

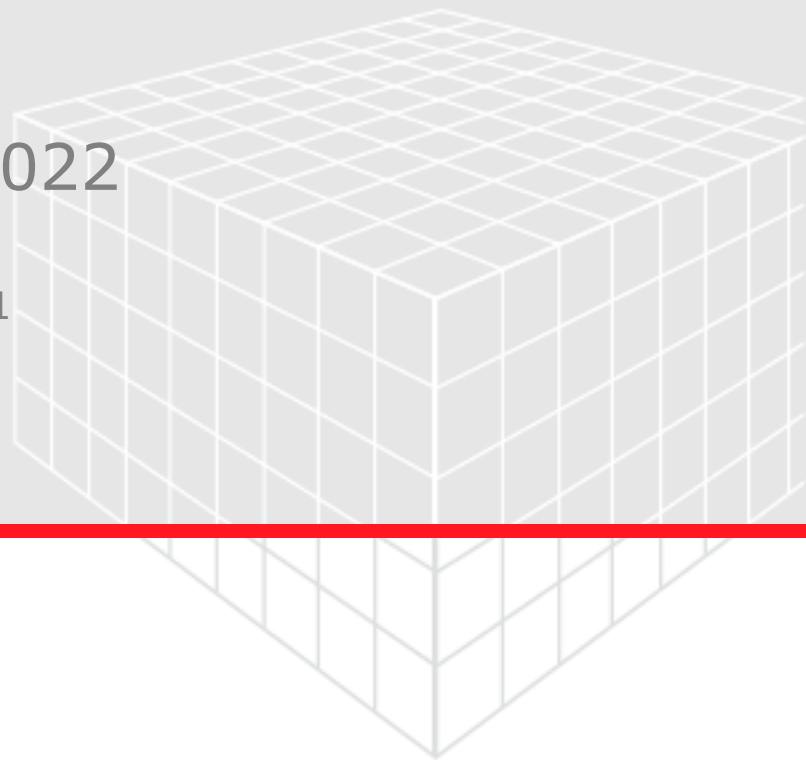
FLOWDICT

User Guide

GeoDict release 2022

Published: November 9, 2021

Reviewed: December 13, 2021



GEO DICT

COMPUTING FLUID FLOW PROPERTIES AND PERMEABILITY	1
PARTIAL DIFFERENTIAL EQUATIONS (PDEs) FOR FLOW MODELS	3
FLOW SOLVERS IN FLOWDICT	5
FLOWDICT POST-PROCESSING AND OUTPUT	6
Mean Flow Velocity / Pressure Drop	6
Flow Permeability	6
Flow Resistivity	7
Gurley Value	7
FLOWDICT COMPUTATIONS	9
FLOWDICT SECTION	10
Solver Options	12
STOKES(-BRINKMAN) AND NAVIER-STOKES(-BRINKMAN) - SOLVER OPTIONS	14
CONSTITUENT MATERIALS	14
BOUNDARY CONDITIONS	17
SOLVER	24
GRID	36
EQUATIONS & REFERENCES	36
HINTS TO ESTIMATE MEMORY REQUIREMENTS FOR FLOW SOLVERS	37
MEMORY ESTIMATION FOR EJ SOLVER	37
MEMORY ESTIMATION FOR SIMPLEFFT	37
MEMORY ESTIMATION FOR LIR SOLVER	38
FORCHHEIMER APPROXIMATION OPTIONS	39
MEASURED EXPERIMENT	40
PREDICTED EXPERIMENT	41
FLOWDICT SIMULATION RESULTS	42
Results for Stokes(-Brinkman) and Navier-Stokes(-Brinkman)	42
RESULTS - REPORT	42
CHARACTERISTIC LENGTH FOR REYNOLDS NUMBER	44
RESULTS - PLOTS	45
FLOW VISUALIZATION	47
CREATE VIDEOS	49
Result folder from EJ, SimpleFFT, and LIR solver	50
Result file (.gdr) from Forchheimer Approximation	51
PREDEFINED	54
Air Permeability Test (Direct)	55
Air Permeability Test (Forchheimer)	56
REFERENCES	58
APPENDIX:	59
THE DISCRETIZATION FOR CONVECTION-DIFFUSION PROBLEMS	59
The central differencing scheme	60
The upwind differencing scheme	60
The power-law scheme	61

COMPUTING FLUID FLOW PROPERTIES AND PERMEABILITY

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

Two categories of experiments can be performed in **FlowDict**: prediction of mean flow velocity for a given pressure drop and prediction of pressure drop for a given mean flow velocity [1]. In **FlowDict**'s post-processing step, the relationship between the predicted mean flow velocity (or pressure drop), and the fluid viscosity and media thickness expressed in [Darcy's law](#) is used to compute and output the material permeability.

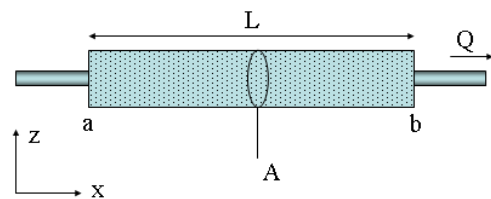
$$\vec{u} = -\frac{K}{\mu} (\nabla p - \vec{f}) \quad (\text{Darcy's law})$$

where \vec{u} is the fluid flow velocity, K is the permeability, μ is the fluid viscosity, p is the pressure, and f is a force density.

Since the viscosity of the fluid is written separately in our formulation of Darcy's law, permeability is a true material property. Instead of pulling the viscosity into the definition of permeability, the viscous resistivity is considered, i.e. the quotient of fluid viscosity and material permeability. Thus, if quantities such as water or air permeability are needed, consider the reciprocal of the viscous resistivity.

Defining permeability as a material property also means that the pressure drop cannot be so high as to result in deformation of the media.

$$Q = \frac{-k A}{\mu} \cdot \frac{(P_b - P_a)}{L}$$



where Q is total discharge (m^3/s or l/s), k is the intrinsic permeability of the medium (m^2), A is the cross-sectional area to flow (m^2), $(P_b - P_a)$ is the total pressure drop (Pascal), μ is the viscosity (Pascal \cdot s), and L is the length over which the pressure drop is taking place (m).

A flow experiment in **FlowDict** consists of three sets of input parameters:

- A three-dimensional representation of a structure or material (geometry model)
- An incompressible i.e. with constant density, Newtonian fluid (gas or liquid with constant viscosity)
- The experimental process parameters, such as mass flow rate (or flow velocity) and flow direction.

FlowDict computations can be used to determine air and water permeability in woven fabrics, to study gas and liquid permeability and pressure drop in filter media, to predict gas permeability for the extraction of gas in reservoir and source rock, and flow characteristics of [hydrocarbons](#) in [oil](#) and [gas](#) reservoirs, and flow properties of groundwater in aquifers.

PARTIAL DIFFERENTIAL EQUATIONS (PDEs) FOR FLOW MODELS

When using Darcy's law, the user should be aware of the following:

1. Darcy's law only applies to very slow (so-called creeping or **Stokes**) flows, with a [Reynolds number](#) close to zero.

$$-\mu \Delta \vec{u} + \nabla p = \vec{f} \quad (\text{Stokes conservation of momentum})$$

The Stokes equations, which are simplified from Navier-Stokes equations by dropping the inertial term, are used to describe the flow. In this regime, changing pressure drop or velocity by a factor, linearly changes the other by the same factor, so that Darcy's law always predicts the same value for the permeability.

2. The second caveat lies in the definition of length (L) in Darcy's law. This length is meant not to include the inlet and outlet (inflow and outflow regions) that some virtual flow experiments require. Under the assumptions of slow flow and void of inlet and outlet, choosing the Stokes flow option in **FlowDict** and using the computed permeability is valid.

For faster flows, one may not neglect the influence of **inertia**. The **Navier-Stokes** equations and the continuity equation are adopted to describe the conservation of momentum and the conservation of mass.

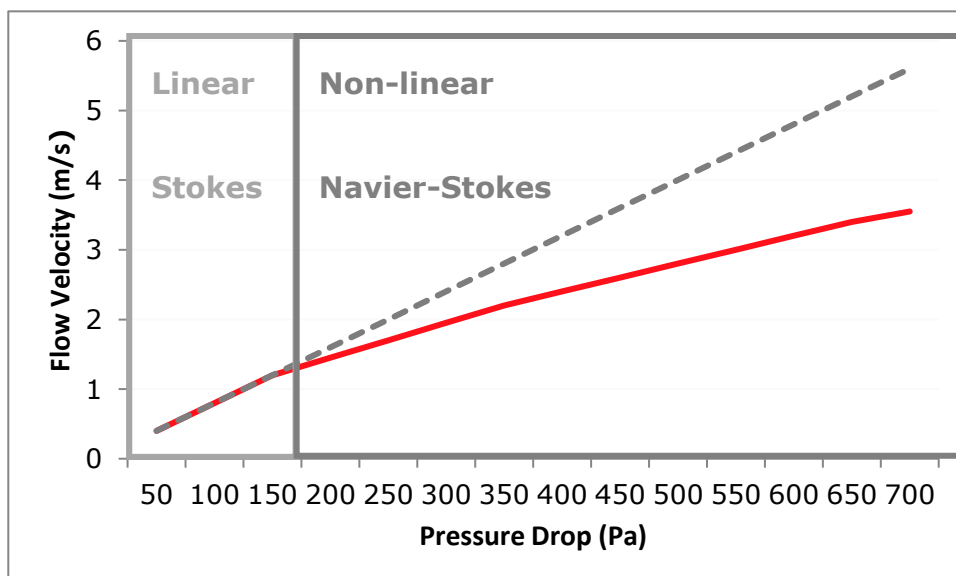
$$-\mu \Delta \vec{u} + (\rho \vec{u} \cdot \nabla) \vec{u} + \nabla p = \vec{f} \quad (\text{Navier-Stokes conservation of momentum})$$

$$\nabla \cdot \vec{u} = 0 \quad (\text{conservation of mass})$$

On the surfaces of solids, the no-slip boundary condition is used:

$$\vec{u} = 0 \quad (\text{no-slip boundary condition})$$

For such flows, the relation between pressure drop and mean velocity is not linear and does not provide the material permeability properties.



When both a fast-flow and a porous medium exist in a region, a third set of equations is needed, the Navier-Stokes-Brinkman equations

$$-\mu \Delta \vec{u} + (\rho \vec{u} \cdot \nabla) \vec{u} + \mu K^{-1} \vec{u} + \nabla p = \vec{f} \quad (\text{Navier-Stokes-Brinkman conservation of momentum})$$

where K^{-1} is the inverse of the permeability tensor and μK^{-1} is the flow resistivity.

The additional term accounts for porous media formed by sub-grid sized media. All experiments in FlowDict, even for Navier-Stokes, 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, and FlowDict users are advised to carefully examine the validity of FlowDict predictions for their own application area.

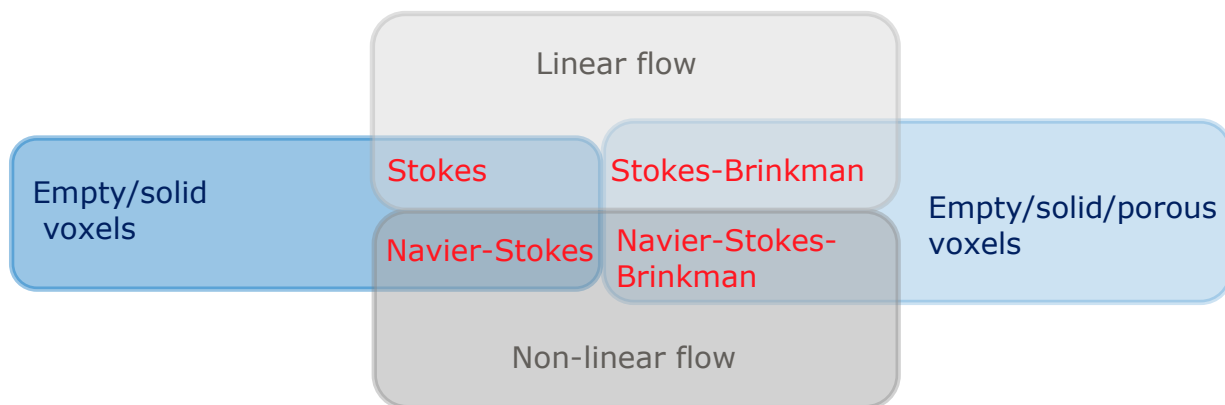


Table of possible conservation of momentum equations:

$-\mu \Delta \vec{u}$			$+\nabla p = \vec{f}$	(Stokes conservation of momentum)
$-\mu \Delta \vec{u}$		$+\mu K^{-1} \vec{u}$	$+\nabla p = \vec{f}$	(Stokes-Brinkman conservation of momentum)
$-\mu \Delta \vec{u}$	$+(\rho \vec{u} \cdot \nabla) \vec{u}$		$+\nabla p = \vec{f}$	(Navier-Stokes conservation of momentum)
$-\mu \Delta \vec{u}$	$+(\rho \vec{u} \cdot \nabla) \vec{u}$	$+\mu K^{-1} \vec{u}$	$+\nabla p = \vec{f}$	(Navier-Stokes-Brinkman conservation of momentum)

FLOW SOLVERS IN FLOWDICT

To solve the partial differential equations (PDEs) describing the fluid flow, three solution methods, called solvers, are available in FlowDict.

- 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 equation](#) only.

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.

- 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.

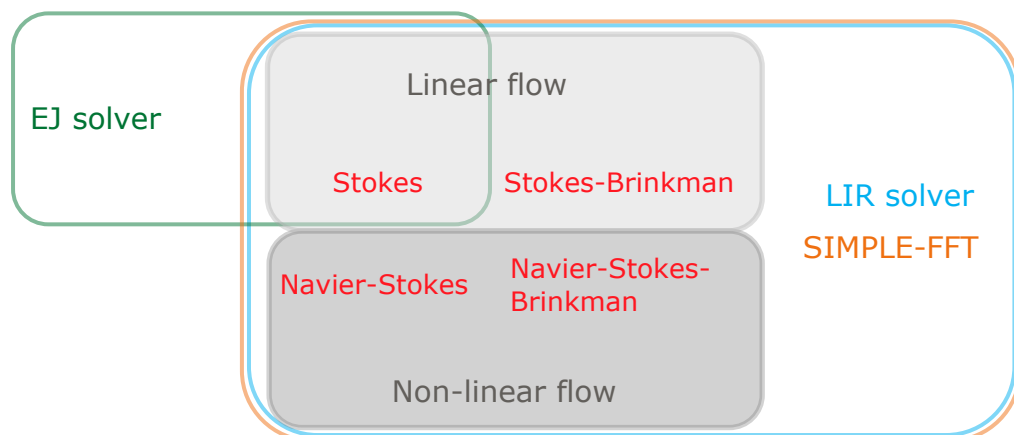
Therefore, faster flows, which are not necessarily laminar and are modeled by the [Navier-Stokes equation](#), can be computed with the **SimpleFFT** solver.

The solver can also deal with porous voxels and thus solves the Stokes-Brinkman and Navier-Stokes-Brinkman equations.

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

- 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.

It can be used to solve the Stokes, Stokes-Brinkman, Navier-Stokes, and Navier-Stokes-Brinkman equations.



FLOWDICT POST-PROCESSING AND OUTPUT

In a post-processing step, **FlowDict** interprets the results obtained by the chosen solution method (**EJ**, **SimpleFFT**, or **LIR**), using the specified mean flow velocity or pressure drop, fluid viscosity, and media thickness, and outputs the material permeability.

The outcome of the flow computations is shown in the **GeoDict** result file (*.gdr) in four ways:

MEAN FLOW VELOCITY / PRESSURE DROP

The flow velocity fields are calculated by the selected flow solver in the three directions by setting up the pressure drop and selecting the computation directions (Boundary Conditions). In **FlowDict**, the default value corresponds to the virtual pressure drop of 0.02 Pa (0.0002 mbar = $2 \cdot 10^{-7}$ bar) across the structure in the Z-direction.

The directions X and Y can also be selected, so that the flow is calculated in those directions. In this case, the result file shows the average flow velocities in the three directions calculated for the entered pressure drop. Non-zero values in X- and Y-directions mean that the calculation also gave some flux in the directions perpendicular to the direction of the applied pressure drop. This is due to directional anisotropies of the structures and to the lack of boundaries that could stop the flow in the perpendicular directions (in contrast to experiments where boundaries are usually present).

FLOW PERMEABILITY

For the **Flow Permeability**, the structure's permeability tensor is found from [Darcy's law](#):

$$\vec{v}^i = -\mu^{-1} \begin{bmatrix} K_{xx} & K_{xy} & K_{xz} \\ K_{yx} & K_{yy} & K_{yz} \\ K_{zx} & K_{zy} & K_{zz} \end{bmatrix} \nabla p^i$$

Here \vec{v}^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

$$\nabla p^i = \frac{\Delta p^i}{L^i} \quad \text{with } \Delta p^1 = \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix}, \Delta p^2 = \begin{bmatrix} 0 \\ 2 \\ 0 \end{bmatrix}, \Delta p^3 = \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix}$$

is the pressure gradient (or pressure difference) in the i^{th} direction.

$L^1 = NX * L_v$, $L^2 = NY * L_v$, and $L^3 = NZ * L_v$ 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**).

For the linear **Stokes EJ** (Explicit Jump), **Stokes SimpleFFT**, and **Stokes LIR** flow solvers, this description yields a permeability tensor K that is independent of the applied pressure drop, as well as from the used fluid viscosity. Thus, the permeability is considered a material property.

The **Navier-Stokes** model for the **SimpleFFT** and **LIR** flow solver is a non-linear method. Hence, \bar{v}^i is not proportional to ∇p^i anymore. In the non-linear case, the computed velocities are typically lower than in the linear case. In this case, the permeability tensor K depends on the pressure drop and the viscosity, and thus, it is NOT a material property.

FLOW RESISTIVITY

In real measurements of **Flow Resistivity**, the pressure drop is usually applied in the **Z**-direction. That is, for the default setting of 2 mbar we consider

$$\Delta p^3 = \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix}$$

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

Formally, the flow resistivity tensor is defined as μK^{-1} , where μ is the fluid viscosity.

Since the fluid flow is computed only in one direction, not all entries of the permeability tensor K are available. Thus, when the **Flow Resistivity** is selected in FlowDict, the flow resistivity in the Z-direction is approximated by

$$\frac{\mu}{K_{zz}}$$

If the other directions are also selected, the flow resistivity for them is approximated by dividing the viscosity by the corresponding diagonal entries of the permeability tensor. Thus, flow resistivity is NOT a material property, because it always depends on the fluid viscosity.

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:

$$\begin{aligned} \Delta p &= 0.176 \text{ psi} &= 1213.48 \text{ Pa} \\ A &= 1 \text{ in}^2 &= 0.00064516 \text{ m}^2 \\ V &= 100 \text{ cm}^3 &= 10^{-4} \text{ m}^3 \end{aligned}$$

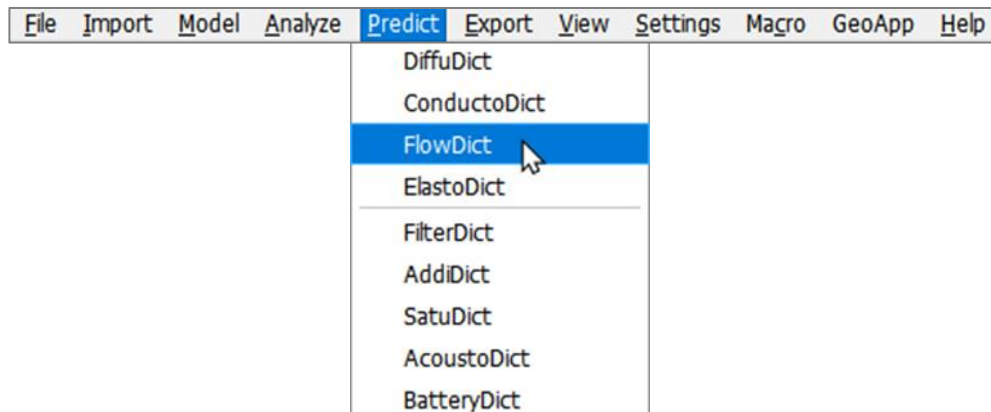
the Gurley value can be obtained from Darcy's law with:

$$\begin{aligned}
 G &= \frac{\mu V L}{\kappa \Delta p A} \\
 &= \frac{1.834 \times 10^{-5} \times 10^{-4} \times L}{\kappa \times 1213.48 \times 0.00064516} \\
 &= 2.3426 \times 10^{-9} \times \frac{L}{\kappa}
 \end{aligned}$$

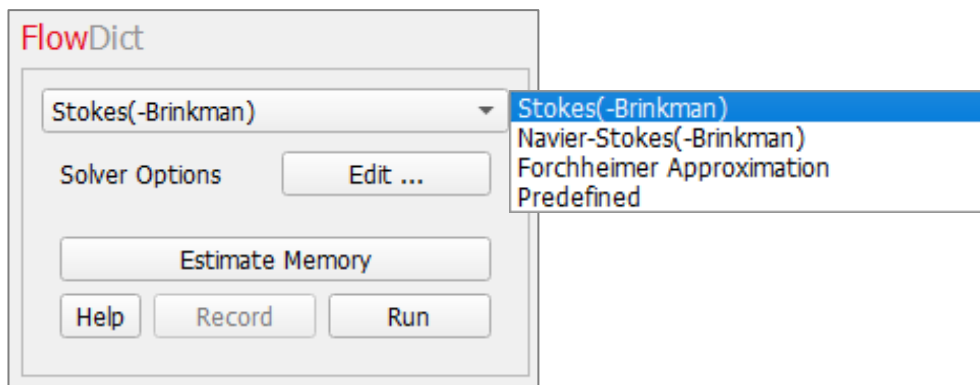
The Gurley value depends on the permeability and the physical length of the computational domain in the directions of interest.

