

Application of computational fluid dynamics for the optimization of homogenization processes in wine tanks

Jonas Müller and Kai Velten

Hochschule Geisenheim University, Modeling and Simulation Workgroup, 65366 Geisenheim, Germany

Abstract. Mixing processes for modern wine-making occur repeatedly during fermentation (e.g. yeast addition, wine fermentation additives), as well as after fermentation (e.g. blending, dosage, sulfur additions). In large fermentation vessels or when mixing fluids of different viscosities, an inadequate mixing process can lead to considerable costs and problems (inhomogeneous product, development of layers in the tank, waste of energy, clogging of filters). Considering advancements in computational fluid dynamics (CFD) in the last few years and the computational power of computers nowadays, most large-scale wineries would be able to conduct mixing simulations using their own tank and agitator configurations in order to evaluate their efficiency and the necessary power input based on mathematical modeling. Regardless, most companies still rely on estimations and empirical values which are neither validated nor optimized. The free open-source CFD software OpenFOAM (v.2.3.1) is used to simulate flows in wine tanks. Different agitator types, different propeller geometries and rotational speeds can be modeled and compared amongst each other in the process. Moreover, fluid properties of different wine additives can be modeled. During optical post-processing using the open-source software ParaView (v.4.3) the progression of homogenization can be visualized and poorly mixed regions in the tank are revealed.

1. Introduction

Computational fluid dynamics uses various numerical methods to describe a wide range of problems concerning temperature, velocity and other properties in fluids. Despite its broad acceptance and widespread use across many scientific fields and engineering disciplines, CFD is used very little in the food industry [3] and not yet at all in the wine industry. Even if an awareness for the great potential value of CFD in wineries existed, inadequate knowledge of CFD methods and presumed costs for software might deter larger wineries of implementing CFD analyses. Industrial applications for which CFD could be useful in wineries include temperature and humidity controlled barrel rooms [1], cooling and heating systems of wine tanks [2], cold storage rooms [4], winery equipment and processes (e.g. filters, pumps, and flotation devices), and homogenization processes in the tank. The benefits of CFD analyses in wineries include better designs before installation of costly equipment, predicting design alterations when equipment is already in place, and finding the best possible combination of variables to ensure optimum performance of existing equipment or use of existing facilities. This study aims to reveal the real cost and potential benefit of CFD analyses in the wine industry using mechanical mixing in wine tanks as an example.

Another goal of this research is to better understand the mechanical mixing processes during winemaking and the resulting potential for savings with regards to the power input and the avoidance of problems in large wine tanks.

1.1. Mixing in wineries

Mixing is the distribution of at least two types of mass particles which differ in at least one property (chemical

composition, temperature, viscosity, aggregate state, density, etc.). The goal of mixing is the equalization of these different components in a compound [5]. Very different types of mixing occur in wineries such as liquid-solid suspensions (e.g. punchdowns, bâtonage), liquid-liquid mixing with high viscosity gradients (e.g. dosage in sparkling wine), gas-gas homogenizations (e.g. airing fermentation rooms), gas-liquid nebulization (e.g. humidity control), and gas-solid fumigation (e.g. burning sulfur in oak barrels).

Most mixing processes in wineries, however, are homogenisation processes involving two miscible liquids (juice, must, wine) with similar physical properties. During fermentation the desorption of CO₂ induces enough turbulence to homogenize all but the largest fermentation vessels. Before and after fermentation, however, naturally occurring diffusion (presence of concentration gradients) and convective heat transfer (presence of temperature gradients) may not induce sufficient flow for homogenization. Here mixing can be achieved by inducing a flow to the liquid either by hand, by pumping, or by mechanical mixing. Benefits of efficient mechanical mixing include enhanced additive mixing, enhanced rates of cooling during fermentation or cold stabilization through liquid circulation, destratification, and better blend quality.

1.2. Mechanical mixer types in wineries

Smaller wineries (up to 50.000 L tanks) usually rely entirely on mobile mixing units. These hollow-shaft agitators can be mounted on closed tank valves and, after opening the shut-off device, the retractable propeller blades can be pushed far enough into the liquid for mixing. These mobile agitators are very versatile and interchangeable among tanks.

