NTNU – Trondheim
Norwegian University of
Science and Technology

Model-based characterization of mili-reactor

Petr Vanek

Chemical Engineering
Submission date: August 2013
Supervisor: Heinz A. Preisig, IKP
Co-supervisor: Matthias Kleiner, BASF
Andreas Daiss, BASF

Norwegian University of Science and Technology
Department of Chemical Engineering

Acknowledgment

First of all I would like to thank professor *Heinz A. Preisig* for his guidance, support and assistance during the whole two years and especially for his help which allowed me to collaborate with BASF in Germany.

I would like to show my gratitude to Dr. *Matthias Kleiner*, Dr. *Andreas Daiss* and Dr. *Jenine Chungyin Cheng* from Polymer Research Department in **BASF**, Ludwigshafen for the opportunity to spend 3 months in one of the biggest chemical companies in the world and gain very useful knowledge and experience.

I am also thankful to staff from Department and International offices on both **NTNU Trondheim** and **ICT Prague** for helping me with all organizational things during whole my studies.

Last but not least I am grateful to my family and friends for their support when it was needed the most.

Summary

This thesis focuses on development of a functioning, reliable and representative computation fluid dynamics model of milli-reactors in OpenFOAM in order to characterize them. Characterization and validation of the model was done by residence time distribution analyses, where model data were compared with experimental measurements.

The model was developed for Fluitec reactor with CSE-X4 static mixer elements with diameter of 21 mm. The residence time distribution was calculated for various flow conditions in the laminar flow regime. The model followed the experimental procedure where residence time distribution is measured by injection of tracer and evaluation of its concentration at the outlet from the reactor.

The comparison of calculated and experimental integrated residence time distribution curves showed good agreement. Although, it would be possible to improve the model if more information of the experimental system were provided. It was also possible spot some dead zones inside the reactor, but the experimental data suggest that it was not successfully found all of them.

I declare that this is an independent work according to the exam regulations of the Norwegian University of Science and Technology (NTNU).

Petr Vaněk

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Mixing is the very basic unit operation in chemical engineering and from process point of view it is very simple, but if we look at the mixing process itself it can be very complicated. Mixers are either active where usually a rotating propeller is mixing the fluid or static where all parts of the mixer are fixed and the mixing is performed by flow of the fluid around obstacles in the flow stream. Static mixers are nowadays gaining importance in various fields and applications, because of their high performance and low maintenance demands.

In order to avoid problems with scaling up of the processes from laboratory trials to production size, it is currently put a great effort to switch from batch to continuous processes. For production of specialized polymers and pharmaceutics, it seems the best to use static mixer tube reactors with small dimensions since it is not required to produce big amount rather then it is required to have tight control of the product quality, hence milli-reactors.

Nowadays suitability of milli-reactors is investigated for production of polymers. In present there are pilot trials in operation. The current aim is to be able to completely avoid the pilot step in a process development. It would be very beneficial to be able to scale up directly from laboratory trials to production scale without the pilot project in between. With batch reactors this is not possible since the volume/area ratio is changing significantly during the scale up. Hence there is a need to make an intermediate step to see the scale up effect. On the other hand milli-reactors seem to be appropriate for this approach since it is possible to keep the volume/surface ratio nearly constant or with very small changes.

In these days milli-reactors are used only in pilot scale and development of the product is performed in batch reactors. It would be very beneficial if it was possible to use the milli-reactors also in the product development. This would mean to scale the currently operating reactors down into laboratory size, with keeping their performance. This issue is currently under research and a big part of the research is done by computational fluid dynamics.

Combination of polymers and very small dimension results in serious problems with clogging and gel formation. It is expected that the gel formation occurs mainly in dead

volume. Hence it is very important to locate these volumes and find a solution how to minimize them either by adjustment of the operating conditions or by adjustment of the reactor.

Even though CFD modeling is on very high level, it is still only an approximation of real systems. So, there is need to compare the calculations with experimental data in order to verify the model. After the model is verified and the error of approximation is established, it is possible to use the model for predicting of the behavior of the system.

The main task of this thesis is to develop a reliable model of milli-reactor, which consists of static mixer and heat exchanger. The flow in the reactors is modeled in terms of residence time distribution and the results are compared with measured data.

There is a lot of research done in the field of static milli- and micro- mixers. The research focuses on both the reactor development and the reactor characterization. Some of the research is summarize in Chapter 2

An introduction to CFD is given in Chapter 3. There are presented and summarized theoretical basics — equations and principles — of fluid dynamics and numerics.