FLOWDICT SECTION

Switch to the **FlowDict** module by selecting **Predict** → **FlowDict** in the **Menu bar**.



In the **FlowDict** section, the available equations to solve are listed in the pull-down menu. After choosing the appropriate equations for the material structure, the options for the selected equation need to be entered.



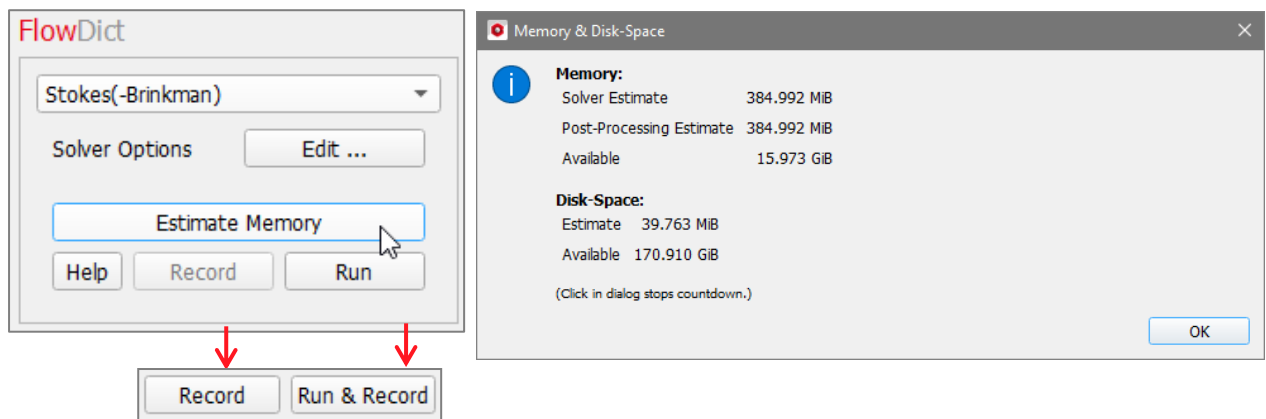
For the selection between the flow model equations to be solved, the conditions of the flow, the properties of the fluid, and the components of the structure must be taken into account. Find detailed information on PDEs on page [3](#).

The **Stokes(-Brinkman)** equation provides good results for fluids with high viscosity, such as oil, with slow, laminar flows defined by small Reynolds numbers but can lead to non-physical effects.

The **Navier-Stokes(-Brinkman)** equation is preferable for gas filtration processes (e.g., of air), because it allows for non-linear effects. The equation entails being solved with the **SimpleFFT** and **LIR** solver and, implicitly, slightly longer run times and higher memory requirements.

The **Forchheimer Approximation** has been introduced in **FlowDict** to approximate pressure drop or mean velocity for very high Reynolds number that cannot be computed by solving the **Navier-Stokes(-Brinkman)** equation with the available solvers. With at least two mean velocity / pressure drop pairs, it is possible to approximate other pairs with quadratic regression by least squares analysis.

When clicking **Estimate Memory & Disk-Space**, **FlowDict** can estimate the needed memory 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** message. A more detailed description about these estimations can be found in page [37](#).

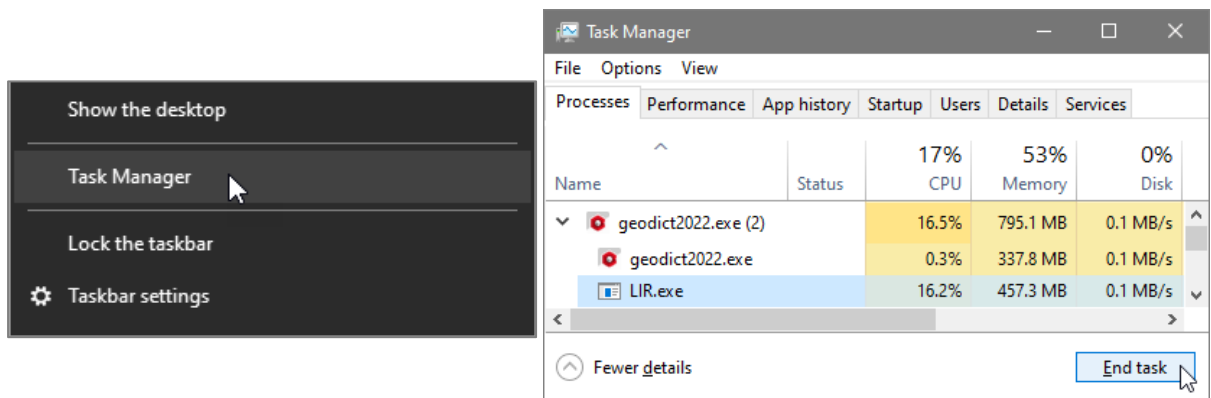


When the input parameters for the selected **FlowDict** flow solver have been entered, clicking **Run** in the **FlowDict** section starts the computations. When recording a macro, **Record** becomes active and **Run** changes to **Run & Record**.

The flow solver process can be stopped when clicking **Cancel** or **Stop** in the progress dialog. Depending on the solver's internal processes and actual memory usage, cancelling the computation may not be instantaneous.

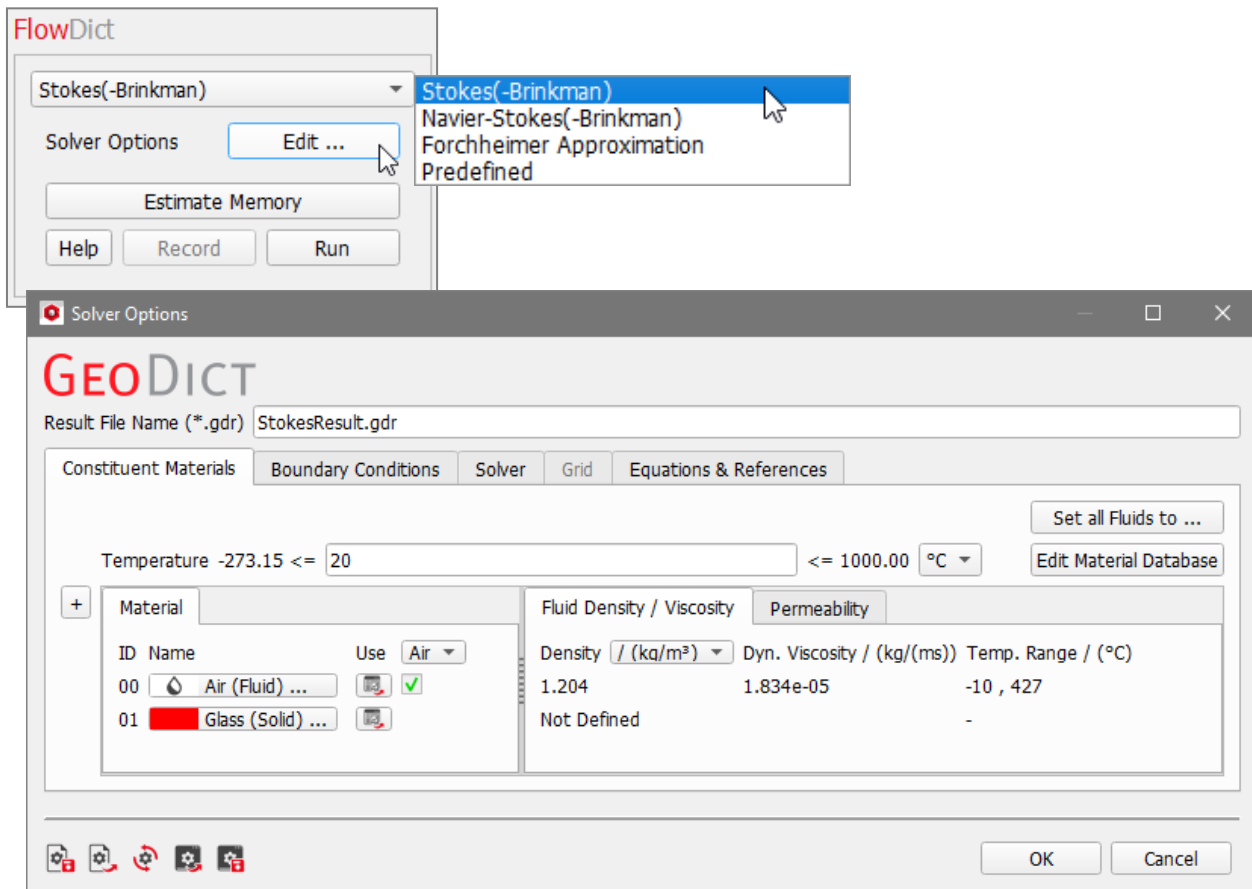
If the **Cancel** button is clicked, the calculation stops without reporting a meaningful result: no permeability, mean velocity, or pressure drop is reported, and no flow field is written on the hard drive. If the **Stop** button is pressed then the solver treats the current simulation as being converged. The solver reports a permeability, mean velocity or pressure drop and a flow field is written on the hard drive.

If the memory requirements are close to maximal computer capacities, you may need to use the *Task Manager* to stop the **FlowDict** process. Access the Windows Task Manager dialog by clicking the right mouse button on the Windows task bar.



SOLVER OPTIONS

Depending on the equation chosen, the corresponding **Solver Options** dialog opens when clicking the **Solver Options' Edit...** button in the **FlowDict** section.

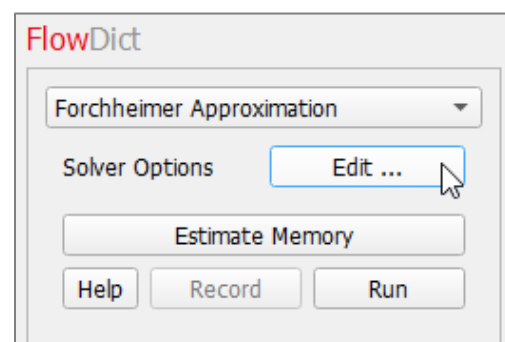


At the top of the dialogs for all equations, a customized **Result File Name (*.gdr)** should be entered to differentiate the results of sets of **FlowDict** computations. The resulting *.gdr file will be placed inside the chosen project folder. An additional directory with this name is created to keep the intermediate computation files (PDE solver files).

Clicking **OK** confirms the entered solver options and clicking **Cancel** closes the **Solver Options** dialogs without modifications.

The parameters settings in the **Solver Options** dialogs for the **Stokes(-Brinkman)** and the **Navier-Stokes (-Brinkman)** equations are almost identical and organized under the **Constituent Materials**, **Boundary Conditions**, **Solver**, **Grid**, and **Equations & References** tabs.

The **Forchheimer Approximation Options** dialog is completely different. There are no tabs, and all solver parameters are entered directly.



Forchheimer Approximation Options

GEODICT

Result File Name (*.gdr)

Measured Experiment

Thickness of measured media mm

Density of measured media kg/m³

Viscosity of measured media kg/ms

☒ Extrapolate from given Velocity/Pressure pairs

☐ Extrapolate from .gdr result files

Number of Parameters

	Velocity / (m/s)	PressureDrop / (Pa)
1		
2		

Predicted Experiment

Thickness of predicted media mm

Density of predicted media kg/m³

Viscosity of predicted media kg/ms

☒ Use Measured Experiment

☐ Air Permeability (ISO 9237)

☐ Air Permeability (ASTM D737)

☒ Pressure Drop Pa

☐ Mean Velocity m/s

☐ Flow Rate l/min

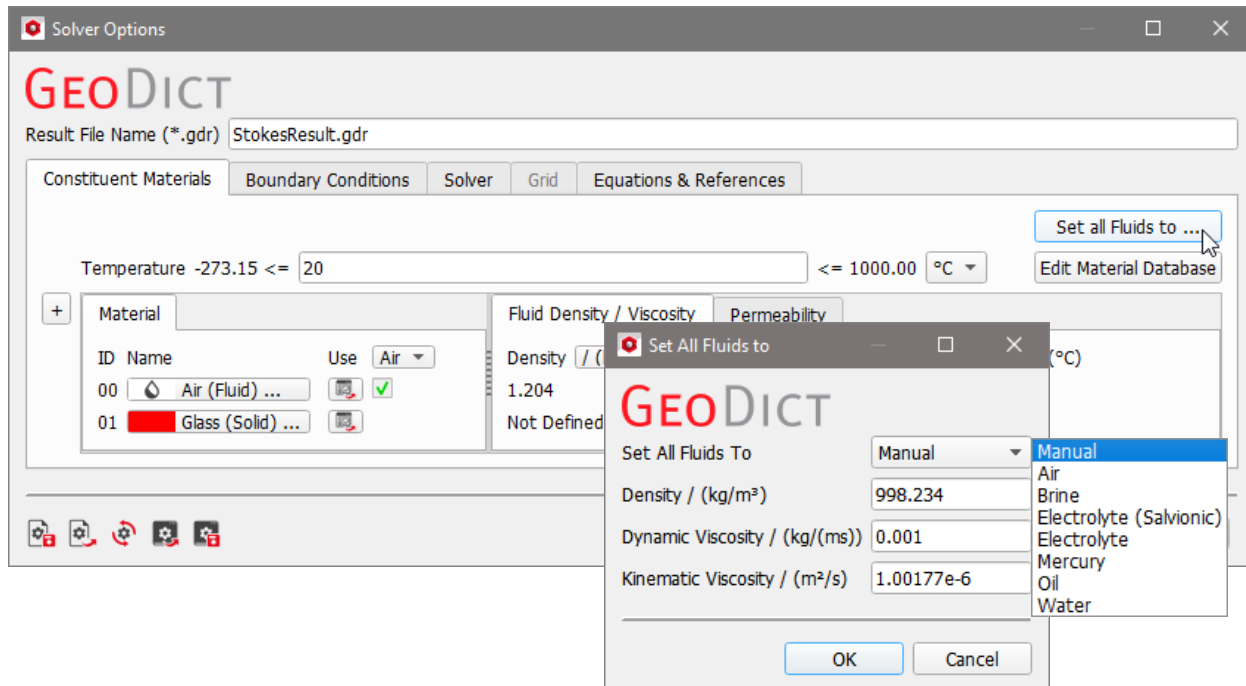
on Flow Area cm²

OK Cancel

Constituent Materials

For the flow solvers 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.

The fluid flowing through all pore and porous materials can be set by clicking **Set all Fluids to** and selecting it from the pull-down menu.

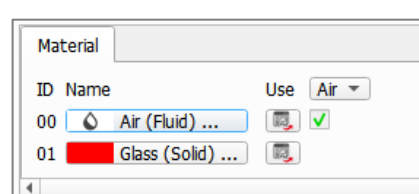
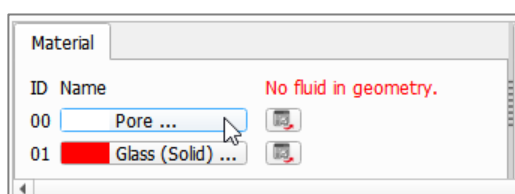


If one of the seven predefined fluids (Air, Brine, Electrolyte (Salvionic), Electrolyte, Mercury, Oil, or Water) is used, the values for **Density**, **Dynamic Viscosity** and **Kinematic Viscosity** are taken from the **GeoDict** Material Database and are dependent on the given **Temperature** value. If **Manual** is chosen, 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 **Edit Material DataBase...** button. For more information on editing, expanding, and using the [GeoDict Material Database](#) refer to the corresponding handbook in this User Guide.

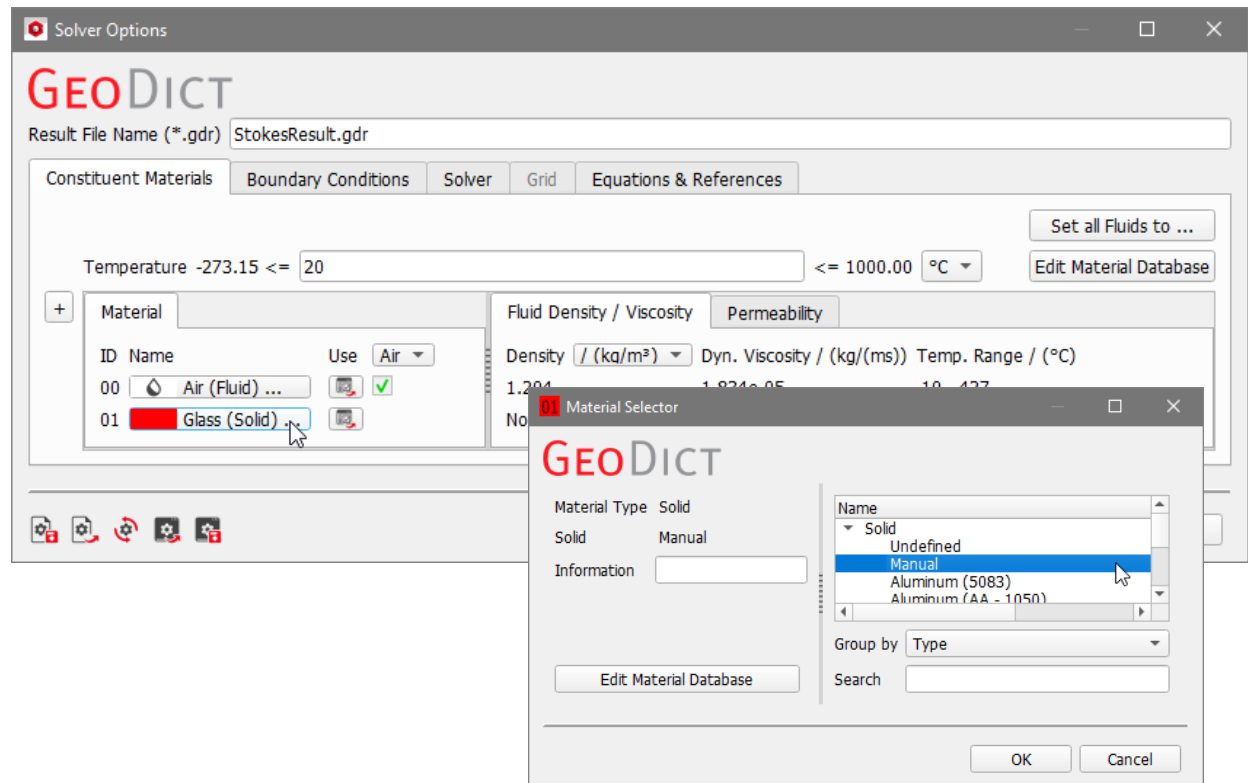
The **Temperature** is selectable in Kelvin (K), Celsius (°C), and Fahrenheit (F), and has default values of 293.15 K, 20.0°C, or 68 F, respectively.


Under the **Material** subtab, the fluid used for the simulation can be selected from the pull-down menu for **Use**. If other fluids are contained in the structure, they will be treated as solid materials 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 material ID to be treated as fluid, as described below.

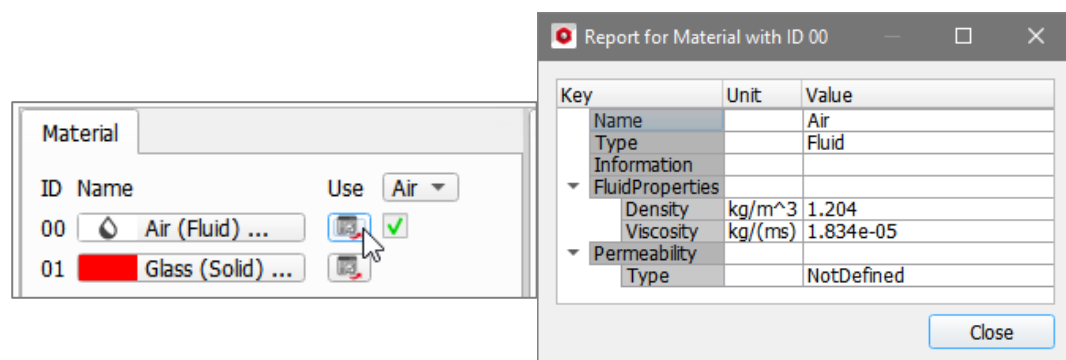


The density and the dynamic viscosity of this fluid, at the selected temperature range, is shown under the **Fluid Density/Viscosity** subtab.


Also, under the **Material** subtab, the material **IDs** assigned to the materials that are present in the structure model are shown (00, 01, 02, etc.). These material IDs, which fall into one of the material categories (Solid, Fluid, or Porous Solid), can be selected through the **Material Selector** dialog, by clicking on the material buttons.



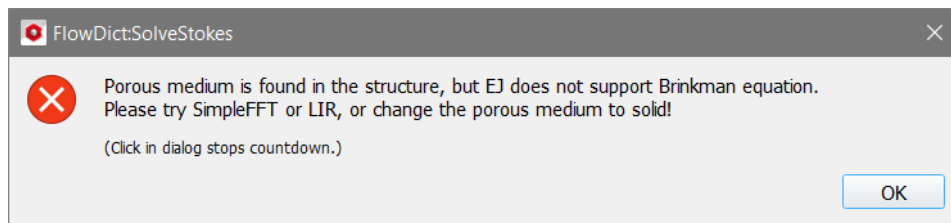
Click the **Material Report** button  to see the relevant parameters for the material with a selected ID.



Fluid flow can happen in pores and porous materials, but not in solid materials.