Larger tanks (>100.000 L) have permanently installed mixers running continuously during wine making processes during which convection or CO₂ bubble formation usually suffice for mixing in smaller vessels (e.g. fermentation, cold stabilization, wine storage). Whereas mobile agitators are usually installed on the side of the tank, permanently installed mixers may also be installed centrally.

Different propeller geometries can be found in winery mixers with the most common ones being 3-blade impellers, pitched blade propellers, disk-type agitators (Rushton turbine), and impeller propellers. As mentioned above, mobile agitators rely on retractable propellers not wider than the diameter of the shaft. Mobile and fixed agitators may or may not be equipped with variable frequency drives allowing for the adjustment of the propeller speed (revolutions per minute).

1.3. Mechanical mixer types in wineries

Common methods to experimentally measure mixing time (pH meter or decolorisation method) and flow fields (laser Doppler velocimetry or particle image velocimetry), are rather impractical to conduct in a winery setting where many different combinations of mixers and tanks are feasible. However, many turbulence models and CFD solvers have been successfully validated using these experimental methods [6] and are customizable to typical mixing problems in wineries.

The performance of the mixer (thrust) is dependent on the flow losses in the tank which are directly related to its geometry. By modeling the actual mixers and tanks at a winery, practical application of CFD simulations can provide an insight into important parameters like average velocity, velocity distribution, mixing time, and flow fields, without conducting experiments.

2. Materials and methods

An analysis of a tank-mixer scenario can be divided into two equally important parts: the case study and the CFD case. Figure 1 gives an overview of a typical workflow with the case dependent steps in yellow and the software based CFD analysis steps in orange. All software discussed (Salome, OpenFoam, ParaView, DAKOTA) is open source and freely available. The last step is the analysis of the results- in the discussed case with regards to the mixing quality.

2.1. Case study

The first step is comprised of the system analysis of the situation at hand and compiling the critical factors for the ensuing CFD case.

2.1.1. Tank and propeller geometry

The geometry of the tank and mixer have a direct influence on the CFD solution since the flow is dependent on the mixer thrust and the flow losses due to wall friction, corner losses, and losses from tank obstacles.

Wine tanks differ widely in volume, height to diameter ratio, fixtures, and shape. Technical drawings from the manufacturer are the best source of geometrical data but if these cannot be obtained, manual measurements can suffice. Distances and angles are measured as accurately as

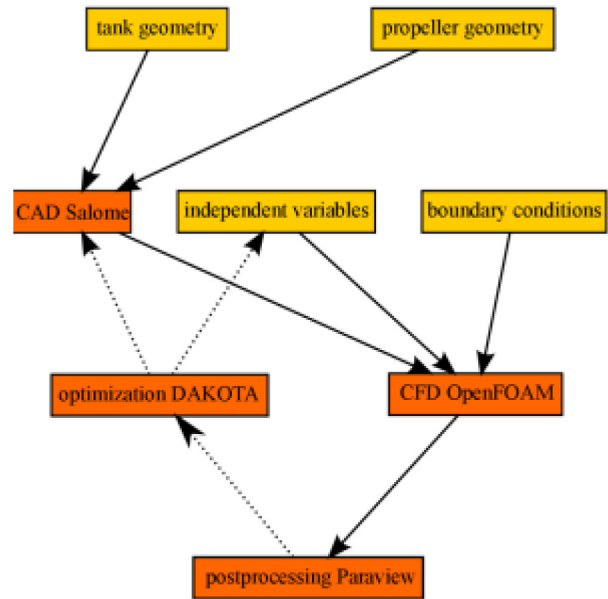


Figure 1. Workflow of a winery mixing process CFD analysis.

possible because even a slight divergence of, for example, the diameter of the tank, will have a great impact on the solution. Fixtures and baffles are sized even if they may be skipped later in the CAD model for modeling purposes. If volumetric information is known about the tank (e.g. total volume of tank, volume per centimeter, volume of tank floor) it can be used to verify the technical data or the measurements from the tank.

