The OpenFOAM (Open Field Operation and Manipulation) CFD Toolbox can simulate anything from complex fluid flows involving chemical reactions, turbulence and heat transfer, to solid dynamics, electromagnetics and the pricing of financial options. OpenFOAM is produced by OpenCFD Ltd, is freely available and open source, licensed under the GNU General Public License.

OpenFOAM is supplied with numerous pre-configured solvers, utilities and libraries and so can be used like any typical simulation package. However, it is open, not only in terms of source code, but also in its structure and hierarchical design, so that its solvers, utilities and libraries are fully extensible.

OpenFOAM uses finite volume numerics to solve systems of partial differential equations ascribed on any 3D unstructured mesh of polyhedral cells. The fluid flow solvers are developed within a robust, implicit, pressure-velocity, iterative solution framework, although alternative techniques are applied to other continuum mechanics solvers. Domain decomposition parallelism is fundamental to the design of OpenFOAM and integrated at a low level so that solvers can generally be developed without the need for any ‘parallel-specific’ coding. (www.opencfd.co.uk/openfoam).

OpenFOAM is produced by OpenCFD Ltd, is freely available and open source, licensed under the GNU General Public Licence. It is therefore important to have a calculating tool that can handle very hard CFD simulations in reasonably execution time. In the last years computing power of processors has reached a technological limit and then the technological solution adopted to solve these CFD problems is to divide the work among parallel processors (parallel working).

For a CFD solver therefore it is important to be scalable or, in other words, to have a significant improvement in performance as the number of processors used to solve the problem. On numerical fluid dynamics, computational cost plays a crucial role. Many phenomena can’t be treated with approximate numerical methods but by others more complex, such as RANS (Reynolds Averaged Navier Stokes) for example, requiring the adoption of resolution meshes need a large amount of computational calculation.

The Fluent implementation of the cavity case was converted in its OpenFOAM equivalent one. To make an automatic conversion of the mesh we used the utility fluentMeshToFoam provided by OpenFOAM while manual changes were necessary to convert settings of boundary conditions, initial conditions, physical characteristics of the fluid (viscosity etc.) and the characteristic parameters of the numerical simulation. The case obtained by this conversion was found to be fully equivalent to the original case provided by the OpenFOAM tutorial. We are now studying the capabilities and limitations for converting to OpenFOAM CFD cases made by Fluent. The goal is to make this procedure as much as possible robust, efficient and automatic on the ENEA-GRID.

OpenFOAM parallel performances

On numerical fluid dynamics, computational cost plays a crucial role. Many phenomena can’t be treated with approximate numerical methods but by others more complex, such as RANS (Reynolds Averaged Navier Stokes) for example, requiring the adoption of resolution meshes need a large amount of computational calculation. It is therefore important to have a calculating tool that can handle very hard CFD simulations in reasonably execution time. In the last years computing power of processors has reached a technological limit and then the technological solution adopted to solve these CFD problems is to divide the work among parallel processors (parallel working).

For a CFD solver therefore it is important to be scalable or, in other words, to have a significant improvement in performance as the number of processors used to solve the problem.

The Fluent implementation of the cavity case was converted in its OpenFOAM equivalent one. To make an automatic conversion of the mesh we used the utility fluentMeshToFoam provided by OpenFOAM while manual changes were necessary to convert settings of boundary conditions, initial conditions, physical characteristics of the fluid (viscosity etc.) and the characteristic parameters of the numerical simulation. The case obtained by this conversion was found to be fully equivalent to the original case provided by the OpenFOAM tutorial. We are now studying the capabilities and limitations for converting to OpenFOAM CFD cases made by Fluent. The goal is to make this procedure as much as possible robust, efficient and automatic on the ENEA-GRID.