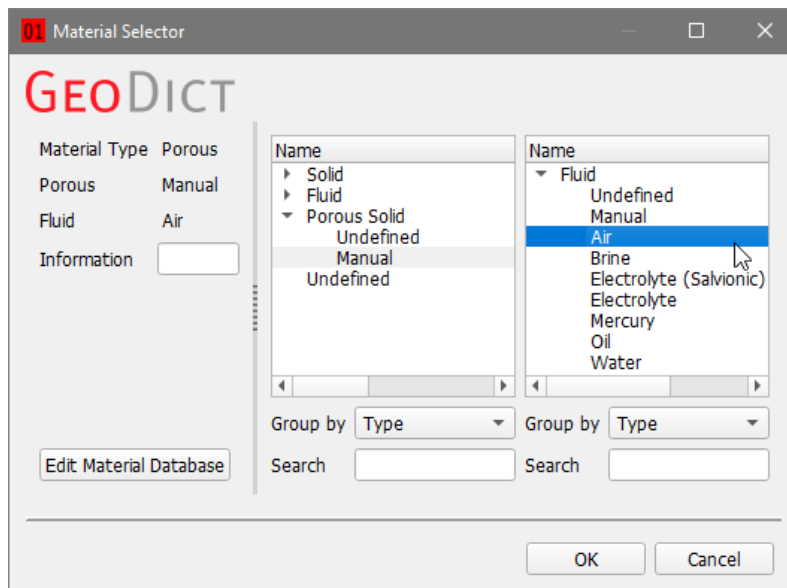
Next to the **Material Report** icon for all fluids and porous materials in the geometry a checkbox is displayed. The flow happens in all materials, where this checkbox  is checked.

When the structure model is made up of solid, pore, and porous voxels, the **SimpleFFT** or the **LIR** solver must be chosen under the **Solver** tab, as described on page [24](#). For the **EJ** solver an error message is shown after clicking **Run** (see page [9](#)).

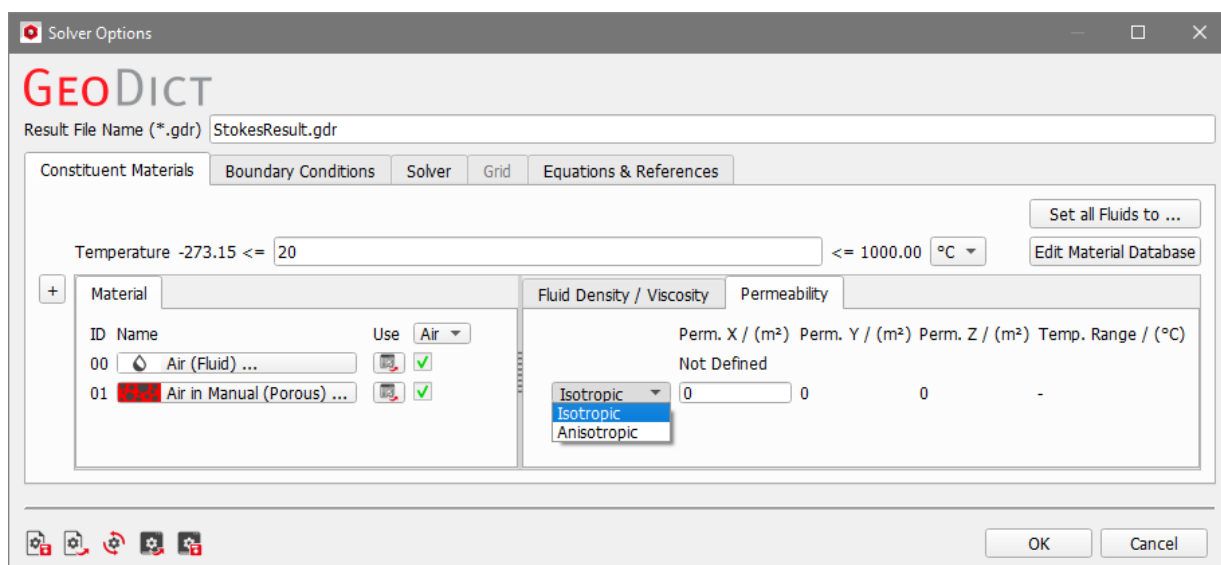


Empty voxels are infinitely permeable. For all other structure voxels, the user must define whether they are solid or porous. Solid voxels are set as shown above.

For porous voxels, the user sets the **Material Type** (in the Material Selector dialog) to **Porous** and chooses the fluid that occupies the voxels. The permeability of porous voxels is set under the **Permeability** subtab in the **Solver Options** dialog. Being permeable, it can be also defined whether the material displays isotropic or anisotropic orientation to the fluid flow.



When the material is isotropic, the permeability entered only for the X-direction is used for all directions, whereas for the anisotropic orientation of the material, the permeabilities in the three Cartesian directions are needed.



Currently, the local orientation of porous materials with anisotropic properties is not taken into account. The **SimpleFFT** must be chosen if anisotropic permeabilities are entered.

Boundary conditions

Boundary conditions are a required component of the mathematical model to reduce boundary effects. They direct the motion of the flow and determine the behavior of the flowing fluid when it finds an obstacle, such as the geometry (structure) or the boundary of the domain. Numerically, the chosen boundary conditions dictate the values taken by the parameters to make possible for the solvers to solve the flow differential equations. Different boundary conditions can be used in the flow direction and in the directions tangential to the flow.

Computation Directions

One or more **Computation Directions** to apply the boundary conditions may be selected.

To obtain the whole 3x3 permeability matrix in the result file, it is necessary to choose all three directions. To some extent, checking all directions for the run of the solver prolongs computational time.

GEODict Solver Options

Result File Name (*.gdr): StokesResult.gdr

Constituent Materials | **Boundary Conditions** | Solver | Grid | Equations & References

Computation Directions

☐ X ☐ Y ☒ Z

Boundary Conditions in Flow Direction

☒ Periodic
☐ Symmetric (Dirichlet)
☐ Velocity inlet, Pressure outlet
☒ Add implicit in-flow region / (Voxel): inflow 10 outflow 10

Experiment Input / Output

☒ Pressure Drop: 0.02
☐ Mean Velocity: 0.1
☐ Flow Rate: 60

Boundary Conditions in Tangential Direction

☒ Periodic ☐ Symmetric

Pore-Solid Boundary Conditions

Slip Length / (m): 0

Results

Input Map | Log Map | Post Map | **Results** | Flow Visualization | Create

Report | Plots | Map

Permeability tensor / (m²)

unknown	unknown	-3.12372e-14
unknown	unknown	6.4129e-14
unknown	unknown	2.44845e-12

Error Bound tensor

unknown	unknown	0.311825%
unknown	unknown	0.0352493%
unknown	unknown	0.785243%

Maximum error bound is 0.785243%
 (green): error <= 2%; (yellow): 2% < error <= 5%; (red): error > 5%.

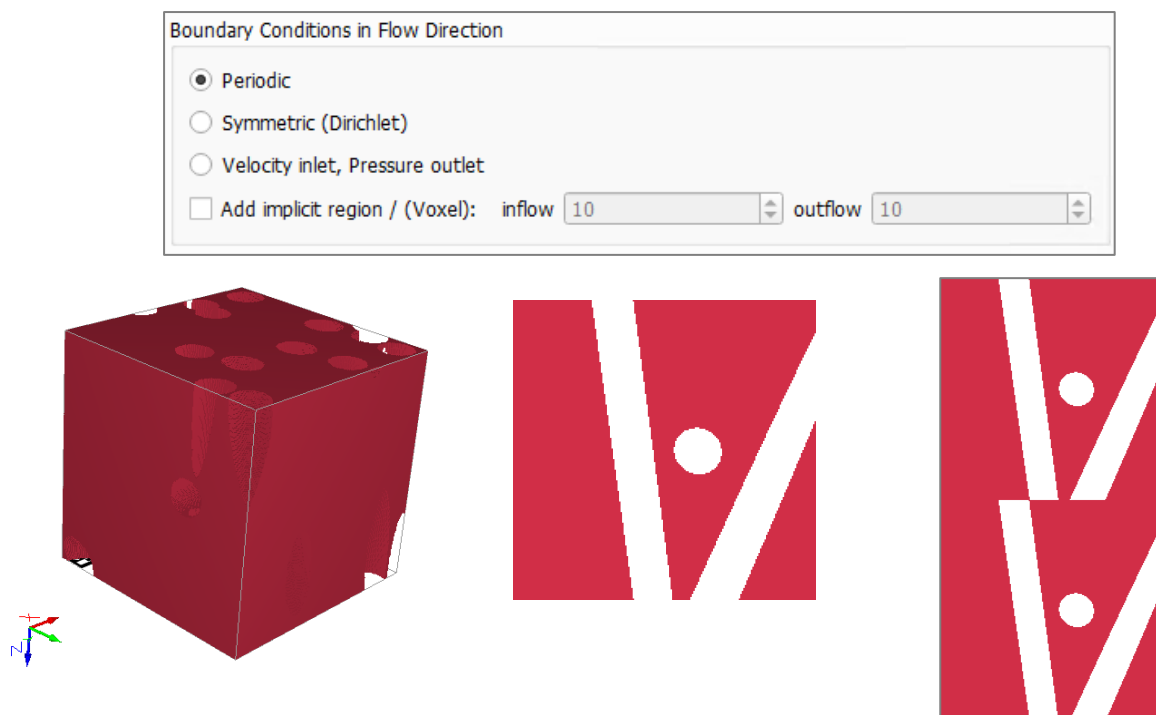
Boundary Conditions in Flow Direction

The **Boundary Conditions in Flow Direction** can be checked to be **Periodic**, **Symmetric**, or also to follow **Velocity inlet, Pressure outlet** boundary conditions.

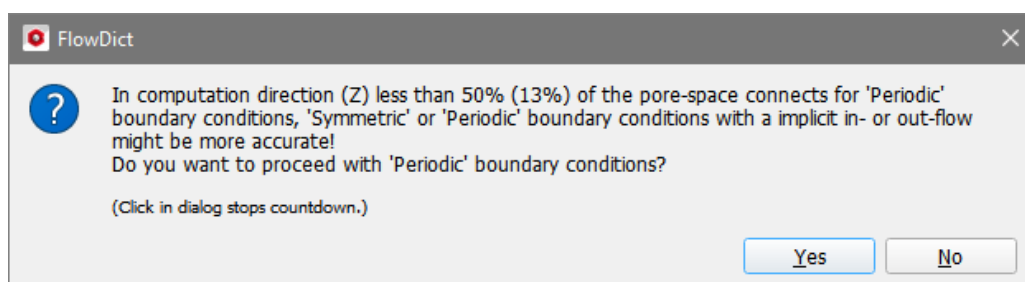
Periodic boundary conditions are recommended for periodically generated structure models and for non-periodic structures with high porosity. Note, that selecting **Velocity Inlet, Pressure outlet** for **Stokes(-Brinkman)** automatically selects the **SimpleFFT** solver in the **Solver** tab. The other two solvers cannot solve the Stokes(-Brinkman) equation with velocity inlet and pressure outlet, and thus, are greyed out. The **LIR** supports this boundary condition for **Navier-Stokes(-Brinkman)** only. The options in the **Solver** tab are described on page [24](#) and for more information about the different solvers refer to page [5](#).

After checking **Periodic** or **Velocity inlet, Pressure outlet** an inflow region (also called inlet) and outflow region (also called outlet) can be automatically added by checking **Add implicit region** and entering its size in voxels. The default added implicit inflow and outflow are 10 voxels, respectively. The inlet and outlet are essential to avoid the possibility of closing the flow channels when the structure is periodically repeated

For example, for the following structure choosing periodic boundary conditions in the direction of the flow without adding an implicit region results in the flow channels being artificially closed.



A warning appears when trying to run flow computations without adding an inflow region.



To open the channels and make the flow possible, the user should add an inlet

Boundary Conditions in Flow Direction

☒ Periodic

☐ Symmetric (Dirichlet)

☐ Velocity inlet, Pressure outlet

☒ Add implicit region / (Voxel): inflow 10 outflow 10



or choose symmetric boundary conditions.

Boundary Conditions in Flow Direction

☐ Periodic

☒ Symmetric (Dirichlet)

☐ Velocity inlet, Pressure outlet

☐ Add implicit region / (Voxel): inflow 10 outflow 10



Even so, when possible, we suggest using a periodic boundary condition, as the computational memory requirements are fairly low and therefore, the computational time is shorter.

The **Velocity inlet, Pressure outlet** (VinPout) boundary conditions apply a constant flow velocity in the inlet and a constant pressure drop in the outlet. The final result is influenced by the size of the inlet and outlet. Both are set with the value entered in **Add implicit inflow region**.

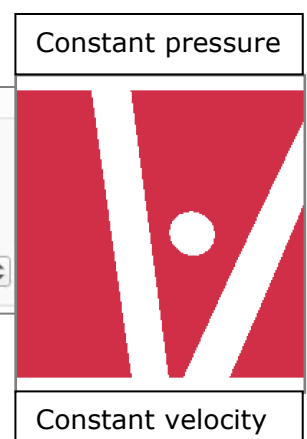
Boundary Conditions in Flow Direction

☐ Periodic

☐ Symmetric (Dirichlet)

☒ Velocity inlet, Pressure outlet

☒ Add implicit region / (Voxel): inflow 10 outflow 10



In some cases, it is very important to choose a larger inflow and outflow region, either by checking **Add implicit region** and entering values or with **ProcessGeo** (Process → Embed).

We recommend using at least 100 voxels for the inflow region and for the outflow region for weaves and for fast flows computed with **Navier-Stokes** equations.

Experiment Input / Output

In fluid dynamics, three experiments are typical:

- Measure the mean velocity for the applied pressure drop.
- Measure the pressure drop for a given mean velocity
- Measure the pressure drop for a given mass flow rate

Experiment Input / Output

☒ Pressure Drop 0.02 Pa

☐ Mean Velocity 0.1 m/s

☐ Flow Rate 60 l/min on Flow Area 100 cm²

The flow properties pressure drop, or mean velocity can be entered in the **Experiment Input / Output** panel. Input a prescribed **Pressure Drop** value and obtain the calculated **Mean Velocity** in the result file as output. Alternatively, the input of the mean velocity results in the output of pressure drop.

The **Pressure Drop** is the difference between the inflow and outflow pressures, and the **Mean Velocity** is the average speed of the flow in the positive Z-direction. The default values are set at 0.0002 mbar, or 0.02 Pascal ($1 \text{ Pa} = 0.01 \text{ mbar} = 10^{-5} \text{ bar}$) for pressure drop and 0.01 m/s for mean velocity.

In homogenization experiments, it is usual to input the pressure drop whereas for filtration experiments, it is the flow mean velocity, derived from the mass flow rate (g/cm^3).

The **Flow Rate on Flow Area** in l/min per cm^2 (default) or in other units, can be inputted to obtain the pressure drop at a given mean velocity. The volumetric flow rate per flow area is the volume of fluid which passes per unit time through a given area.

Boundary Conditions in Tangential Direction

The Boundary Conditions in Tangential Directions can be checked to be **Periodic**, **Symmetric**, **No-Slip**, or **Expert**.

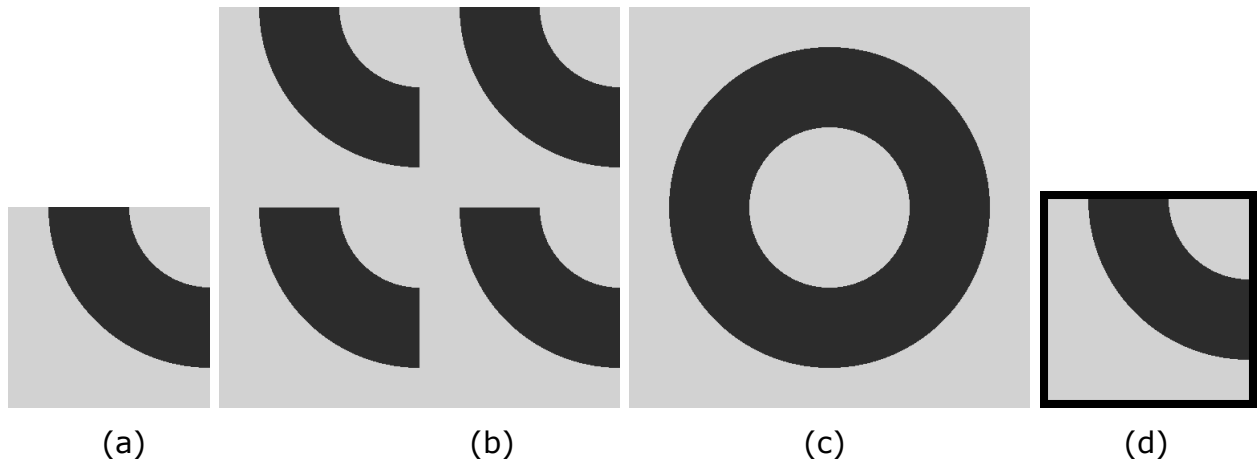
Boundary Conditions in Tangential Direction

☒ Periodic ☐ Symmetric ☐ No-Slip ☐ Expert

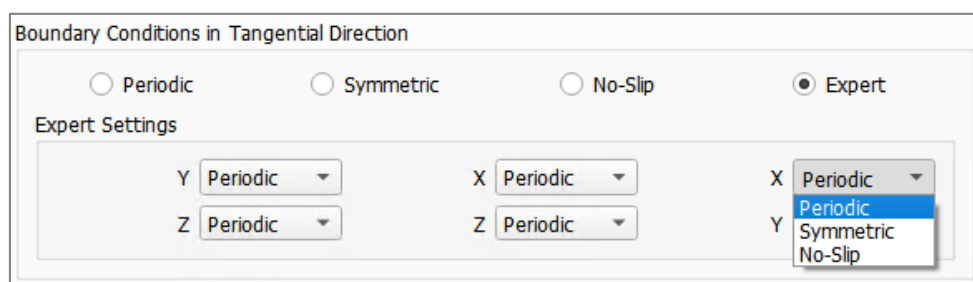
With the default **Periodic** selected, the process of periodic continuation is internally done during the run of the solver, repeatedly adding the volume structure in the directions tangential to the flow direction. Choosing the appropriate boundary condition depends on the structure's design.

For example, imagine a structure with a cross-section as shown in (a).

- If the expected pattern of the geometry is repeated in both tangential directions (b), the flow is computed with periodic boundary conditions.
- If instead the geometry has mirror symmetry (c), symmetric boundary conditions are taken.
- If the structure is encased in a closed wall (d), the no-slip boundary conditions are used in tangential directions.



The boundary conditions in the two directions tangential 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.



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 $n+1$.

Pore-Solid Boundary Conditions

The **Slip Length** allows to include sliding effects in the flow simulation. Sometimes the permeability to gases can be somewhat different from the permeability for 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 **Slip Length** of zero corresponds to a flow velocity of zero along the structure.

Pore-Solid Boundary Conditions

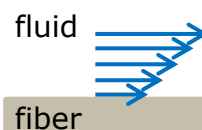
Slip Length / (m)



A non-zero **Slip Length** simulates the sliding of the fluid along the structure's fibers, increasing the fluid mean flux and thus, the permeability of the filter. 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.

Pore-Solid Boundary Conditions

Slip Length / (m)



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.

$$\vec{n} \cdot \vec{u} = 0 \text{ on } \Gamma \quad (\text{no flow into fiber})$$

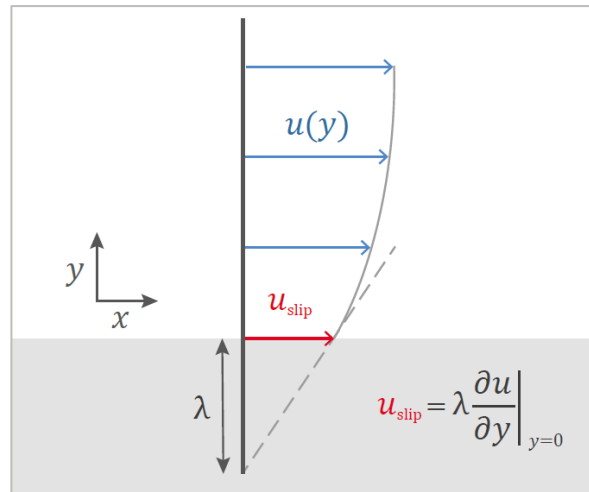
$$\vec{t} \cdot \vec{u} = -\lambda \vec{n} \cdot \nabla (\vec{u} \cdot \vec{t}) \text{ on } \Gamma \quad (\text{slip flow along fibers})$$

Here, \vec{u} is the fluid flow velocity as it was introduced in Darcy's law and the Stokes equation, \vec{n} is the normal direction to the fiber 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

$$u = \lambda \frac{\partial u}{\partial y} \quad (\text{slip flow in straight channels})$$

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

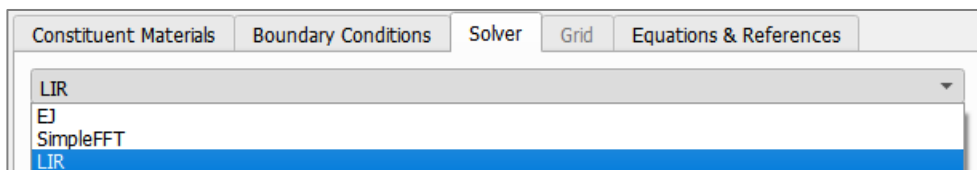


For a **Slip Length** of zero ($\lambda = 0$), the graph that describes the flow velocity in dependence of the distance to the fiber surface, would start at the surface of the fiber. However, for a positive **Slip Length** ($\lambda > 0$) it is shifted into the fiber by λ . Thus, the flow velocity u_{slip}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

Solver

Simulation Stopping Criterion

For **Stokes(-Brinkman)** three different solvers are available in FlowDict: **LIR**, **SimpleFFT**, and **EJ**. The **EJ** solver is not available for **Navier-Stokes(-Brinkman)**. As explained in page 5, the choice between **LIR**, **SimpleFFT**, or **EJ** is purely of numerical nature. Depending on the actual problem, one of them might be faster and/or require less memory. In case one of the methods does not produce the desired results, the other can be tried.