Propeller geometries are usually smaller in size but more complex than tank geometries. Most critical design characteristics are the diameter of the propeller, shape of the blades, number of blades, angle of blades with respect to the shaft, and, if present, the twist of the blades. When available, technical drawings give all the relevant information to reconstruct the geometry of the propeller. Of equal importance is the exact position of the propeller in the tank and its angle with respect to the tank wall.

2.1.2. Independent variables and initial values

Variables specific to the situation at hand need to be defined. The properties of the (incompressible) liquid to be mixed are described using kinematic viscosity ν (in m²/s) and density ρ (in kg/m³). The rotational velocity of the propeller is described using radians per second ω . If the tank is not filled all the way, the fill height is measured, as well.

Contrary to many other flow problems in CFD, tank-mixer problems do not necessitate an inlet and outlet and the entire, confined, liquid domain is simulated. Initial conditions therefore only involve walls which can be assigned a no-slip boundary condition with the velocity U (in m/s) at the wall being 0. Unless a flow field has already been calculated before, the liquid in the tank is considered static at the beginning of the CFD case ($U_{\text{liquid}} = 0$).

2.2. CFD case

A CFD mixing analysis begins with the generation of the propeller and tank geometry using computer aided

engineering (CAD) software. Resulting geometry files are meshed using the utility *snappyHexMesh* which is integrated in OpenFOAM. The rotation of the propeller is defined and modelled using the sliding mesh (SM) or multiple reference frame (MRF) method and boundary conditions are assigned for the CFD case. After choosing the solver and discretization schemes, the simulation is run with OpenFOAM. Postprocessing software ParaView is used to visualize the results of the computation for the user and to calculate key figures (e.g. average velocity, mixing quality). Finally, the optimization software DAKOTA may be used to find more favorable initial values using the results of the CFD analysis.

2.2.1. CAD Salome

The 3D CAD software Salome (Version 7.4, GNU LGPL license) is used to generate all relevant geometries in the fluid domain [7]. The graphical user interface (GUI) allows for the intuitive creation of primitive objects (e.g. cylinder, cone, box) which can subsequently be transformed (e.g. through translation, rotation, scaling), altered (e.g. extruded, chamfered, partitioned), or interconnected through Boolean operations (e.g. fuse, cut, common). The resulting geometries can be exported into a number of file formats (e.g. STL, BREP or STEP files) and used for meshing- either with Salome itself, *snappyHexMesh* (see Sect. 2.2.2), or any other commercial software. While building the geometry, Salome automatically records all steps in a construction history using Python programming language. The script can afterwards be viewed and modified with any text editor. This function becomes very important when changes to the geometry (e.g. angle of propeller or propeller diameter) are used to optimize the ensuing CFD analyses.

2.2.2. CFD OpenFOAM

OpenFOAM (Field Operation And Manipulation) is a library of C++ code used to create executable applications which can be classified as solvers and utilities. All analyses were performed using OpenFOAM Version 2.3.1 (GPL license). Utilities are used to set up cases (e.g. pre-processing, mesh generation, mesh conversion, parallel processing) and post-processing (e.g. visualization, sampling, calculation of field numbers), whereas solvers are executable programs to solve specific problems in continuum mechanics or, in this case, CFD [8].

OpenFOAM evaluates the underlying partial differential equation using the finite-volume-method, it is easily networked with other open source software (e.g. Salome, ParaView, and DAKOTA), and utilities and solvers can be personalized fairly easily since all parts of the underlying C++ library are accessible discretely and are connectable.