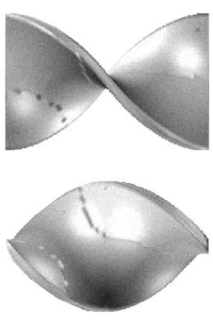
The model is developed in OpenFOAM, an open source CFD toolbox. Another softwares were used in pre- and post- processing of the model and the results. The used software is briefly described in Chapter 4. The development and settings of the model are also presented and described in this chapter.

Chapters 5 and 6 summarize the obtained results and conclude their meaning for the future work.

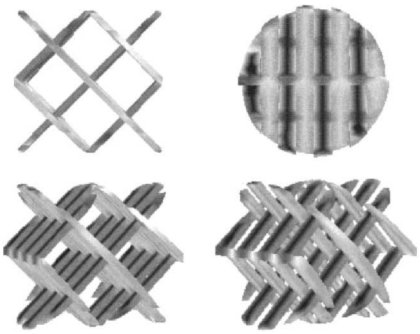
Chapter 2

Literature review

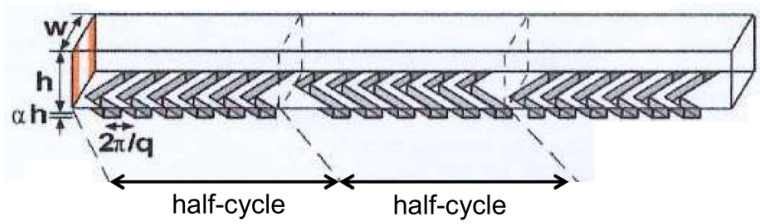
The current research is done in two main directions. One focuses on design and optimization of the design of the mixers and the other is focused on characterization of the mixing quality in current static mixers. Both of the fields use CFD modeling as a tool to obtain the necessary information.



(a) Different views of one Kenics mixing element. *(Taken from [10].)*



(b) Different views of one SMX mixing element. *(Taken from [10].)*



(c) Geometry of staggered herringbone micromixer (SHM). *(Modified from [22].)*

Figure 2.1: Most researched geometries of static mixers.

The main researched geometries of the static mixers are captured in Figure 2.1. The Figures 2.1a and 2.1b are showing only one mixing element. In the reactor are these mixing elements put after each other and the subsequent element is always rotated by $90°$.

An extensive study was done in [4] to investigate the influence of the change in geometrical parameters for staggered herringbone micromixer (see Figure 2.1c). The performance of the mixer was quantified by spatial data statistics, maximum striation thickness and residence time analyses. The modified parameters were groove depth, width and number of grooves per cycle. Particle tracking was used in order to visualize the mixing. One quadrant of the reactor cross-section was filled with equidistant particles and their distribution was observed at the cross-sections along the reactor for different parameter setups. The homogeneity of spatial mixing was quantified by a variance of average square of a difference between distance of the particles and distance between the equidistant particles:

$$\text{var} = \frac{1}{N-1}\sum_{i=1}^{N}(d_i - \bar{d}_R)^2, \tag{2.1}$$

where d_i is point-particle distance, $\bar{d}_R$ expected mean distance and N is number of particles. *"The 'well mixed' criteria was set such that variance value of 0 corresponds to uniform spatial distribution of tracer particles."* [4]

It was concluded that number of grooves per cycle does not have a significant importance on mixing quality. A large groove depth and thickness reduce the striation thickness without increasing the pressure drop. Deep grooves improve the special mixing but the wide grooves appear to cause a dead volume occurrence.

Group of scientists from Eindhoven University of Technology researched and optimized the Sulzer SMX static mixer. [20] The analyses of the mixer was done by use of Mapping Method, which is based on backward particle tracking [21]. It was investigated the effect of three main geometrical parameters on the interface stretching and the mixing efficiency. The changed parameters for SMX mixer are: the number of cross-bars, N_x, the number of parallel cross-bars, N_p and the angle between opposite cross-bars, θ.

The CFD simulation was performed in Fluent 5, which solves the Navier-Stokes equations. In order to avoid the developing flow empty tube with length of two times the diameter is added at the inlet and the outlet. For standard SMX design the mesh consists around 2 millions tetrahedral cells. In the calculations the no-slip boundary condition is assumed for walls and mixing element surfaces. It is assumed a Newtonian fluid with condition yielding in Reynolds number of 0.44.