However, not all options are available for all solvers. Selecting the desired solver from the pull-down menu at the top of the **Solver** tab, activates the corresponding tab below.



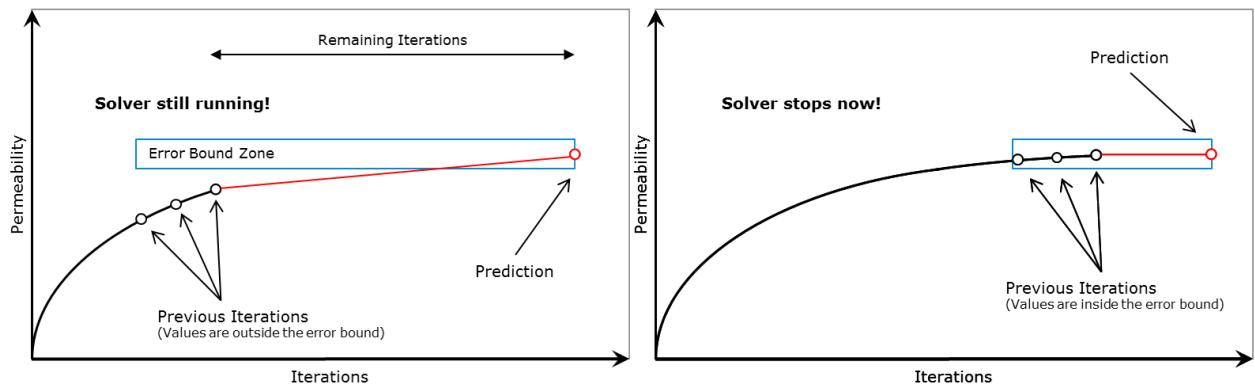
Internally, the solvers in FlowDict solve several equations for values of pressure and velocity at each voxel by using iterative solvers. The basic idea of an iterative method is to:

- Start with some initial guess for the unknown values,
- Improve the current values in each iterative step. The improvement can be fast or extremely slow depending on problem parameters,
- Repeat the iterative process until the stopping criteria is fulfilled.

The iterative process is controlled by setting the values and activation for **Error Bound**, **Tolerance**, **Residual**, **Maximal Iterations**, and **Maximal Run Time (h)**.

The default stopping criterion for the **SimpleFFT** and **LIR** FlowDict solvers is **Error Bound**. For **EJ** the default stopping criterion is **Tolerance**.

The **Error Bound** stopping criterion uses the result of previous iterations and predicts the final solution based on linear and quadratic extrapolation. **Error Bound** is available for the **LIR** and **SimpleFFT** solver. The solver stops if the relative difference in regard to the prediction 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.

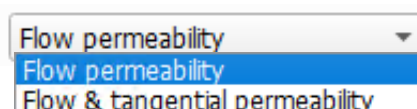


The **Tolerance** stopping criterion looks for stagnation of the method when the process becomes stationary, i.e. the improvement in the permeability value becomes extremely small from iteration to iteration. **Tolerance** is available for all solvers.

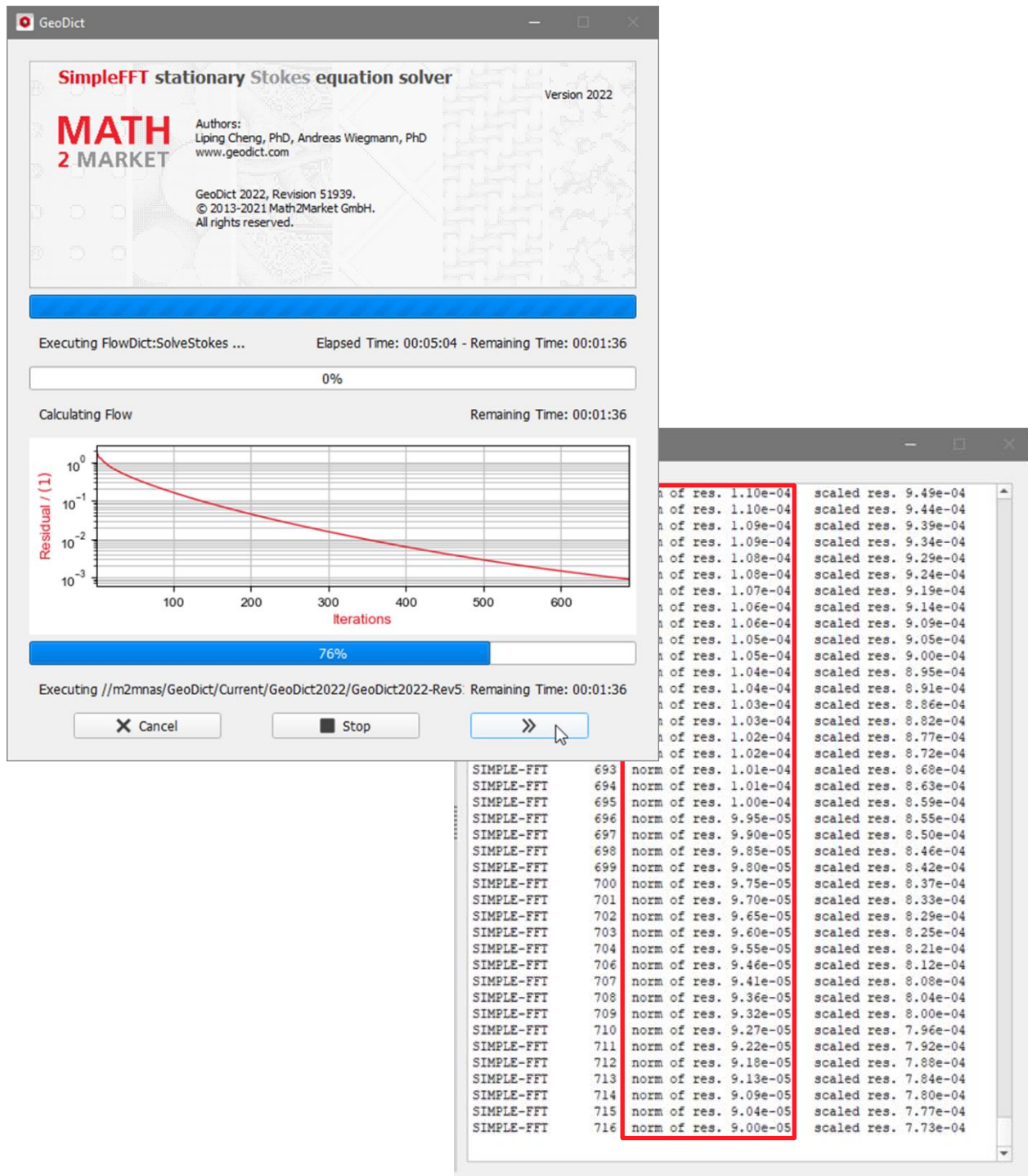
As a default, the solvers check for changes in the permeability after each 100 iterations. The iteration is stopped if the relative change is smaller than the value entered for **Tolerance**. When there is doubt about the quality of the solution, decrease the **Tolerance** value by a factor of ten for that solver. The drawback of this criterion is that flow solvers sometimes stopped too early in case of slow convergence rate or at local extrema of oscillatory convergence curves.

Simulation Stopping Criterion		
<input type="checkbox"/> Error Bound	0.01	Flow permeability
<input checked="" type="checkbox"/> Tolerance	0.0001	Flow permeability
<input type="checkbox"/> Residual	0.0001	
<input type="checkbox"/> Maximal Iterations	100000	
<input type="checkbox"/> Maximal Run Time / (h)	240	
Restart Save Interval / (h)		6

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



By setting the stopping criterion to **Residual**, the computations terminate as soon as the relative norm drops below the selected residual threshold. **Residual** is only available for **SimpleFFT** and **EJ**. 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 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.

For the **SimpleFFT** and **LIR** solver it is recommended using **Error Bound**, as this stopping criterion approximates “relative deviation from the final solution”!

When the solver does not stop due to the stopping criterion **Error Bound**, **Tolerance**, or **Residual** and instead, stops because the **Maximal Iterations** value or the **Maximal Run Time (h)** has been reached, no guarantee on the quality of solution can be given.

Which stopping criterion has occurred can be seen in the result file (GDR file).

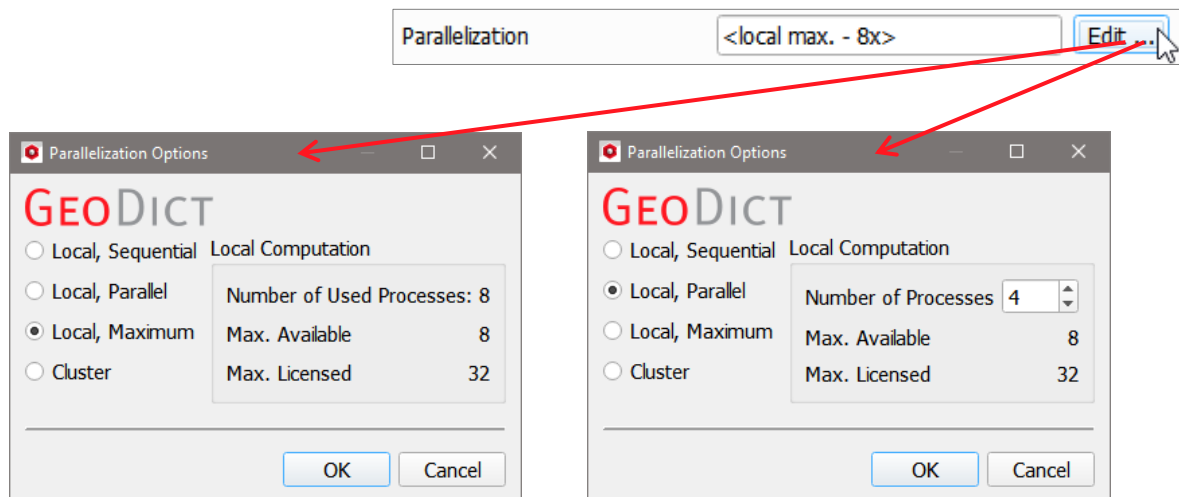
Restart Save Interval

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

Restart Save Interval / (h)

Parallelization

Calculations can be parallelized if the user’s license and hardware allow it. The **Parallelization Options** dialog opens when clicking the **Edit...** button, to choose between **Local, Sequential**, **Local, Parallel** or **Local, Maximum**.



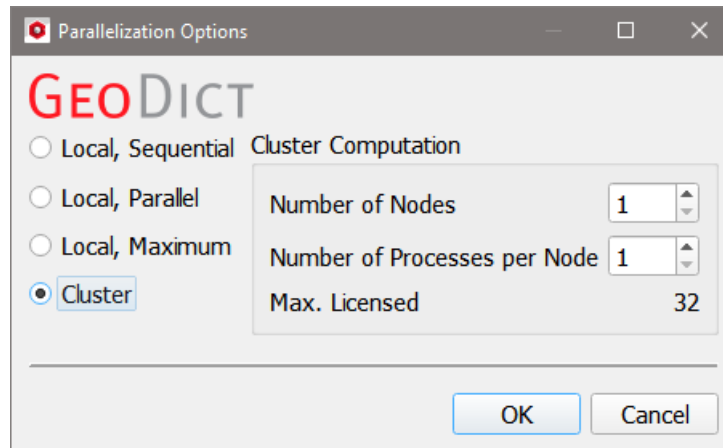
When **Local, Parallel** is selected, the **Number of Processes** can be entered. Then the maximum number of available processors and the maximum number of licensed parallel processes is shown in the dialog.

The parallelization of the solvers is done with two technical methods: MPI parallel or thread parallel. The following table shows the support of both parallelization methods:

	Parallelization method	
	MPI Parallel	Thread Parallel
EJ solver	✓	✗
SimpleFFT solver	✓	✗
LIR solver	✗	✓

The **Cluster** parallelization requires that the solver supports the MPI parallelization method. The **Local, Parallel** parallelization can be done with MPI parallelization or Thread parallelization. Thus, the LIR solver does not support Cluster parallelization.

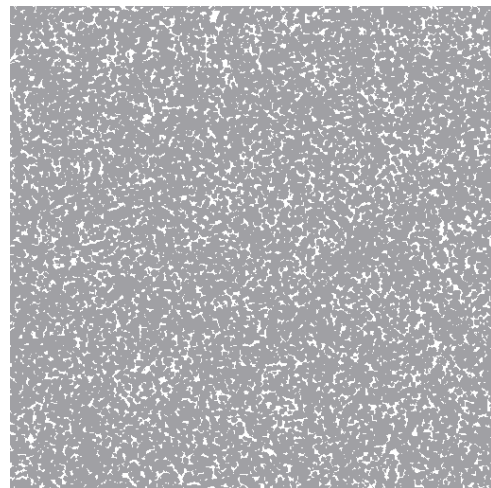
The choice of **Cluster** is for users of Linux systems only. For details on how to set up and run parallel computations, consult the [High Performance Computations](#) handbook of the User Guide.



Parallelization Benchmark Results

Two examples of parallelization benchmark results for FlowDict are shown here. Both benchmark computations were run on our 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.

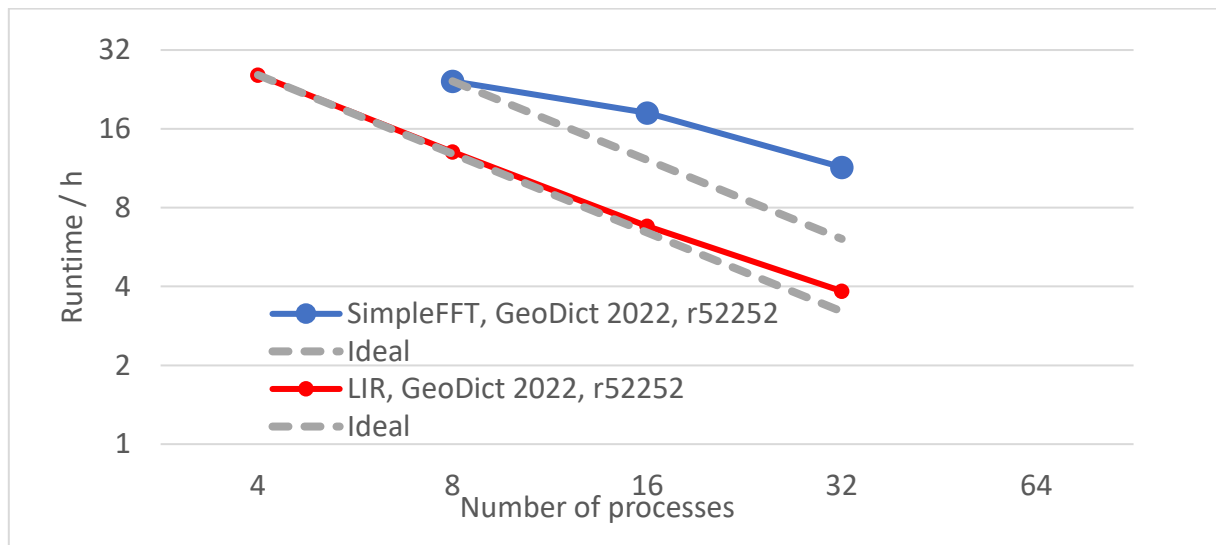
The first example is the computation of the flow through a synthetic X-ray tomography data of a sandstone structure [2]. The structure has a porosity of 13.5% and a size of 2,048 x 2,048 x 2,032 voxels.



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**.

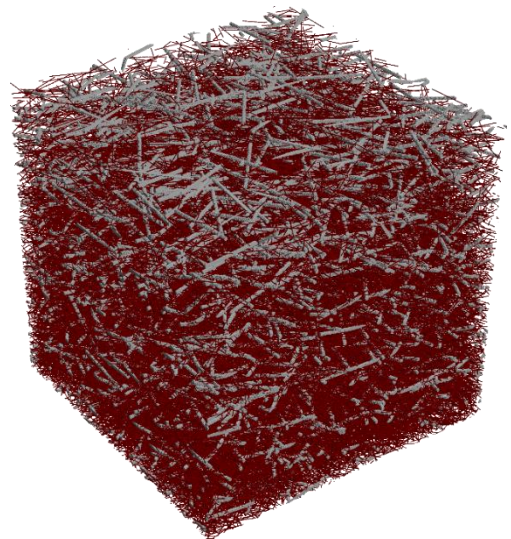
The flow through the structure is computed by solving stokes equations once with **LIR** solver (with speed optimization) and once with **SimpleFFT** solver. The following figure shows the runtime for a different number of processes to compute the permeability of $3.3e-13 \text{ m}^2$ with an error bound of 0.01 with both solvers. Runtime and speedup of the parallelization are better for the **LIR** solver. For both solvers, the ideal speedup, i.e. getting half of the runtime for twice of the number of processes is

also shown. Computations with **LIR** solver are, for both examples shown here, close to the ideal speedup, which is usually not possible to reach for real life examples.



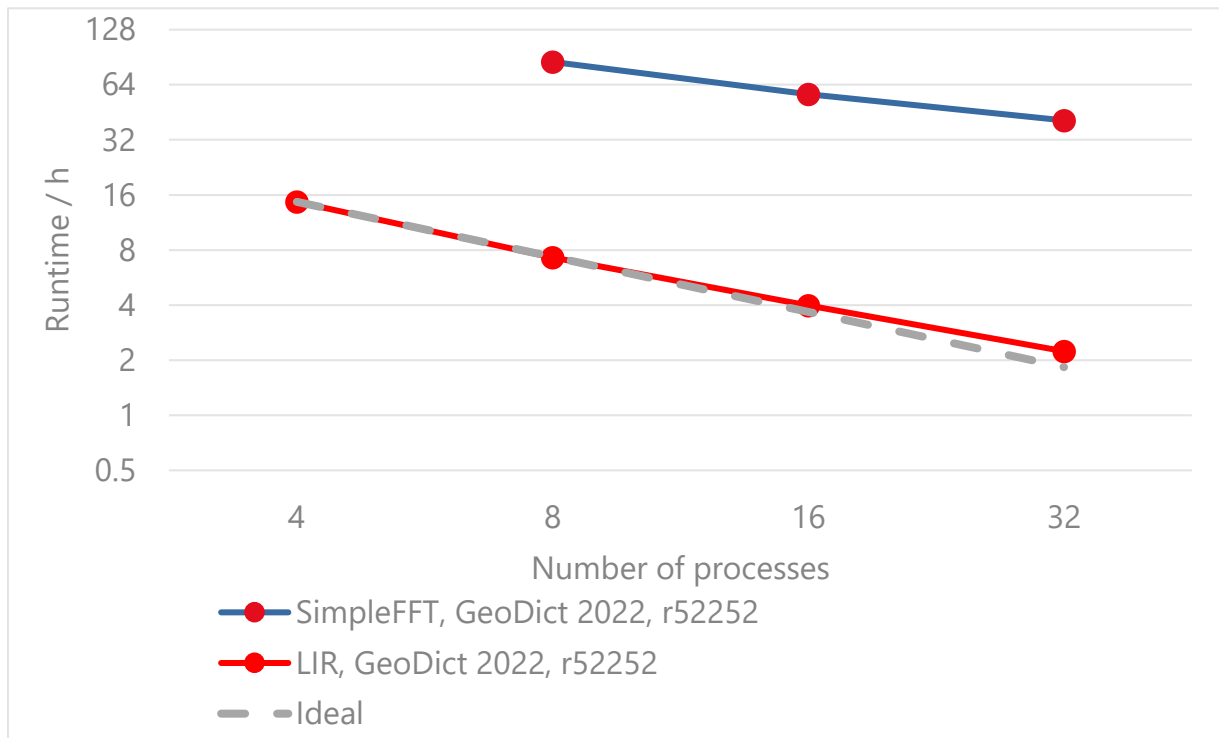
The second benchmark is the computation of the flow through a filter structure with linear increasing fiber density.

The structure size is again $2,048 \times 2,048 \times 2,032$ 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 Stokes equations, with the **LIR** (with speed optimization) and the **SimpleFFT** solver.

The following figure shows the runtime for a different number of processes to compute the permeability of $5.63e-11 \text{ m}^2$ for **SimpleFFT** and $5.69e-11 \text{ m}^2$ for **LIR**. Both results are equal within the chosen error bound of 1%. The benefit of using **LIR** solver compared to **SimpleFFT** is large for the structure with high porosity.



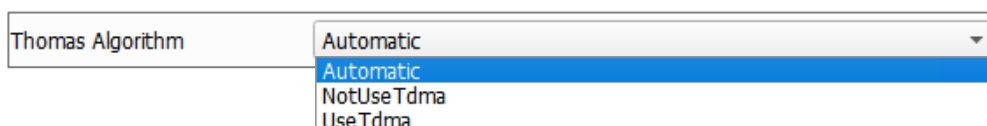
Memory requirements for both examples and both solvers are listed in the following table:

Micro-structure	Memory for computation with LIR	Memory for computation with SimpleFFT for 32 Processes
Sandstone	136GB	660GB
Filter	170GB	660GB

Thomas Algorithm (for SimpleFFT solver)

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.