For the meshing of tank and propeller geometries generated with Salome, the OpenFOAM utilities *blockMesh*, *surfaceFeatureExtract*, and *snappyHexMesh* are used. These utilities allow the user to define the domain and break it up into a coarse mesh (*blockMesh*), then define feature edges in the geometries that should have sharp edges in the final mesh (*surfaceFeatureExtract*), and, finally, create the refined mesh using *snappyHexMesh* which snaps the coarse mesh to the surface of the geometries and adds cell layers near surfaces to create boundary layers. The

snappyHexMesh application also permits the user to define certain mesh quality criteria (e.g. maximum skewness of cells, maximum non-orthogonality of cells) to control the final mesh. It is important to note, that regions of interest (e.g. a rotating zone around the propeller, areas of interest for sampling after the simulation) are already integrated during the mesh generation process.

Two approaches are used for tank-mixer CFD analyses: the moving reference frame (MRF) and the sliding mesh (SM) method. For the MRF method an area around the propeller is modelled as a rotating coordinate system while the fluid outside this zone up to the tank wall is solved in a stationary coordinate system. Most OpenFOAM steady-state solvers are capable of handling MRF using the *fvOptions* framework (e.g. *simpleFoam*). When using the SM method, the position of the propeller is calculated at each timestep and therefore transient problems with strong rotor-stator interactions can also be solved. OpenFOAM solvers capable of this type of dynamic mesh functionality are usually identified by “*DyM*” in their name (e.g. *pimpleDyMFoam*).

Which approach to use is very much case dependent. Some considerations are listed in Table 1.

2.2.3. Postprocessing ParaView

Paraview (Version 4.3, BSD license) is an open source data analysis and visualization application which was initially developed by Kitware Inc. in cooperation with Los Alamos National Laboratory [9].

The Paraview-based post-processing application *paraFoam* recognizes the openFOAM file structure (Fig. 3) and is capable of depicting the geometry, the mesh, as well as data fields, written to the time folders. Once the mixer-tank OpenFOAM simulation has converged (steady-state case) or has been solved up to the first written out time-step (transient case), this software allows the user to look at the flow field data using a graphical user interface. Various parameters (e.g. velocity, pressure, phase percentages) can be visualized by applying filters (e.g. contour, clip, glyph, stream tracer) in the 3D domain. Many other filters can help the user with further analyzing the data (e.g. integrate variables, plot data, calculator).

2.2.4. Optimization DAKOTA

DAKOTA (Design Analysis Kit for Optimization and Terascale Applications) is a C++ programmed open

Table 1. Advantages and disadvantages of MRF and SM approach in propeller-tank simulations.

Sliding mesh method	Multiple reference frame method
Long computing time due to short timesteps	Shorter runtime until convergence
Strong interaction with stationary zone	Weak interaction with stationary zone
Transient solution possible	Steady-state solution
Intersection calculations at each time step	Robust method

source toolkit by Sandia Corporation which provides an interface between the OpenFOAM model and iterative optimization methods (Version 6.1., GNU LGPL license) [10]. The OpenFOAM CFD case is linked to the DAKOTA framework and defined input parameters (e.g. geometry of propeller blade, viscosity of fluid) can be changed automatically according to integrated optimization methods using defined outputs from a previous CFD case (e.g. average velocity, mixing quality). Another potential use for this software with respect to mixer-tank scenarios can be the estimation of parameters (e.g. thrust or torque coefficient of propellers) using experimental data.

2.3. Implementation

Analyses for this study were performed using a regular work station equipped with an Intel® Xeon® E5-1620v2 processor and 32GB RAM. All simulations were run on the operating system gm.linux (v.3.01) [11].

3. Results and discussion

In the results section we will examine the CFD analysis step-by-step.

3.1. Case study

Table 2 shows the case specific variables used for the tank-mixer system analysis.

3.2. CFD case

Hereafter several illustrations are presented showing the progress of an exemplary tank-mixer CFD analysis.

3.2.1. CAD Salome

The 3D CAD software Salome (Version 7.4, GNU LGPL license) was used to create the geometry for the CFD simulation as described in Sect. 2.2.1.

