drop 1** and **Pressure Drop 2**), which will be used for the extrapolation. The first value can be very small, but the second can be selected as high as possible (but low enough that convergence of the solver is still achieved).



The app uses the Navier-Stokes command 58 to simulate the air flow at the prescribed pressure drops. The boundary conditions and all solver parameters are automatically selected by the app. Afterwards, the app uses the Forchheimer Approximation command 93 to estimate the air permeability. You can find the result at the bottom of the report. Note, that the air permeability is reported as velocity in m/s here.



1.8 References

F.H. Harlow and J.E. Welch, *Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with Free Surface*, Physics of Fluids 8, 2182–2189 (1965)
S. Linden, A. Wiegmann and H. Hagen, *The LIR space partitioning system applied to the Stokes equations*, Graphical Models 82, 58-66 (2015), https://doi.org/10.1016/j.gmod.2015.06.003
F. Lomeland, E. Ebeltoft and W.H. Thomas, *A new versatile relative permeability correlation*, Presented at the International Symposium of the Society of Core Analysts held in Toronto, Canada (2005)
K. Mattila, T. Puurtinen, J. Hyväluoma, R. Surmas, M. Myllys, T. Turpeinen, F. Robertsén, J Westerholm and J. Timonen, *A prospect for computing in porous material research: very large fluid flow simulations*, Journal of Computational Science 15, 62-76 (2016)
S. Partankar, *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing Corporation (1980)
N. Saxena, R. Hofmann, F.O. Alpak, S. Berg, J. Dietderich, U. Agarwal, K. Tandon, S. Hunter, J. Freeman and O.B. Wilson, *References and benchmarks for pore-scale flow simulated using micro-CT images of porous media and digital rocks*, Advances in Water Resources 109, 211-235 (2017)
D.J. Tritton, *Physical fluid dynamics (2nd ed.)*, Oxford: Claredon Press (2006), ISBN 978-0198544937
H.K. Versteeg and W. Malalasekera, *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*, Pearson, Prentice Hall, 2nd Edition (2007)
A. Wiegmann, O. Iliev and A. Schindelin, *Computer Aided Engineering of Filter Materials and Pleated Filters*, Global Guide of the Filtration and Separation Industry by E. von der Luehe, VDL – Verlag, 191-198 (2010)
A. Wiegmann, *Computation of the permeability of porous materials from their microstructure by FFF-Stokes*, Report of the Fraunhofer Institute for Industrial Mathematics (ITWM) Nr. 129 (2007)
Z. Zeng and R.A. Grigg, *A Criterion for Non-Darcy Flow in Porous Media*, Transport Porous Media 63(1), 57–69A (2006), http://dx.doi.org/10.1007/s11242-005-2720-3

Citation: Math2Market GmbH, 2026: GeoDict 2026 User Guide, FlowDict, Math2Market GmbH, Germany, doi.org/10.30423/userguide.geodict

MATH 2 MARKET

Math2Market GmbH
Richard-Wagner-Str. 1, 67655 Kaiserslautern / Germany
www.math2market.com