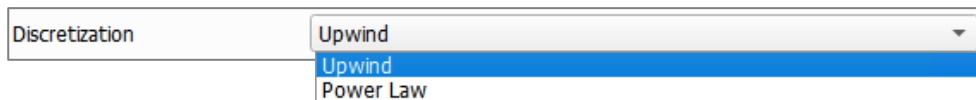
For more information about the Thomas algorithm see the [Wikipedia](#) article.

Discretization

Discretization is an experimental feature only available for **Navier-Stokes(-Brinkman)**, when the **SimpleFFT** solver is selected. As described on page 3, Navier-Stokes uses the additional term for **inertia**:

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

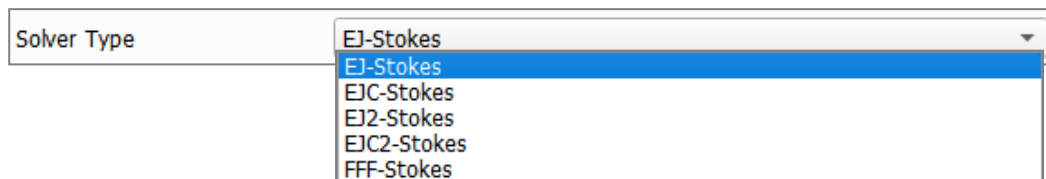
This term can be discretized with two different methods: **Upwind** and **Power Law**. The descriptions of the upwind and Power Law schemes can be found in the Appendix (page 59).



In most cases, using **Upwind** is recommended, since using **Power Law** requires more memory and runtime. However, if the solver does not converge using Upwind, Power Law can be tried.

Solver type (for Stokes EJ solver)

For the computation of **Stokes (EJ)** flow, FlowDict provides five different ways to implement no-slip boundary conditions of the flow solver: **EJ-Stokes**, **EJC-Stokes**, **EJ2-Stokes**, **EJC2-Stokes**, and **FFF-Stokes**.



The implementations describe how the no-slip boundary conditions are discretized, i.e. at which discrete point the tangential velocity reaches zero.

The default **EJ-Stokes** (Explicit Jump-Stokes solver) sets the tangential velocity to zero at the center of the voxel surfaces. It is a newer implementation of the Finite Volume solver that solves moderately slower, but it is noticeably more accurate for low porosity structures.

EJC-Stokes sets the tangential velocity to zero at the voxel corners.

EJ2-Stokes sets the tangential velocity to zero at the center of the voxel surfaces and uses a 2nd order approximation of the velocity field.

EJ2C-Stokes sets the tangential velocity to zero at the voxel corners and uses a 2nd order approximation of the velocity field.

FFF-Stokes sets the tangential velocity to zero at the voxel centers.

Relaxation (for SimpleFFT and LIR solver)

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.

Velocity relaxation	Stable	<input type="range"/>	Fast	<input type="text" value="1"/>
Pressure relaxation	Stable	<input type="range"/>	Fast	<input type="text" value="1"/>

For the **LIR** solver, this balance is managed through the **Relaxation**. The value should be between zero and two. For relaxation values smaller than one (<1.0), the simulation is more stable. For relaxation values larger than one (>1.0), the simulation is faster.

Relaxation	<input type="text" value="1"/>
------------	--------------------------------

Restart from .gdr file

When continuing an iteration process or using similar parameter values for a **FlowDict** computation, a great amount of time can be saved by restarting the process from an existing **GeoDict** result file (*.gdr).

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

<input checked="" type="checkbox"/> Restart from .gdr File	<input type="text"/>	<input type="button" value="Browse"/>
--	----------------------	---------------------------------------

Discard PDE Solver Files / Analyze Geometry

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.

<input type="checkbox"/> Discard PDE Solver Files
<input checked="" type="checkbox"/> Analyze Geometry

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

Checking **Analyze Geometry** performs the analysis of the geometry and is enabled by default. It should be turned off when geometry analysis is not necessary.

The flow solver tries to solve the flow in all the pores of the structure, regardless of whether they are open or closed pores. However, flow in the closed pores may not

exist or does not influence the whole flow field. The efforts in solving the flow in the closed pores are wasted.

For this reason, a geometrical analysis is routinely run before the solver computations to determine whether a through path exists, to remove unconnected solid components, and to fill up unconnected empty spaces. This geometry analysis also takes time, so that when the user knows beforehand that closed pores do not exist in a structure 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 (for LIR)

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

If the option **Write Compressed Volume Fields** is checked then the adaptive grid is used as compression method for writing out velocity and pressure (VAP) fields. 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.

☒ Write Compressed Volume Fields

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

Use Multigrid Method (for LIR)

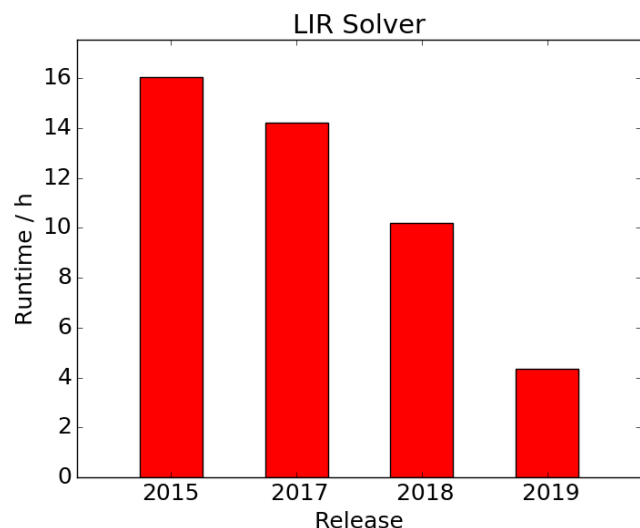
The **Multigrid Method** was introduced to speed-up the computation and reduce the runtime significantly. The main idea of **Multigrid** is the usage of multiple coarser adaptive grids to speed up convergence behavior but requires only little more memory.

☒ Use Multigrid Method

The method is available for solving the **Stokes** and **Stokes-Brinkman** equations in **FlowDict** and **FilterDict** as well as for solving diffusion, thermal and electrical conduction in **DiffuDict** and **ConductoDict** and is enabled by default.

The strength of the **Multigrid Method** is shown for Stokes flow on four of the digital rocks used in [3] with porosities ranging from 9% to 22%.

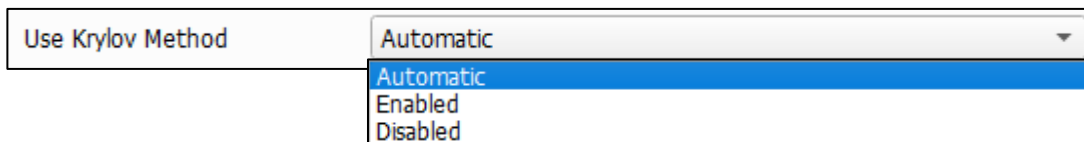
The graph on the right displays the total runtime for the **GeoDict** 2019 release, when the **Multigrid Method** was introduced, compared to three earlier **GeoDict** releases without **Multigrid**. The runtime could be reduced by more than 50% compared to **GeoDict** 2018.



Use Krylov Method (for LIR)

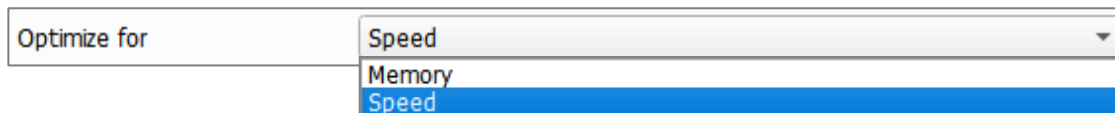
In GeoDict2022 another speed-up option was introduced accelerate the convergence behavior of the **LIR** solver. The runtime of the **LIR** solver depends on many different properties of the structure and the simulation parameters. For example, the runtime is high if the porosity is very small – let's say less than 15%. This is due to the growing complexity of the pore space in low porous structures. For **Stokes-Brinkman** the runtime also depends on the permeability of porous voxels and the voxel length. For **Navier-Stokes** it depends on the applied velocity of pressure drop. For example, the **Velocity-in Pressure-out** boundary condition can increase the runtime significantly. These challenging simulations can now be tackled with a special solution method called **Krylov Method**.

The method was originally implemented for the Battery-LIR solver but is now available in the flow solver too. Here, the BiCGStab algorithm is used under the hood. This method can reduce the runtime for challenging simulation very drastically, but this is paid by an increased memory consumption. Unfortunately, the new method is not always faster than the old method and therefore we introduced an **Automatic** mode which uses some heuristics to choose the best solution for the user automatically. Of course, it is possible to explicitly enable (**Enabled**) or disable (**Disabled**) the method.



Optimize for (for LIR)

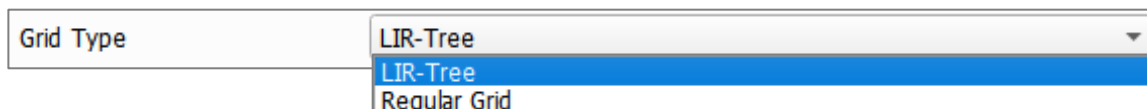
The **LIR** solver can **Optimize for** speed or memory.



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, then the runtime is increased by up to 40% but the solver requires up to 50% less memory.

Grid Type (for LIR)

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

Grid Refinement (for LIR)

The solver can analyze the velocity and pressure field during the computation and improves the adaptive grid in places where more accuracy is needed. The LIR solver

splits cells where a high velocity-gradient or high pressure-gradient occurs. The analysis is enabled if the **Grid Refinement** option is set to **Enabled** or **Manual**.

If the **Grid Refinement** is set to **Enabled**, the solver chooses the **Number of Grid Refinements** and **Threshold for Grid Refinement** automatically.

Grid Refinement	Enabled
	Enabled
	Disabled
	Manual

If the **Grid Refinement** is set to **Manual**, the user can enter the parameters manually.

Grid Refinement	Manual
Threshold for Grid Refinement	0.1
Number of Grid Refinements	10

The **Number of Grid Refinement** controls how many velocity-based and pressure-based grid refinements are allowed during the simulation. The value should be between 0 and 5. Velocity-based and pressure-based grid refinements may increase the number of iterations, runtime and memory requirements.

The **Number of Grid Refinement** can be zero in most of the cases and should be greater than zero 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.

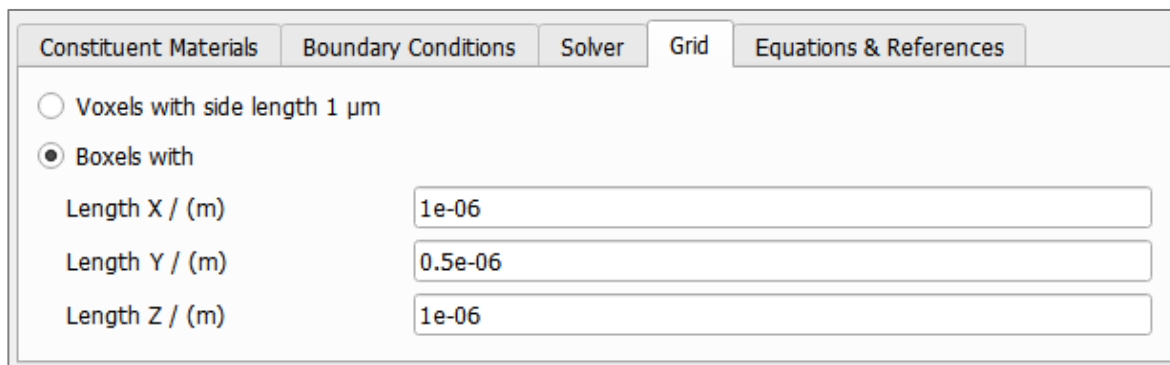
Refinement is done at regions with high-velocity gradient or high-pressure gradient. Cells are split where the current velocity gradient (or pressure gradient) is greater than the **Threshold for Grid Refinements** times the maximal velocity gradient (or pressure gradient). This threshold must be between 0.0 and 1.0. The recommended value range is between 0.05 and 0.1.

Grid

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

Sometimes, grid lengths might 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 geometry is loaded in **GeoDict**, each grid cell gets assigned 1 voxel and the visual representation of the structure looks stretched. To correct this effect, select **Boxels with** e.g. **Length X** = 1 μm and **Length Y** = 0.5 μm .



Constituent Materials		Boundary Conditions		Solver		Grid		Equations & References	
<input type="radio"/> Voxels with side length 1 μm									
<input checked="" type="radio"/> Boxels with									
Length X / (m)				<input type="text" value="1e-06"/>					
Length Y / (m)				<input type="text" value="0.5e-06"/>					
Length Z / (m)				<input type="text" value="1e-06"/>					

Equations & References

The **Equations & References** tab shows the formulas that are used for the solver, described on pages [3ff](#):

- Momentum Conservation
- Mass Conservation
- No-Slip Boundary Condition

No parameters can be edited on this tab.

Also **References** used for this module are given.

HINTS TO ESTIMATE MEMORY REQUIREMENTS FOR FLOW SOLVERS

The memory 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** solver, the **SimpleFFT** solver, and the **LIR** solver.

We assume a structure with

- NX, the voxels in X-direction,
- NY, the voxels in Y-direction,
- NZ, the voxels in Z-direction,
- NI, additional inlet voxel layers,
- NO, additional outlet voxel layers, and
- 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

$$\text{NumberOfVoxels} = \text{NX} * \text{NY} * (\text{NZ} + \text{NI} + \text{NO})$$

The **EJ** and **SimpleFFT** 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

$$\text{NumberOfVoxelsWithOverlap} = \text{NX} * \text{NY} * (\text{NZ} + \text{NI} + \text{NO} + 2 * \text{P})$$

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

Memory estimation for EJ Solver

The memory required by the **EJ** 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:

$$\text{RequiredMemory} = 60 * \text{NumberOfVoxelsWithOverlap}$$

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

$$\text{RequiredMemory} = (-132 * \text{SVF}^2 + 166 * \text{SVF} + 30) * \text{NumberOfVoxelsWithOverlap}$$

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** solver depends on the total number of voxels, the boundary conditions, and the number of parallel processes. A rule of thumb is:

$$\text{RequiredMemory} = 80 * \text{NumberOfVoxelsWithOverlap}$$

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 Solver

The estimation of the **LIR** 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 or Stokes-Brinkman), 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:

$$\text{RequiredMemory} = 5 * 10^7 + \text{NumberOfCells} * (63 + 44 * \text{OptimizeForSpeed} + 8 * \text{WithBrinkman})$$

where

$$\text{NumberOfCells} = \text{NumberOfVoxels} / (60 * \text{SVF}^2 - 56 * \text{SVF} + 15)$$

$$\text{OptimizeForSpeed} = 1 \text{ if Optimize is set for Speed else } 0$$

$$\text{WithBrinkman} = 1 \text{ if Stokes-Brinkman or Navier-Stokes-Brinkman is computed else } 0$$

FORCHHEIMER APPROXIMATION OPTIONS

The **Forchheimer Approximation** allows the user to estimate pressure drop or mean velocities for high Reynolds numbers that present solvers cannot handle yet. It is also possible to change the fluid properties (i.e. viscosity and density) afterwards and estimate pressure drop or mean velocities for other fluids.

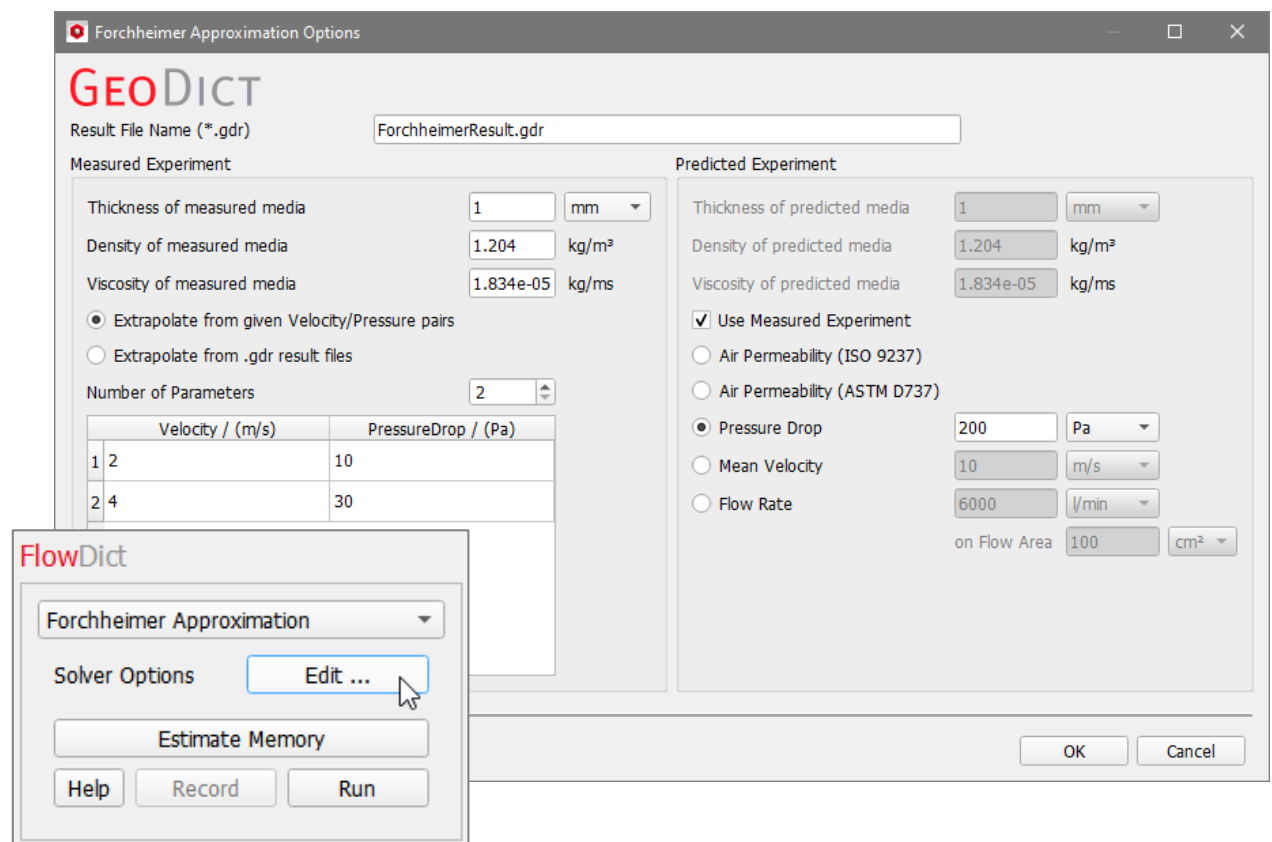
Forchheimer's equation describes the pressure drop of a fluid flow (in a porous media). The equation extends Darcy's law (see page [1](#)) and considers the pressure drop caused by turbulence in addition to the pressure drop caused by dynamic viscosity.

$$-\nabla p = \frac{\mu}{K} \vec{u} + \frac{\rho}{k_2} \vec{u}^2 \quad (\text{Forchheimer's equation})$$

where \vec{u} is the fluid flow velocity, K is the permeability, μ is the fluid viscosity, p is the pressure, and ρ is the density.

The new symbol k_2 is the non-Darcy permeability coefficient and depends on the porous media only. It does not depend on the flowing fluid.

As mentioned above in page [12](#), the **Forchheimer Approximation Options** dialog is completely different to the **Solver Options** dialog for the **Stokes(-Brinkman)** and **Navier-Stokes(-Brinkman)** equations. There are no tabs, and all solver parameters are entered directly in the single dialog.



At the top of the dialog, a customized **Result File Name (*.gdr)** should be entered to differentiate the results of sets of Forchheimer approximations.

The **Forchheimer Approximation Options** dialog is organized in two panels.

Set up the parameters measured in experiments in the **Measured Experiment** panel and the parameters to use for the simulation in the **Predicted Experiment** panel.

Measured Experiment

In the **Measured Experiment** panel, enter fluid properties and the thickness of the media used during the experiments. These values may differ from the corresponding values in the **Predicted Experiment** panel.

Thickness of measured media	<input type="text" value="1"/>	mm
Density of measured media	<input type="text" value="1.204"/>	kg/m ³
Viscosity of measured media	<input type="text" value="1.834e-05"/>	kg/ms

Enter the **Thickness of measured media** used during the experiments.

Density of measured media is the density of the fluid used in the measured experiment.

Viscosity of measured media is the viscosity of the fluid used in the measured experiment.

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.

With **Extrapolate from given Velocity/Pressure pairs** checked by default, 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

<input checked="" type="radio"/> Extrapolate from given Velocity/Pressure pairs	
<input type="radio"/> Extrapolate from .gdr result files	
Number of Parameters <input type="text" value="2"/>	
Velocity / (m/s)	PressureDrop / (Pa)
1 2	10
2 4	30

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, e.g. **Navier-Stokes(-Brinkman)**. The velocity / pressure drop pairs are then extracted from the given files.

<input type="radio"/> Extrapolate from given Velocity/Pressure pairs	
<input checked="" type="radio"/> Extrapolate from .gdr result files	
Number of Parameters <input type="text" value="2"/>	
Gdr	Files
1 Browse...	MeasuredExperiment-1.gdr
2 Browse...	MeasuredExperiment-2.gdr

Predicted Experiment

In the **Predicted Experiment** panel, enter fluid properties and the thickness of the media to use for the prediction. These values may differ from the corresponding values in the **Measured Experiment** panel.