Figure 2 depicts the .stl propeller geometry created using Salome. The relevant measurements of the propeller

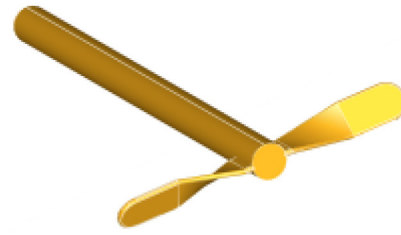


Figure 2. Propeller geometry created with Salome.

were taken on the movable mixer (Schneider Rührwerk, Model TS-VA-SWK). Not shown here is the geometry of the tank (Becker Tanks, Gau Odernheim, CE0035), which was created analogously. The propeller is placed inside the tank in the appropriate position within the Salome CAD project before all geometries are exported into .stl files. Subsequently these files are imported into the CFD case through the *snappyHexMesh* dictionary file.

Figure 3 shows a section of the mesh (orange) created by the *snappyHexMesh* OpenFOAM application. The yellow propeller shows the exact position of the propeller (Fig. 2) around which the mesh was created. The snapping can be seen around the shaft of the propeller outside of the MRF zone (opaque yellow) and the refinement of the cell zone around the propeller is apparent compared to the larger cell sizes further away from the rotor. The final mesh consisted of 377,262 mainly hexahedral cells (average faces per cell 6.14).

3.2.2. CFD OpenFOAM

Parameters for the tank-mixer study are defined in the respective OpenFOAM files.

All OpenFOAM analyses are structured in a defined way. The folders and files depicted in Fig. 4 represent the minimum configuration to run a CFD analysis using Reynolds Averaged Navier Stokes turbulence model (RANS, *k-Epsilon*) with an MRF zone. The *interFoam* solver chosen for the analysis is capable of simulating two incompressible, immiscible fluids. This will allow us to

Table 2. Variables and methods used for the tank- mixer case study.

Variable and unit	Value or method
Kinematic viscosity (m ² /s)	1.31×10^{-6}
Density (water at 10°C in kg/m ³)	1001.28
Rotation (rpm)	1300
Fill level of tank (L)	6300
Diameter of propeller (m)	0.171
Position of propeller (height from floor in m, angle θ in degrees)	1.26, 15.1

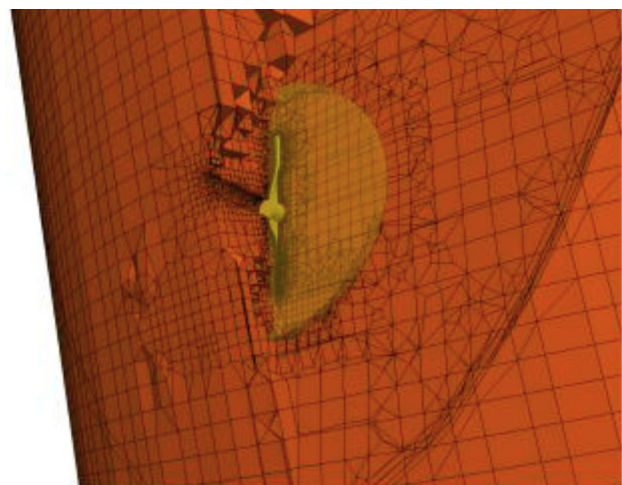


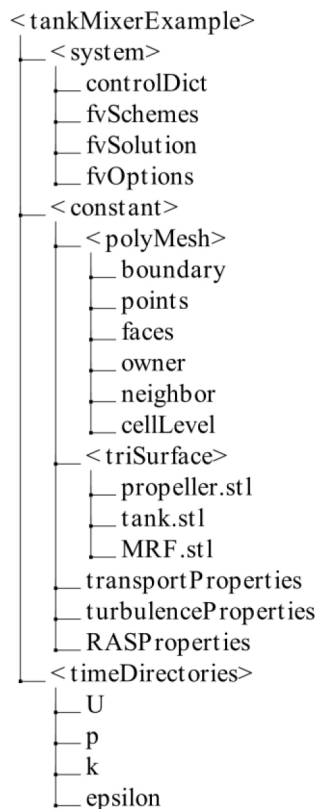
Figure 3. Mesh (orange) created around the propeller and inside the tank geometry (yellow) and MRF zone (opaque yellow).