Thickness of predicted media	1	mm
Density of predicted media	1.204	kg/m ³
Viscosity of predicted media	1.834e-05	kg/ms
<input type="checkbox"/> Use Measured Experiment		

Thickness of predicted media is the thickness of the media that should be used during the experiment.

Density of predicted media is the density of the fluid that should be used during the experiment.

Viscosity of predicted media is the viscosity of the fluid used in the measured experiment.

To use the same values as for the measurement check **Use Measured Experiment**. Then, the value options are greyed out.

Thickness of predicted media	1	mm
Density of predicted media	1.204	kg/m ³
Viscosity of predicted media	1.834e-05	kg/ms
<input checked="" type="checkbox"/> Use Measured Experiment		

After setting up the fluid properties and media thickness, enter a pressure drop, mean velocity, or flow rate for the predicted experiments. 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. These parameters are similar to the **Boundary Conditions** parameters for the other solvers as seen above (page [20](#)).

<input checked="" type="radio"/> Pressure Drop	200	Pa
<input type="radio"/> Mean Velocity	10	m/s
<input type="radio"/> Flow Rate	6000	l/min
		on Flow Area 100 cm ²

In addition, there are two presets for pressure drop.

- **Air Permeability (ISO 9237)**: The prescribed pressure drop will be 200 Pa according to ISO 9237.
- **Air Permeability (ASTM D737)**: The prescribed pressure drop will be 125 Pa according to ASTM D737.

<input checked="" type="radio"/> Air Permeability (ISO 9237)
<input type="radio"/> Air Permeability (ASTM D737)

FLOWDICT SIMULATION RESULTS

At the end of the solver computations for FlowDict, the following are automatically saved in the selected project folder:

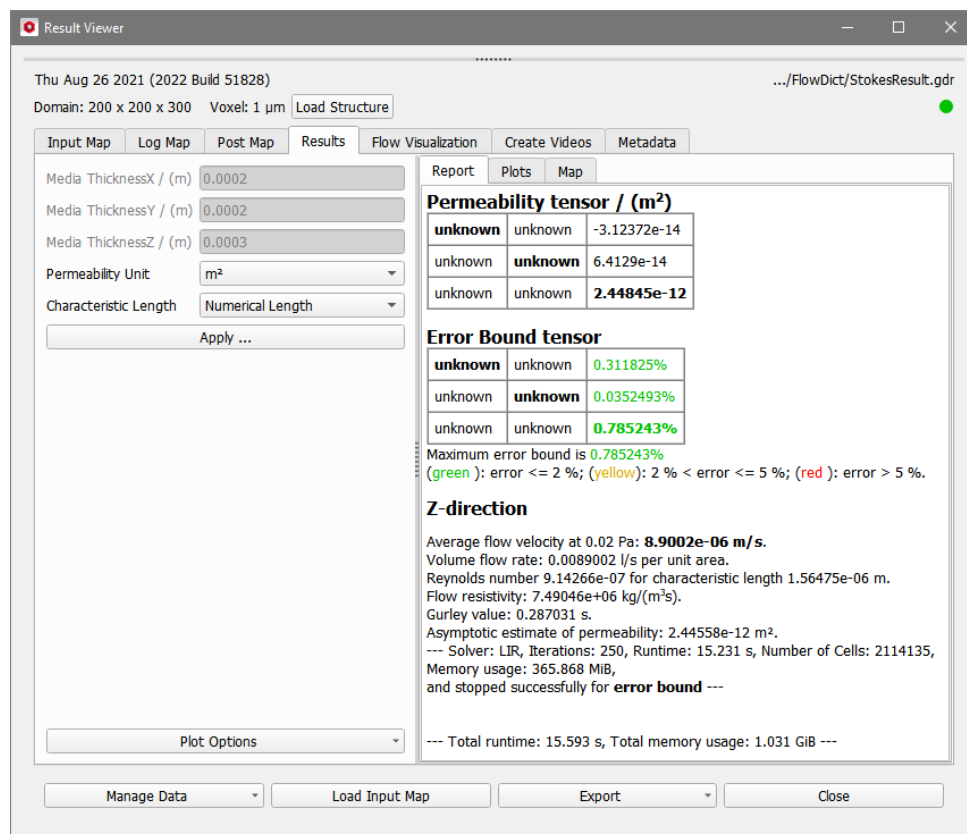
- A result file, e.g. StokesResult.gdr, the default name when solving the Stokes(-Brinkman) equation.
- A result folder with the same name as the result file (e.g. StokesResult). This folder is deleted after the solver's run if **Discard PDE Solver Files** is checked in the **Solver Options** dialog.



Find detailed information about the Result Viewer options, see the [Result Viewer handbook](#) of this User Guide.

RESULTS FOR STOKES(-BRINKMAN) AND NAVIER-STOKES(-BRINKMAN)

After the solver has finished, the result file (.gdr) is saved in the project folder and the **Result Viewer** automatically opens.



RESULTS - REPORT

The **Report** tab under the **Results** tab shows the computed **Permeability** (3x3 tensor) and values for **Average flow velocity** at a given pressure drop (or **Pressure Drop** at a given mean flow velocity), **Flow resistivity**, estimated **Reynolds number**, and **Gurley value** in the selected computation directions. Information about the convergence of the iterative solver is also displayed: Iterations, Runtime, Memory usage, and applied stopping criterion.

Below the **Permeability tensor** the **Error Bound tensor** (or **Tolerance 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 where the **Flow & tangential permeability** is considered in the stopping criterion as described on page [25](#).

For user post-processing, the value of the physical **Media Thickness** in X-, Y-, and Z-direction can be changed in the post-processing settings section on the left. The **Media Thickness** corresponds to the symbol **L** in the description of permeability in page [1](#).

Input Map Log Map Post Map Results Flow Visualization Create Videos Metadata

Media ThicknessX / (m) 0.0002

Media ThicknessY / (m) 0.0002

Media ThicknessZ / (m) 0.0003

Permeability Unit m²

Characteristic Length Numerical Length

Apply ...

Report Plots Map

Permeability tensor / (m²)

unknown	unknown	-3.12372e-14
unknown	unknown	6.4129e-14
unknown	unknown	2.44845e-12

Error Bound tensor

unknown	unknown	0.311825%
unknown	unknown	0.0352493%
unknown	unknown	0.785243%

Maximum error bound is 0.785243%
(green): error <= 2 %; (yellow): 2 % < error <= 5 %; (red): error > 5 %.

Z-direction

Average flow velocity at 0.02 Pa: **8.9002e-06 m/s**.
 Volume flow rate: 0.0089002 l/s per unit area.
 Reynolds number 9.14266e-07 for characteristic length 1.56475e-06 m.
 Flow resistivity: 7.49046e+06 kg/(m²s).
 Gurley value: 0.287031 s.
 Asymptotic estimate of permeability: 2.44558e-12 m².
 --- Solver: LIR, Iterations: 250, Runtime: 15.231 s, Number of Cells: 2114135,
 Memory usage: 365.868 MiB,
 and stopped successfully for **error bound** ---

Plot Options

--- Total runtime: 15.593 s, Total memory usage: 1.031 GiB ---

When the user enters different values for the media thickness and clicks **Apply**, the permeability and flow resistivity in the **Results-Report** and the **Results-Map** is changed according to the new length for each direction.

Another post-processing option is the **Permeability Unit** which can be changed to **m²**, **mDarcy**, or **Darcy**. When the user changes the unit and clicks **Apply**, the permeability tensor is converted to the new unit.

For **Characteristic Length** four possibilities can be chosen: **Numerical Length**, **Max. Pore Diameter**, **Max. Fiber/Particle Diameter**, and **Manually specified**. If **Manually specified** is selected, a characteristic length has to be specified below. Further information to the **Characteristic Length** is given below on page [44](#).

Note that after clicking Apply, the result file is changed and is not the original result file produced by the simulation!

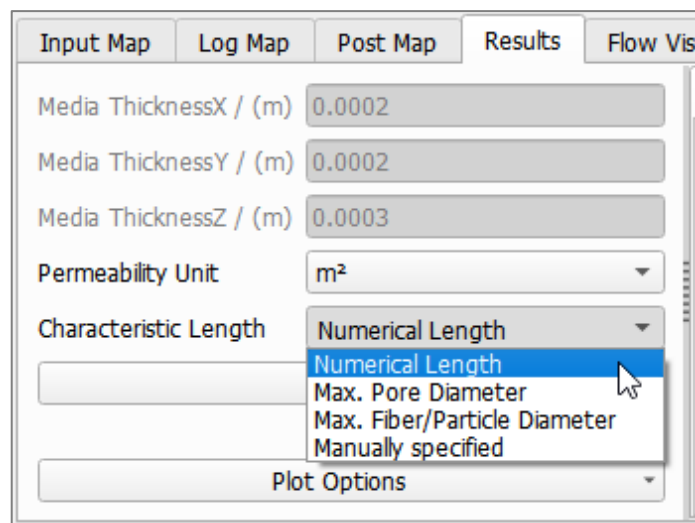
Characteristic Length for Reynolds Number

In situations where high velocity occurs, non-Darcy behavior happens, that is important to describe fluid flow in porous media. To identify the beginning of non-Darcy flow, one of the criteria is the Reynolds number, Re , which is defined as

$$Re = \frac{\rho u L}{\mu}$$

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 ($\text{Pa}\cdot\text{s}$ or $\text{N}\cdot\text{s/m}^2$ or $\text{kg/m}\cdot\text{s}$), 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:



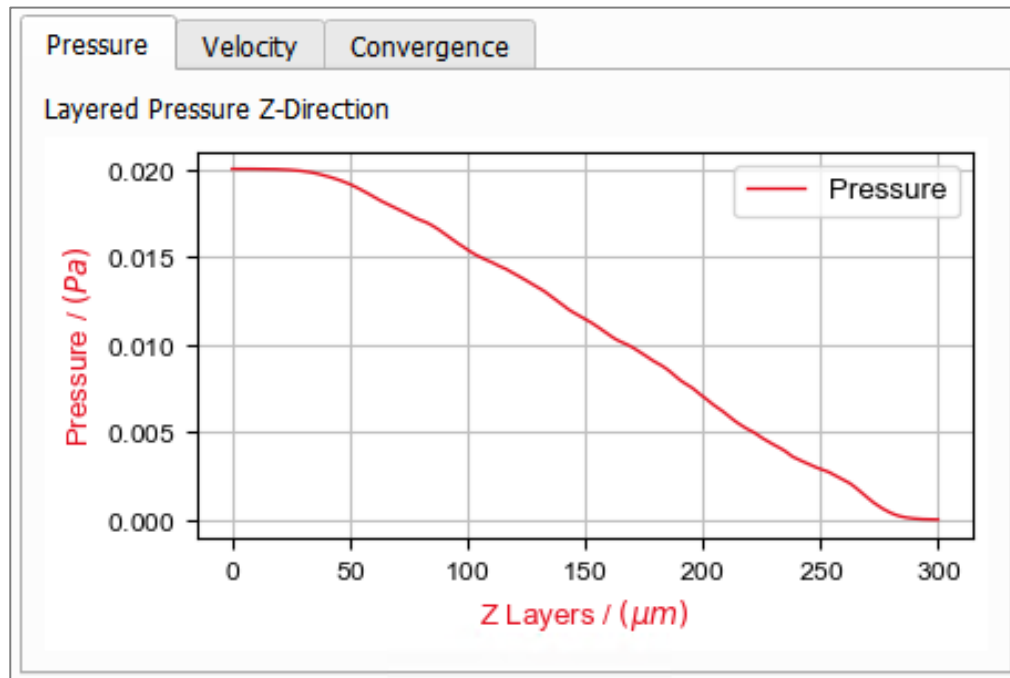
1. **Numerical Length.** Since permeability is expressed in units of m^2 , the square root of the permeability can be in length units, $L = \sqrt{\kappa}$. [4] This option is used for Stokes by default.
2. **Maximum Pore Diameter.** The characteristic length can be the diameter of the largest through pore. From structure analysis the maximum pore diameter is found. This option is used for Navier-Stokes by default.
3. **Maximum Fiber/Particle Diameter.** The characteristic length can also be the largest fiber diameter for fibrous media and the largest particle size for grain structures. From structure analysis, the largest fiber/particle diameter is found.
4. **Manually specified.** When none of the previous three options fit the users' definition, the characteristic length can be manually specified. The **Specify Characteristics Length: / (m)** must be entered when this option is chosen.

Specify Char. Length / (m) 1

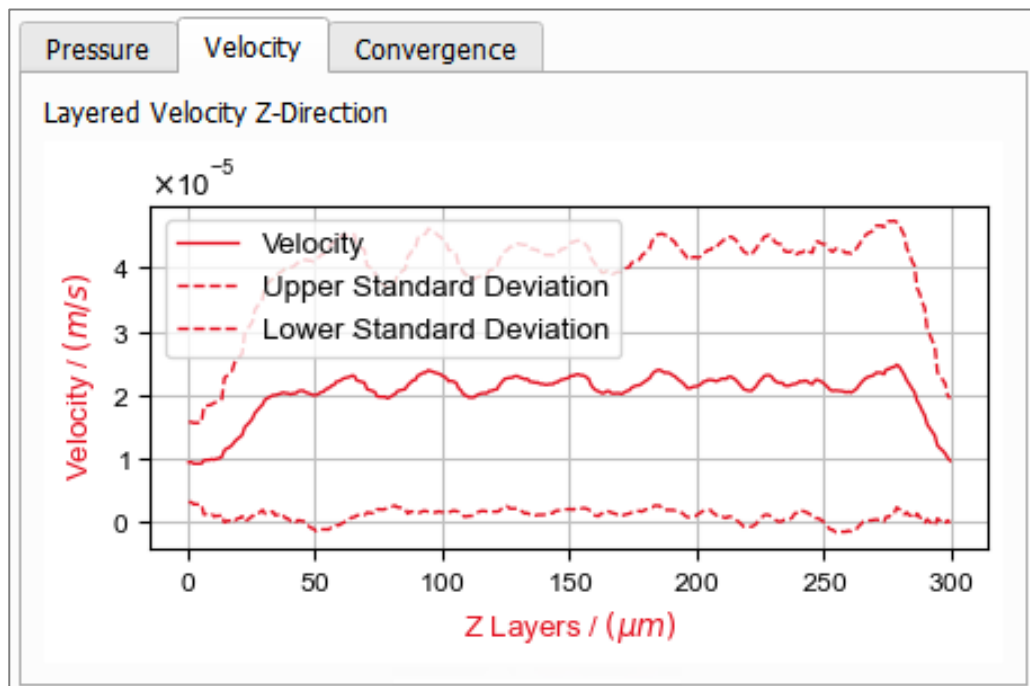
Observe the changes under the **Results – Report** subtab in the Result Viewer after clicking **Apply**. The Reynolds number and the Characteristic Length are updated.

RESULTS - PLOTS

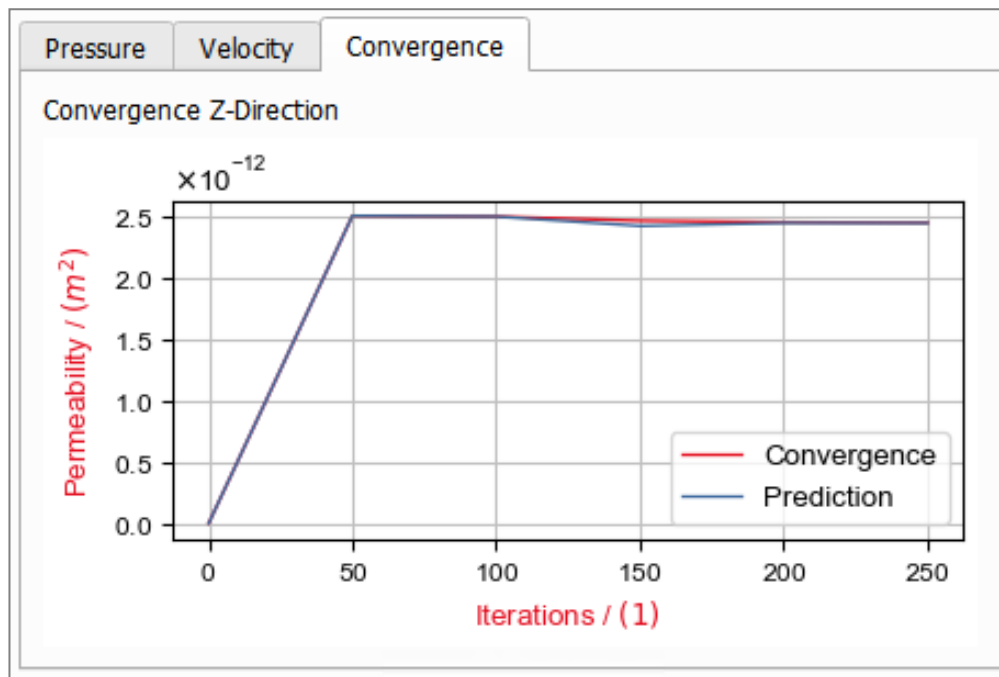
Under the **Pressure** tab, a graph of the changes in **Pressure** across the layers is shown. There is one plot for each direction previously chosen in the **Options** dialog (see page [17](#)). Here, only the Z-direction had been chosen for computations.



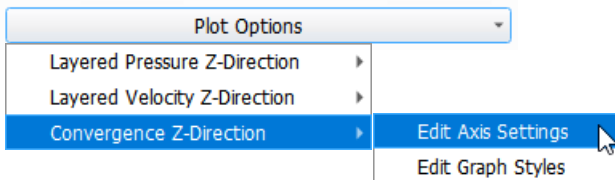
Under the **Velocity** tab, a graph of the changes in **Velocity** across the layers is shown. Here, the velocity component in flow direction is shown. A graph is generated for each calculated direction. The velocity component in the chosen direction is averaged over the pore voxels for each layer in this direction.



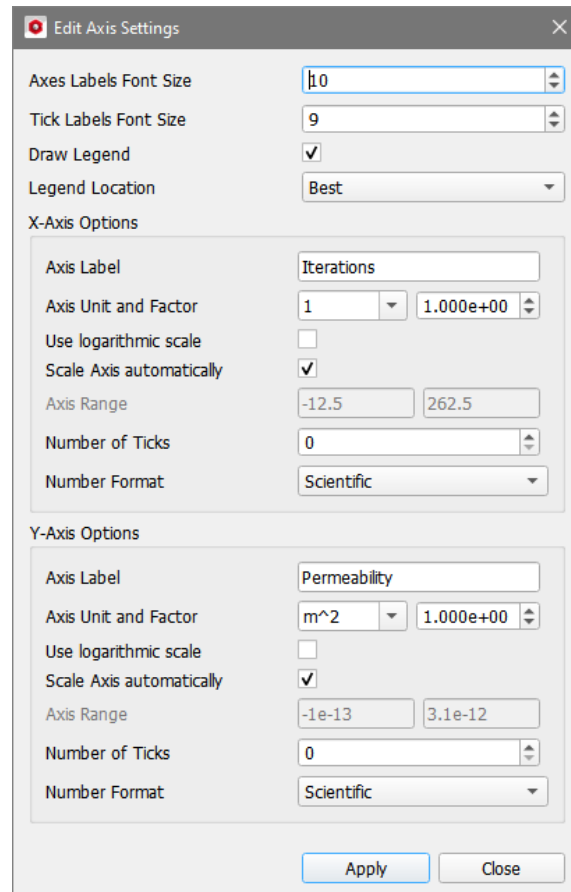
The **Plots** tab also includes a **Convergence** tab with a chart of the **Permeability** at given **Iteration** values. See the convergence graphs of the iterative solver for each of the calculated directions.



As usual, right clicking on the plot opens a small dialog where to choose to **Edit Axis Settings**. Also, this dialog can be opened by selecting the plot and then Edit Axis Settings from the **Plot Option** pull-down menu in the **Post-Processing** panel on the left.

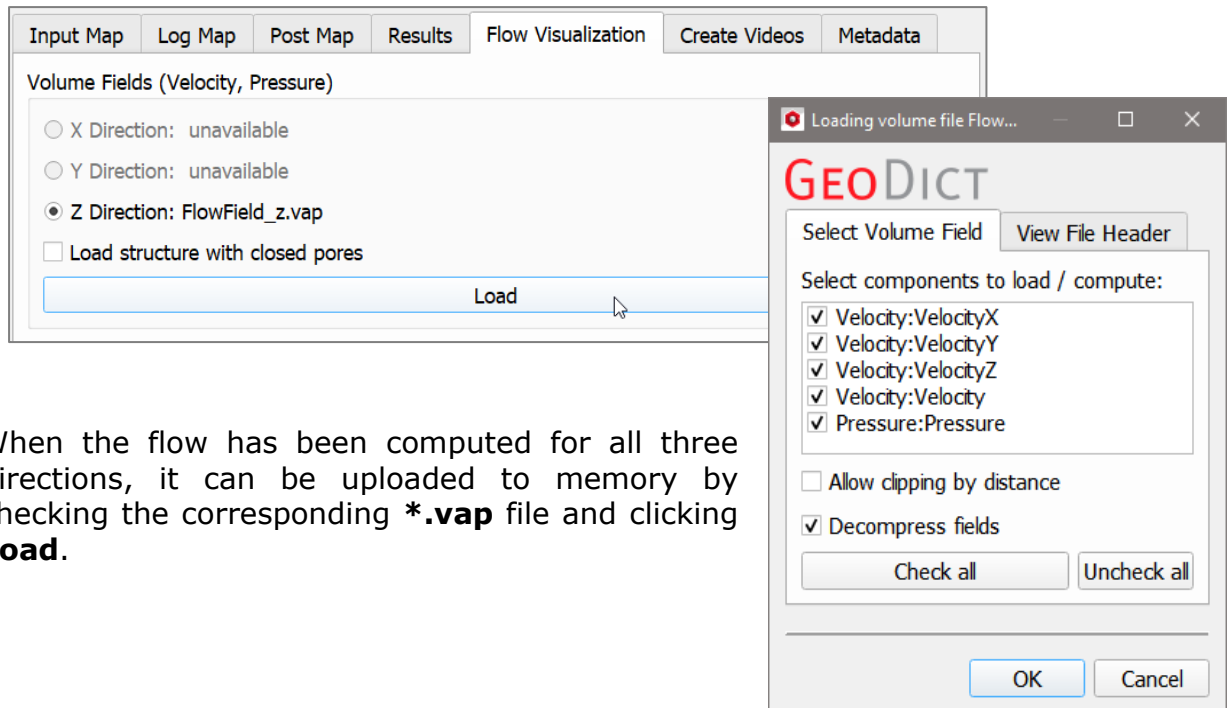


The scale of the X- and Y-axis can be manually entered to compare Convergence, Velocity/Layers or Pressure/Layers graphs from different permeability computations on the same or various structure models.