Table 3. Variables and methods used for the tank- mixer CFD case.

Variable and unit	Value or method
Modelling method for rotation	MRF, 134.14 rad/s
Turbulence model	RANS, $k-\epsilon$
Time discretization scheme	Euler
Gradient discretization scheme	Gauss linear
Solver	<i>interFoam</i>

simulate different fill heights with air as the second phase in the headspace and thus graphically compare the simulated surface waves to experiments in the tank.

The files in the <system> folder contain all the information for the solution procedure itself (see Table 3). The <constant> folder contains the geometrical (<triSurface> folder) and physical information (*Properties files), as well as the information for the mesh (<polyMesh> folder). The results of the analysis is written out in so-called time folders which are named for the actual time simulated (transient) or the iterations computed (steadyState). The initial time directory <0> contains the boundary conditions and initial values for the individual scalar and vector fields. All subsequently created time directories written

**Figure 4.** File structure of the tank-mixer OpenFOAM case.

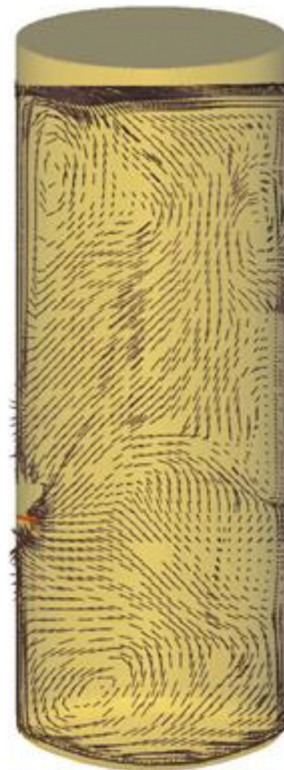
out by OpenFOAM contain information computed for each individual cell.

In this simulation the computational time for 1 s real-time simulation was approximately 5 h, even though the computationally cheaper MRF approach was used (see Sect. 2.2.2). In addition to the method of simulating the rotation, cost in computational time with tank-mixer simulations depend strongly on the cell number and maximum velocities. Velocities near the tip of the propeller blades were highest with approximately 11.6 m/s. Because of these high velocities and the small cell sizes around the propeller region, the time steps have to be very small (2.5×10^{-5} s).

3.2.3. Postprocessing ParaView

Different filters are applied to the velocity flow field after 60 s of realtime simulation. The visualization in Fig. 5 uses the *Slice* filter to show a center pane normal to the x -axis of the flow field (parallel to and centered in the propeller). Afterwards a *Glyph* filter was used on the velocity flow field U in order to depict the direction of the flow in this region. Several larger vortex areas are apparent near the bottom, right side, and top of the tank and a large region in the middle where the flow is pulled toward the propeller.

Figure 6 illustrates the three dimensional flow after 60 s of simulation using the *streamTracer* filter. This filter follows the flow at a given time starting from a number of random seed points in a defined location. Velocity was scaled to 0.5 m/s because the very high velocities in close proximity to the propeller are not of interest with respect to