FLOW VISUALIZATION

Under the **Flow Visualization** tab, the results of the flow simulation can be loaded for graphical visualization in 2D-Cross section view or 3D-Rendering.



When the flow has been computed for all three directions, it can be uploaded to memory by checking the corresponding ***.vap** file and clicking **Load**.

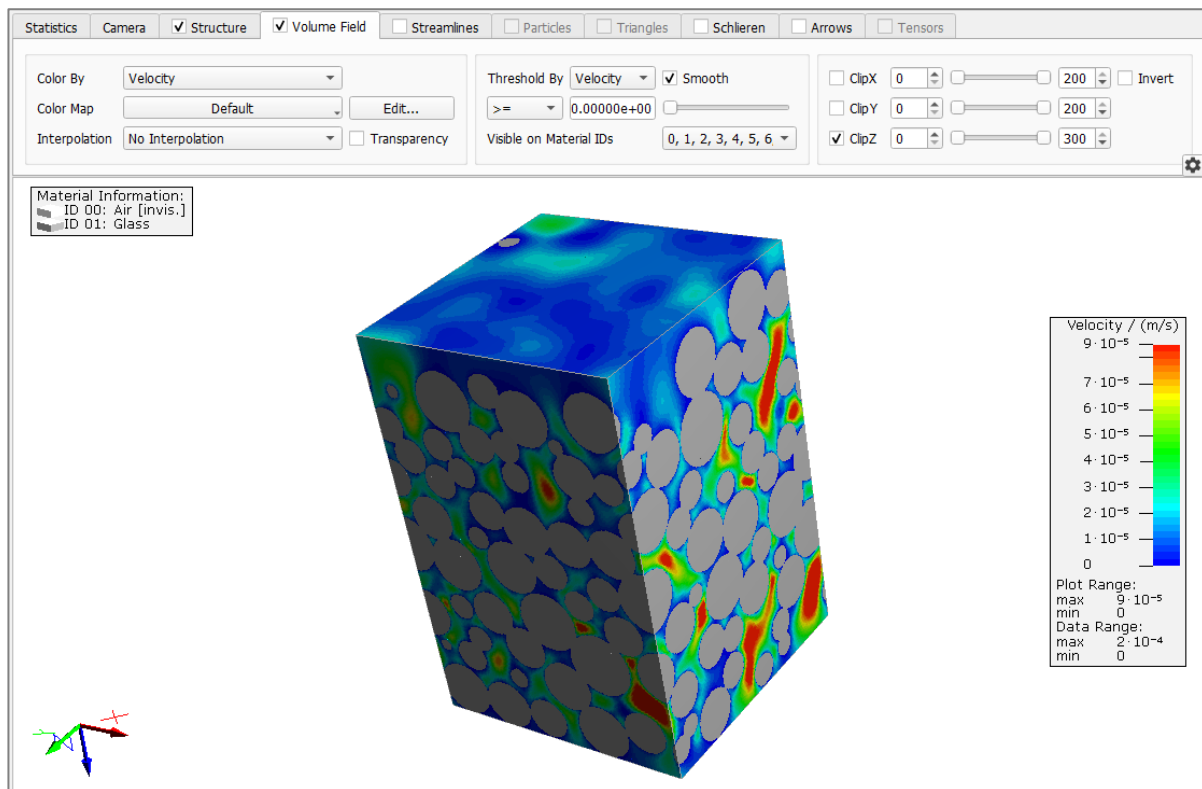
Choose the flow field components: VelocityX, VelocityY, VelocityZ, Velocity and/or Pressure. The directions that were not selected are not computed, and, thus, are unavailable.

By checking **Decompress fields**, the flow field is decompressed to a regular grid instead of remaining compressed. The flow field was stored with compression (computed with the **LIR** solver) and keeping it compressed requires 80-90% less memory for visualizations compared to decompressed flow fields. However, not all visualization features are supported for compressed flow fields.

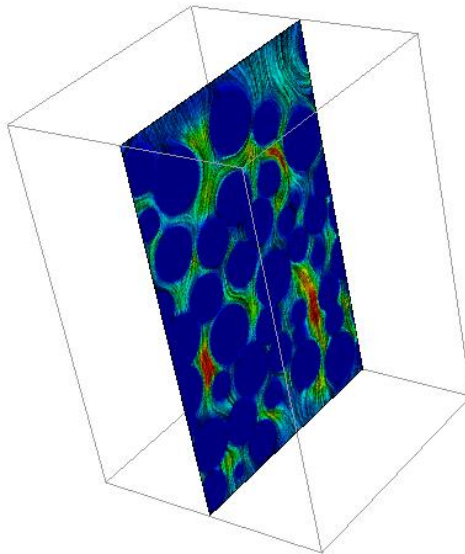
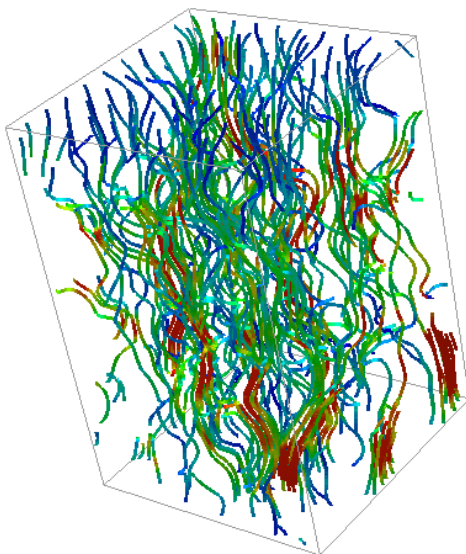
The original structure that the solver used internally is also loaded if **Load structures with closed pores** is checked. This structure also shows pores that have been closed due to prior geometry analysis.

Clicking **Load** prepares the results of the flow simulation for visualization, but also **File → Load Volume Field...** can be selected in the Menu bar.

The initial display is carried out using the default values, but the visualization of the flow field can be adjusted through the parameters under the **Volume Field** tab, in the **Visualization panel**, above the Visualization area.



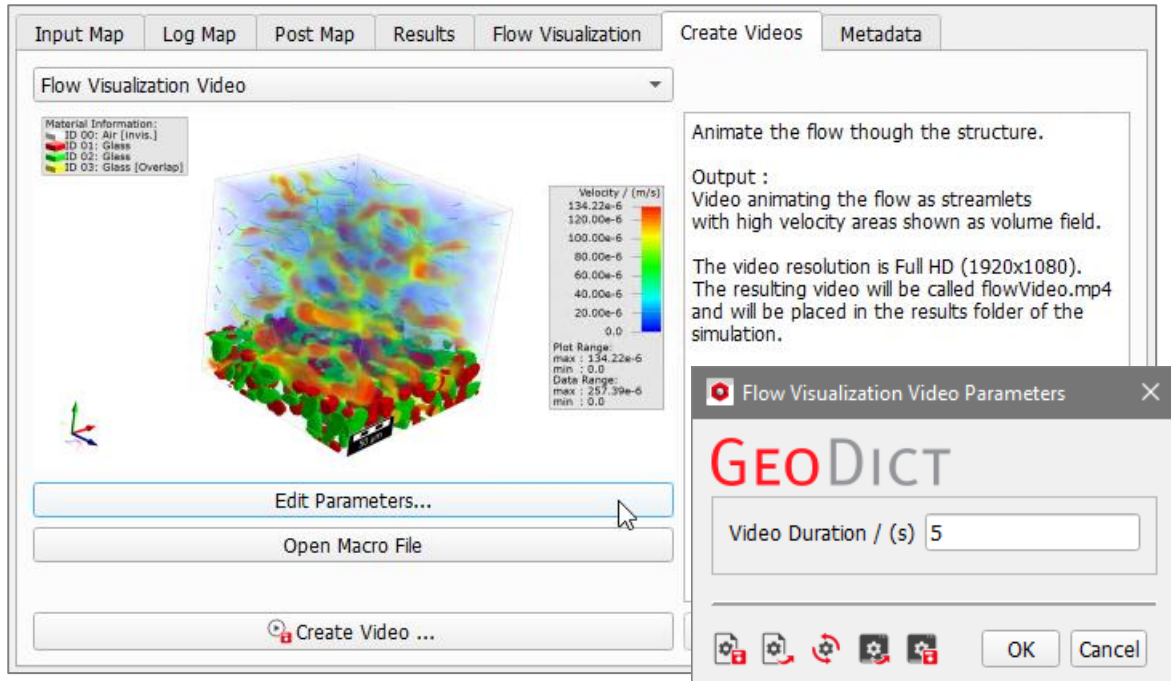
There are many possibilities to visualize the results. For example, visualize streamlines or Schlieren.



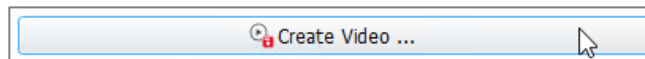
More information about the visualization parameters is available in the [Visualization handbook](#) of this User Guide.

CREATE VIDEOS

For the solvers EJ, SimpleFFT and LIR also the **Create Videos** tab is available. It provides a predefined Python macro to create a video of the structure with animated flow streamlets with high velocity areas shown as volume field. Clicking **Edit Parameters** to change the **Video Duration** in the **Flow Visualization Video Parameters** dialog.



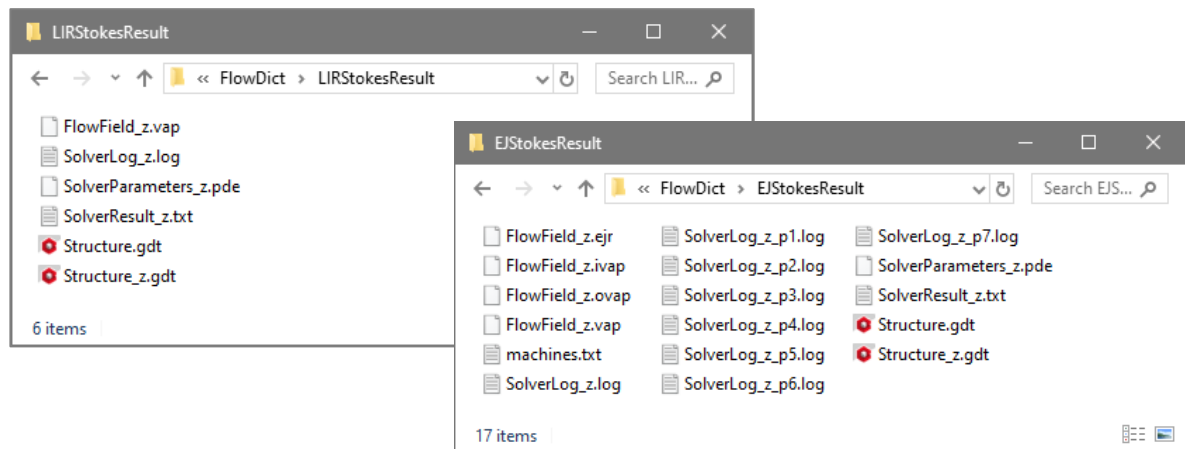
Click **Create Video** to start the video generation. The video is placed in the result folder.



Find out more about the options in this tab in the [Result Viewer handbook](#) of this User Guide.

RESULT FOLDER FROM EJ, SIMPLEFFT, AND LIR SOLVER

The result folder, saved in the selected project folder, contains the following file formats:



- ***.gdt:** GeoDict 3D compressed file format for the description of the geometry (e.g. Structure_z.gdt or Structure.gdt)
- ***.log:** diary of the solver computing flow properties for each computed direction (e.g. SolverLog_z.log). When using MPI parallelization, the *_p1.log, *_p2.log, and *_p3.log contain the information for each processor running the computations.
- ***.pde:** partial differential equation file format, with input parameters for the solvers (e.g. SolverParameters_z.pde)
- ***.ejr:** files for the restart of computations for each computed direction (only for EJ)
- ***.txt:** Text document format (e.g. SolverResult_z.txt)
- ***.vap:** visualization file format (e.g. FlowField_z.vap). The *.ivap and *.ovap files contain data about the inflow/outflow region and are only present if periodic with inflow/outflow regions was selected in the boundary conditions tab
- When using parallelization, a machine file is present with the name machines.txt that contains the name of the compute nodes where the computations took place.

RESULT FILE (.GDR) FROM FORCHHEIMER APPROXIMATION

The Result Viewer of a result file obtained from running the **Forchheimer Approximation** is shown here as an example.

The following input parameters were used:

GEO DICT
Result File Name (*.gdr) ForchheimerResult.gdr

Measured Experiment

Thickness of measured media: 1 mm

Density of measured media: 1.204 kg/m³

Viscosity of measured media: 1.834e-5 kg/ms

☒ Extrapolate from given Velocity/Pressure pairs
☐ Extrapolate from .gdr result files

Number of Parameters: 2

	Velocity / (m/s)	PressureDrop / (Pa)
1	2	10
2	4	30

Predicted Experiment

Thickness of predicted media: 1 mm

Density of predicted media: 1.204 kg/m³

Viscosity of predicted media: 1.834e-5 kg/ms

☒ Use Measured Experiment
☐ Air Permeability (ISO 9237)
☐ Air Permeability (ASTM D737)
☒ Pressure Drop: 150 Pa
☐ Mean Velocity: 10 m/s
☐ Flow Rate: 6000 l/min on Flow Area 100 cm²

OK Cancel

The calculation is very fast because no partial differential equation is solved.

After the computation has finished, the result file (.gdr) is saved in the project folder and a Result Viewer of the result file automatically opens.

Fri Aug 27 2021 (2022 Build 51842) .../FlowDict/ForchheimerResult.gdr

Input Map Log Map Post Map **Results** Metadata

X-Y axis
☒ Pressure drop - Velocity
☐ Velocity - Pressure drop
 Pressure Unit: Pa Apply ...

Report Plots Map

Forchheimer Equation for measured Experiment:
 $\Delta P = 2.5 u + 1.25 u^2$
 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 $\kappa = \mu * L / f_1 = 1.834e-05 * 0.001 / 2.5 = 7.336e-09$,
 Inertial permeability $\kappa_1 = \rho * L / f_2 = 1.204 * 0.001 / 1.25 = 0.0009632$,

Forchheimer Equation for predicted Experiment:
 $\Delta P = 2.5 u + 1.25 u^2$
 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 $\kappa = \mu * L / f_1 = 1.834e-05 * 0.001 / 2.5 = 7.336e-09$,
 Inertial permeability $\kappa_1 = \rho * L / f_2 = 1.204 * 0.001 / 1.25 = 0.0009632$,

The estimation is 10 m/s for 150 Pa.

Air permeability : 11.6886 m/s for 200 Pa according to ISO 9237:1995(E).
Air permeability : 9.04988 m/s for 125 Pa according to ASTM D737 - 04 (2016).
 or 904.988 cm³/s/cm², or 1781.47 ft³/min/ft².

--- Total runtime: 106 ms, Total memory usage: 1.185 GiB ---

Manage Data Load Input Map Export Close

The (collapsible and expandable) header section box shows the path and the **Result File Name** previously entered in the **Options** dialog. This result file is accessible at any time by selecting **File** → **Open Results (*.gdr)...** from the Menu bar.

The computational results are accessed through tabs in the Result Viewer. They follow the same pattern as those for the other FlowDict solvers.

The **Results - Report** tab shows the approximation of Forchheimer's equation that fits the entered velocity / pressure drop pairs.

The first coefficient approximates:

$$\frac{\mu}{K} = 2.5$$

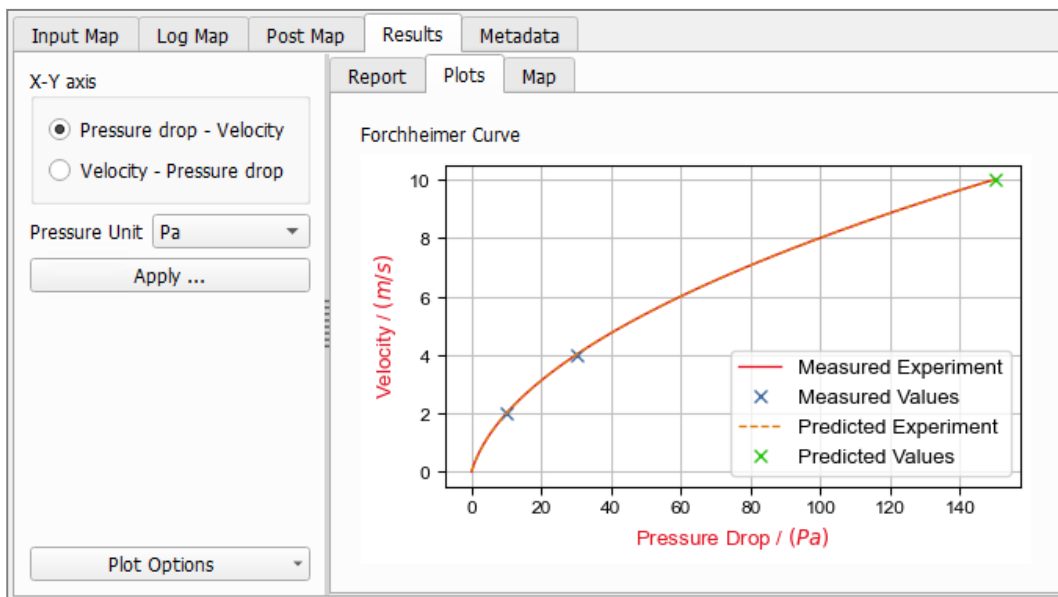
The second coefficient approximates:

$$\frac{\rho}{k_2} = 1.25$$

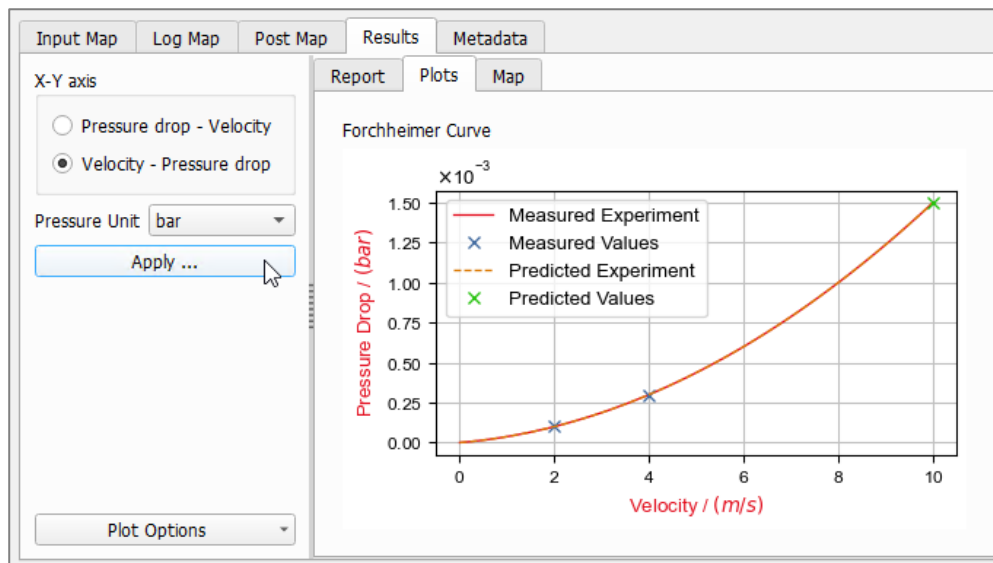
In this example, the estimation of the velocity for the given **Experiment Input** and a pressure drop of $\nabla p = 150$ is

$$\vec{u} = 10 \text{ m/s}$$

The **Results - Plots** subtab shows the computed quadratic polynomial in a graph. The velocity / pressure drop pairs (2/10 and 4/30) entered in the **Forchheimer Approximation Options** dialog are highlighted on the curve with a blue X.



The graph can also be plotted with interchanged axes by checking **Velocity – Pressure drop** and clicking **Apply ...** on the post-processing settings section of the Result Viewer. Also, the **Pressure Unit** can be changed from **Pa** to **bar**.



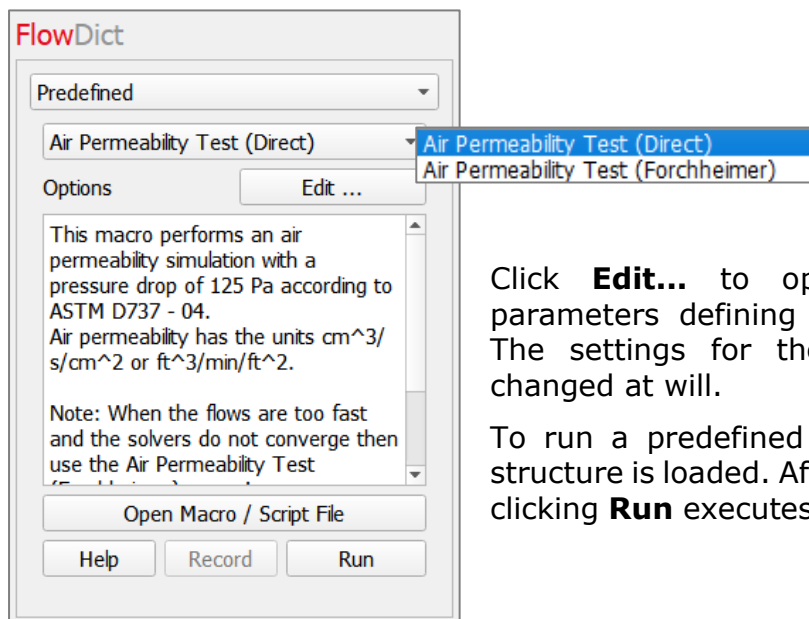
PREDEFINED

When **Predefined** is selected in the **FlowDict** section, representative flow simulation examples can be chosen from the pull-down menu in the **Predefined** panel.

Currently, two simulations are predefined:

- Air Permeability Test (Direct)
- Air Permeability Test (Forchheimer).

When predefined simulations are run, **GeoDict** macros corresponding to the simulations are called and executed. These macros are available in the **FlowDict** folder in the **GeoDict** installation folder. They can be opened with a text editor, to observe their syntax and the steps involved in the simulation and can also be edited.



Click **Edit...** to open a dialog with the parameters defining these permeability tests. The settings for these parameters may be changed at will.

To run a predefined simulation, make sure a structure is loaded. After modifications are done, clicking **Run** executes the corresponding macro.

A result file (.gdr) and a folder with the same name, containing the result files, are automatically saved inside the project folder. The .gdr file can be opened in **GeoDict** through **File** → **Open Results (*.gdr) ...** in the menu bar.

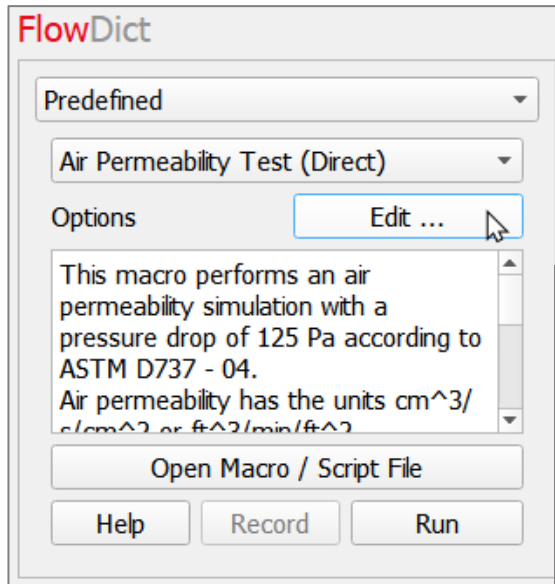
After opening the .gdr file, the user can directly access all parameters used for the simulation by clicking the **Load Input Map** button at the bottom, and then selecting the corresponding solver (**Navier-Stokes(-Brinkman)** or **Forchheimer Approximation**) in the FlowDict section pull-down menu, and clicking the **Solver Options' Edit...** button. In this way, all parameter values used for the simulation are loaded into the corresponding **Solver Options** dialog and can be examined in detail.

By clicking the **Open Macro / Script File** button, the macro file containing all steps for the predefined simulations can be accessed.

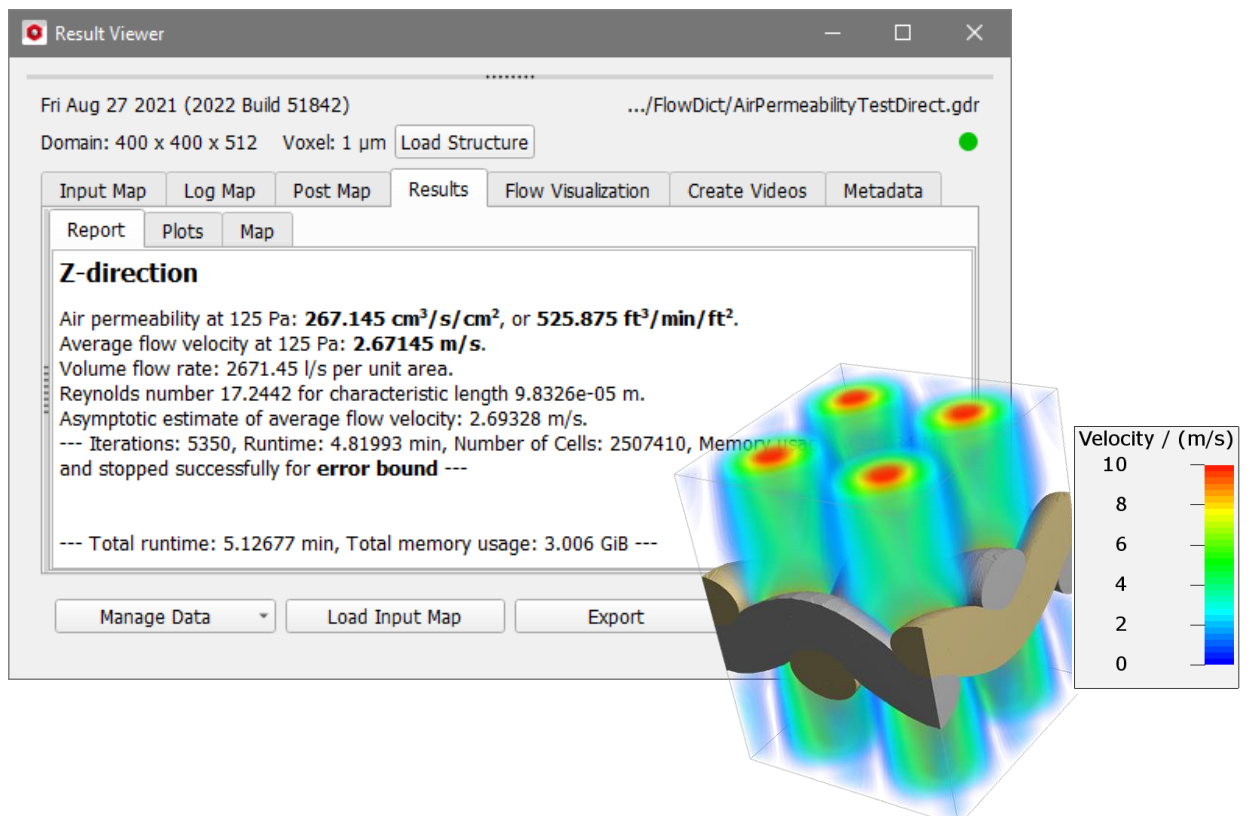
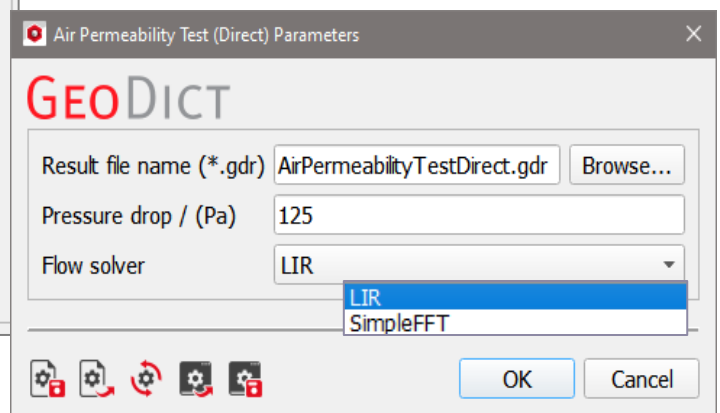
AIR PERMEABILITY TEST (DIRECT)

Air permeability is an important factor in the performance of textile materials, e.g. gas filters or fabrics for air bags.

The first predefined macro performs an air permeability test on the loaded structure. Clicking **Edit ...** the result file name and the pressure drop can be edited. The permeability is computed using **Navier-Stokes(-Brinkman)**, but for the solver LIR or SimpleFFT can be selected.



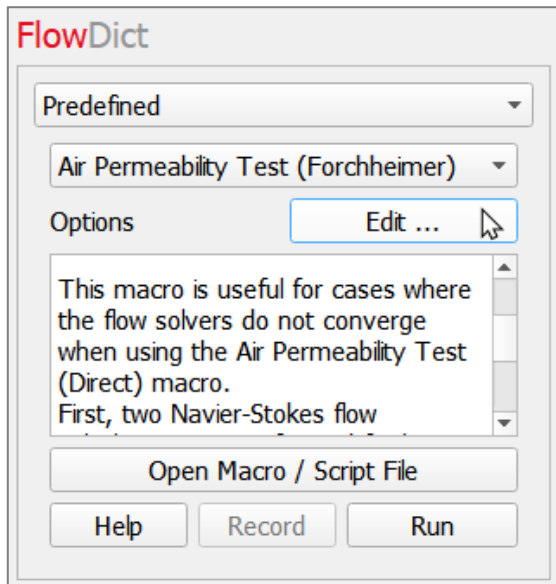
Using the default settings, an air permeability test with a pressure drop of 125 Pa is done for the current structure, according to ASTM D737 - 04.



AIR PERMEABILITY TEST (FORCHHEIMER)

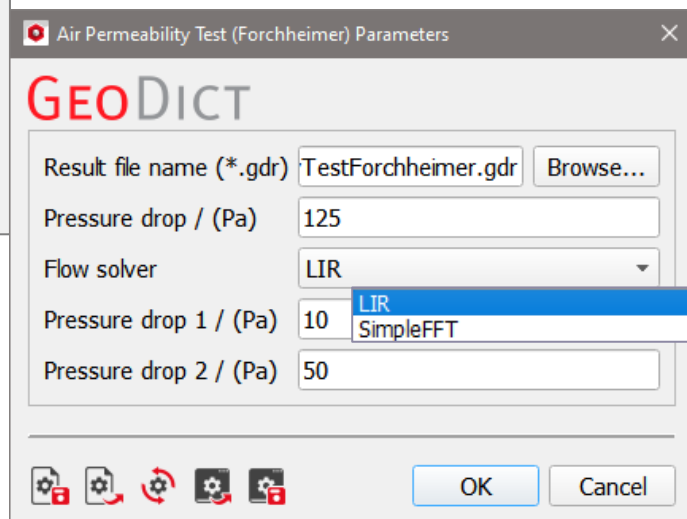
For fast flows the first predefined simulation might not converge. In this case, the **Air Permeability Test (Forchheimer)** can be very useful.

The macro runs two air permeability computations with low pressure drops, using LIR or SimpleFFT. From the results, a **Forchheimer Approximation** is done for the defined pressure drop.



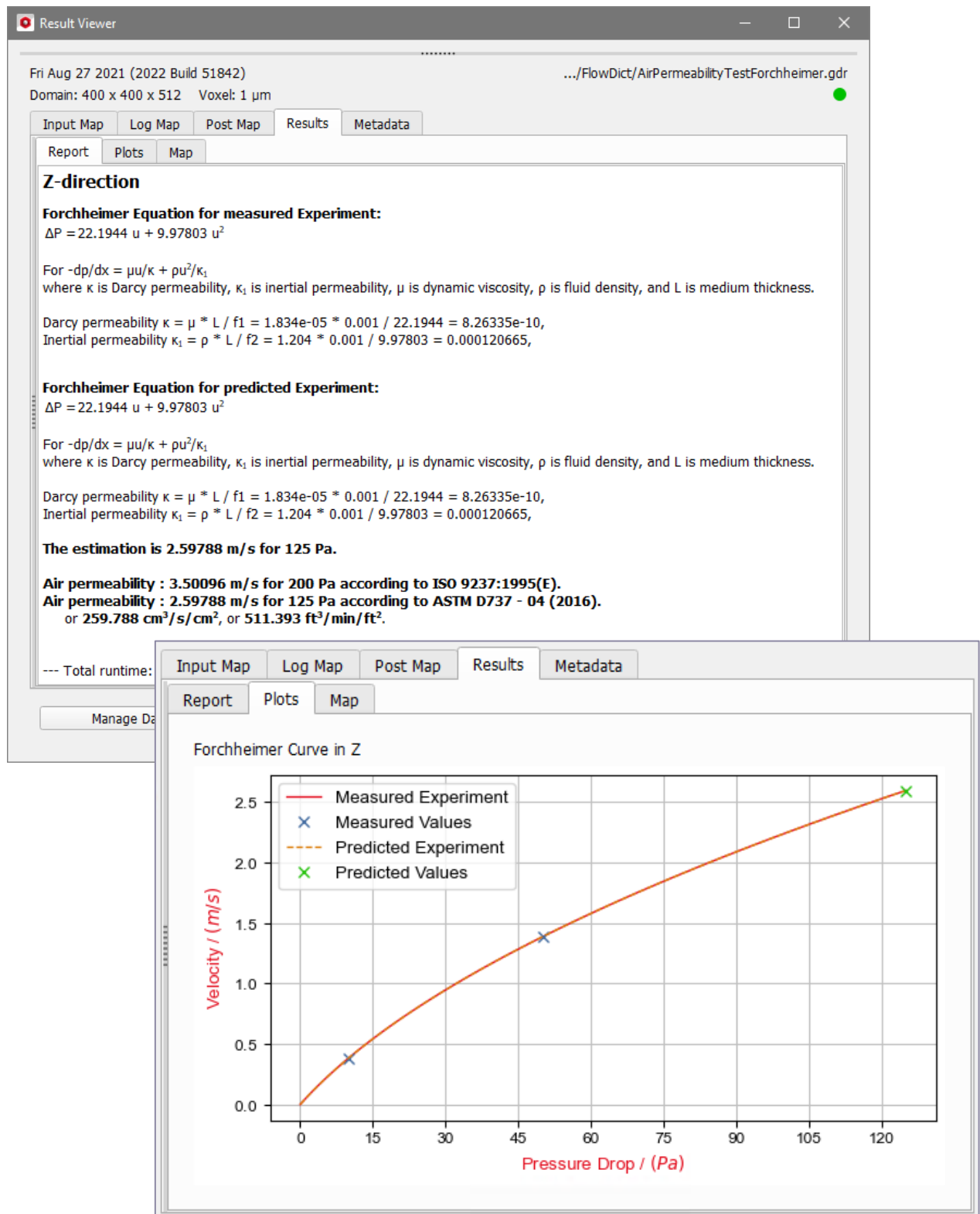
After clicking **Edit ...**, the result file name and pressure drop for the two simulations and the prediction can be changed.

For the two simulations the permeability is computed using **Navier-Stokes(-Brinkman)** but for the solver LIR or SimpleFFT can be selected.



This predefined macro creates three result files and two corresponding result folders. The Forchheimer result file keeps the entered Result File Name.

For the flow simulations the result file name is automatically expanded by the values for Pressure drop 1 (here, 10 Pa) and Pressure drop 2 (here, 50 Pa), respectively.



REFERENCES

- [1] Wiegmann et al.: Computer Aided Engineering of Filter Materials and Pleated Filters. Global Guide of the Filtration and Separation Industry by E. von der Luehe. VDL – Verlag, pp 191-198 (2010).
- [2] Mattila et al.: A prospect for computing in porous material research: very large fluid flow simulations, Journal of Computational Science, 15, pp. 62-76 (2016).
- [3] Saxena et al.: References and benchmarks for pore-scale flow simulated using micro-CT images of porous media and digital rocks, Advances in Water Resources, 109, pp. 211-235 (2017).
- [4] Zeng et al.: 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>.

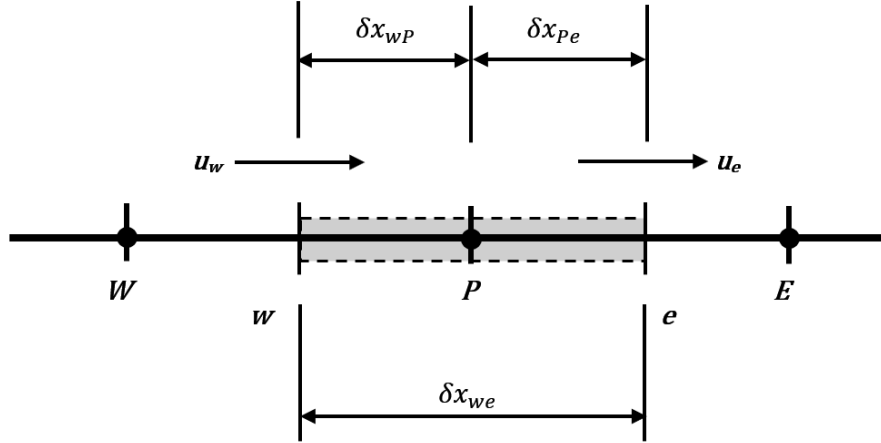
APPENDIX:

THE DISCRETIZATION FOR CONVECTION-DIFFUSION PROBLEMS

Considering the steady convection-diffusion equation:

$$\text{div}(\rho \mathbf{u} \phi) = \text{div}(\Gamma \text{grad } \phi) + S_\phi$$

in a one-dimensional domain:



when the sources are absent, the one-dimensional flow field u is:

$$\frac{d}{dx}(\rho u \phi) = \frac{d}{dx}\left(\Gamma \frac{d\phi}{dx}\right)$$

where Γ is the diffusion coefficient.

Integration of transport equation over the control volume gives:

$$(\rho u A \phi)_e - (\rho u A \phi)_w = \left(\Gamma A \frac{d\phi}{dx}\right)_e - \left(\Gamma A \frac{d\phi}{dx}\right)_w$$

Define the convective mass flux per unit area, F , and diffusion conductance at cell faces, D :

$$F = \rho u, \quad D = \frac{\Gamma}{\delta x}$$

with

$$F_w = (\rho u)_w, \quad F_e = (\rho u)_e, \quad D_w = \frac{\Gamma_w}{\delta x_{WP}}, \quad D_e = \frac{\Gamma_e}{\delta x_{PE}}$$

Assuming $A_e = A_w = A$ and the central differencing approach, the transport equation is:

$$F_e \phi_e - F_w \phi_w = D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W)$$

THE CENTRAL DIFFERENCING SCHEME

If the central differencing scheme is used to represent the diffusion term and use linear interpolation to compute the cell face values for the convective terms,

$$\phi_e = \frac{(\phi_P + \phi_E)}{2} \text{ and } \phi_w = \frac{(\phi_P + \phi_W)}{2}$$

The transport equation becomes

$$\frac{F_e}{2}(\phi_P + \phi_E) - \frac{F_w}{2}(\phi_W + \phi_P) = D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W)$$

re-arranging,

$$\left[\left(D_w + \frac{F_w}{2} \right) + \left(D_e - \frac{F_e}{2} \right) + (F_e - F_w) \right] \phi_P = \left(D_w + \frac{F_w}{2} \right) \phi_W + \left(D_e - \frac{F_e}{2} \right) \phi_E$$

Therefore, in central differencing expression:

$$a_P \phi_P = a_W \phi_W + a_E \phi_E$$

where

$$a_W = \left(D_w + \frac{F_w}{2} \right)$$

$$a_E = \left(D_e - \frac{F_e}{2} \right)$$

$$a_P = a_W + a_E + (F_e - F_w)$$

THE UPWIND DIFFERENCING SCHEME

The upwind differencing scheme takes into account the flow direction when determining the value at a cell face: the convected value of ϕ is taken to be equal to the value at the upstream node.

For the general form

$$a_P \phi_P = a_W \phi_W + a_E \phi_E$$

the coefficients become

	a_W	a_E
$F_w > 0, F_e > 0$	$D_w + F_w$	D_e
$F_w < 0, F_e < 0$	D_w	$D_e - F_e$

or

a_w	a_E
$D_w + \max(F_w, 0)$	$D_e + \max(0, -F_e)$

THE POWER-LAW SCHEME

Define the non-dimensional cell Peclet number as a measure of the relative strengths of convection and diffusion:

$$Pe = \frac{F}{D} = \frac{\rho u}{\Gamma \delta_x}$$

where δ_x is characteristic length (voxel length).

With the power-law scheme, diffusion is set to zero when voxel Pe exceeds 10.

If $0 < Pe < 10$, the flux is evaluated by using a polynomial expression. The net flux per unit area at the west control volume face:

$$q_w = F_w [\phi_w - \beta_w (\phi_P - \phi_w)] \quad \text{for } 0 < Pe < 10$$

$$q_w = F_w \phi_w \quad \text{for } Pe > 10$$

where

$$\beta_w = (1 - 0.1Pe_w)^5 Pe_w$$

The coefficients are given for

$$a_P \phi_P = a_W \phi_W + a_E \phi_E$$

as

a_w	a_E
$D_w \max [0, (1 - 0.1 Pe_w ^5)] + \max[F_w, 0]$	$D_e \max [0, (1 - 0.1 Pe_e ^5)] + \max[-F_e, 0]$

References

- H.K. Versteeg and W. Malalasekera, An Introduction to Computational Fluid Dynamics: The Finite Volume Method, Pearson, Prentice Hall, 2nd edition, 2007.
- S.V. Patankar, Numerical heat transfer and fluid flow. Hemisphere Publishing Corporation, New York, 1980.

Technical
documentation:

Janine Hilden
Liping Cheng
Sven Linden
Barbara Planas

MATH
2 MARKET

Math2Market GmbH

Richard-Wagner-Str. 1, 67655 Kaiserslautern, Germany
www.geodict.com