**Figure 5.** Direction of flow after 60 seconds in center pane.

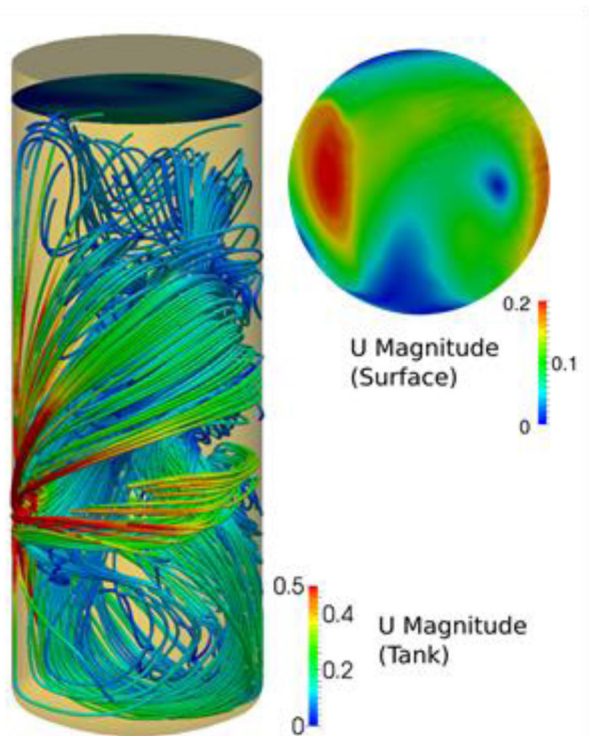


Figure 6. Streamtracer for flow field after 60 s.

the mixing behaviour in the whole tank. This visualization reveals high velocity zones along the tank walls moving perpendicular to the propeller blades. Taking a closer look at the liquid surface using a *Contour* filter and coloring the surface by velocity (0–0.2 m/s) reveals some high and low velocity areas on the surface (Fig. 6). This implies that the mixing of the whole tank all the way up to the liquid surface has started at this time point.

4. Conclusion

It could be shown that openSource software can supersede commercial software for fluid flow analyses regarding winery tank-mixer interactions. The methods can be applied without too much effort on regular personal computers. However, care is required in the geometrical implementation of the model, in grid generation, in selection of proper solution strategies and in the interpretation of the results [12].

References

- [1] A. De Rosis, A. Barbaresi, D. Torreggiani, B. Stefano, P. Tassinari. *Numerical simulations of the airflows in a wine-aging room: A lattice Boltzmann-Immersed Boundary study*. *Comput. Electron. Agric.* **109**, 261–270 (2014)
- [2] M. Bhutta, N. Hayat, M. Bashir, A. Khan, K. Ahmad, S. Khan (2012). *CFD applications in various heat exchangers design: A review*. *Appl. Therm. Eng.* **32**, 1–12 (2012)
- [3] T. Norton, D. Sun. *Computational fluid dynamics (CFD)- an effective and efficient design and analysis tool for the food industry: a review*. *Trends Food Sci. Tech.* **17**, 600–620 (2006)
- [4] X. Jing, Q. Xiaohua, X. Shiqiong. *Numerical simulation and verification of airflow in cold-store*. *Transactions of Chinese Soc. Agric. Eng.* **2**, 4 (2005)
- [5] M. Kraume. *Mischen und Rühren*. Wiley-VCH, Berlin (2005)
- [6] J. Joshi, and N. Nere, and C Rane, and BN Murthy, C Mathpati, A. Patwardhan, V. Ranade. *CFD simulation of stirred tanks: Comparison of turbulence models (Part II: Axial flow impellers, multiple impellers and multiphase dispersions)*. *Can. J. Chem. Eng.* **89**, 754–816 (2011)
- [7] V. Bergeaud, V. Lefebvre. *SALOME. A software integration platform for multi-physics, pre-processing and visualisation*. *Proceedings : Joint international conference on supercomputing in nuclear applications and Monte Carlo* (2010)
- [8] OpenFOAM Foundation. *OpenFOAM User Guide Version 2.3.1*. OpenFOAM Foundation (2014), <http://www.openfoam.org/>
- [9] U. Ayachit. *The ParaView Guide: A Parallel Visualization Application*. Kitware Inc., New York (2015)
- [10] B.M. Adams, M.S. Ebeida, M.S. Eldred, J.D. Jakeman, L.P. Swiler, A. Stephens, D.M. Vigil, T.M. Wildey. *Sandia Technical Report 2014–4633*. Sandia National Laboratories, New Mexico (2014)
- [11] M. Günther, K. Velten. *Mathematische Modellbildung und Simulation*. Wiley-VCH, Berlin (2014)
- [12] B. Blocken (2015). *Computational Fluid Dynamics for urban physics: Importance, scales, possibilities, limitations and ten tips and tricks towards accurate and reliable simulations*. *Build. Environ.* [in press] doi: 10.1016/j.buildenv.2015.02.015