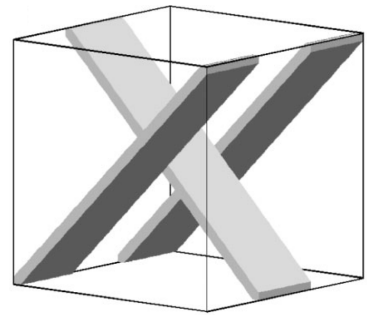


Figure 2.2: The basic most simple SMX mixing element. *(Taken from [20])*

It was concluded that the optimal design of the SMX mixing element with square cross-section obeys rule $(n, N_p, N_x) = (n, 2n-1, 3n)$ where $n = 1, 2, 3, \ldots$ is a design parameter. It was proved that the basic design with $n = 1$ (see Figure 2.2), e. i. design with one parallel bar and three crossing bars (with angle $\theta = 90°$) gives the best mixing for the lowest energy consumption and

higher values of n yield in more compact mixers. Increasing the angle θ slightly increases the interfacial stretching and decreases the pressure drop per element.

Eric Fourcade and his group developed a new method of calculation of average value of the rate of striation thinning, α. [10] A laminar flow in Kenics (Fig. 2.1a) and SMX (Fig. 2.1b) static mixer elements was investigated by CFD. The model was evaluated for one element and for three subsequent elements. A particle tracking procedure was used on the calculated flow field and compared to a laser induced fluorescence experiment.

This method was developed since the rate of striation thinning is predicted to have an influence on molecular weight distribution in polymer production. [9] It uses CFD calculated velocity field by POLY3D$^{\text{TM}}$ code. Particles are tracked in order to compare striation thickness, s, at the inlet cross-section of a mixing element and at the outlet cross-section of the element. The relation between striation thickness, s and rate of striation thinning, α, is:

$$\alpha(t) = -\frac{\mathrm{d}\ln s}{\mathrm{d}t}\,. \tag{2.2}$$

The paper [10] shows issues that were found during the research, especially with finding the striation thickness of particle streams which get split during their way through the mixing element. Despite these issues it was possible to calculate the values of α. The CFD and striation model was verified by pressure drop comparison (for the comparison see [18]) and also by laser induced fluorescence.

Laser induced fluorescence is a non-invasive technique where fluorescent dye is excited by laser and the emitted light is measured by CCD camera and PC based image capture. The emitted light is proportional to the dye concentration. The numerical and experimental results showed a good resemblance. Hence, it could be concluded that the way of calculation of the striation thinning parameter α is credible. It was also proven that the average value of α is constant per element, which is in agreement with [17].

Staggered herringbone mixer (Figure 2.1c) and diagonal mixer (i. e. simplified SHM with no direction change in the grooves) were investigated by J. Aubin *et al.* [3]. The particle tracing was performed and consequently the variance of tracer dispersion was evaluated together with rate of deformation and stretching. The simulation was done by CFX5, where first a steady state solution of the velocity and pressure field was obtained and then the vector equation of motion was integrated. Water at 25°C and 1 atm was used as operated fluid with $Re \approx 2$. Standard boundary conditions were used — no-slip boundary at all walls, constant pressure at the outlet ($P = 0$).

The 2 480 equally distributed particles were injected in upper right corner of the cross-section and their distribution was observed on number of cross-sections through the mixer. The results were compared with experimental data obtained by Stroock *et al.* [22] and it showed good agreement. The conclusion was the same for both [3] and [22], that 3 cm of the staggered herringbone mixer are sufficient in order to get fully mixed flow with a Peclet number $< 10^6$.

Chapter 3

Theory

This chapter provides a brief overview of basics of computational fluid dynamics. The basic equations will be presented and described as well as the basics of numerical methods.

3.1 Introduction to CFD

"Computational fluid dynamics (CFD) is the art of replacing the integrals or partial derivatives (as the case may be) in governing equations with discretized algebraic forms, which in turn are solved to obtain numbers for the flow field values at discrete points in time and/or space." [2]

Computational fluid dynamics predicts behavior of a fluid in terms of mass transfer, temperature transfer, chemical reaction and other related phenomena. The prediction is based on mathematical description of the problem and its numerical solution. The equations describing the behavior appear in partial differential, integral or algebraic form. These equations form a set which is then solved numerically.

CFD is used in various fields from aerospace engineering to sport equipment development. Its wide use is tightly connected with the growth of computational power and capacity. In present CFD is used as research tool as well as design tool. The modeling is used in cases where experiments are not capable of giving all the necessary information or it is impossible to carry out the experiment.

CFD was originally solved by user-designed programs. These were designed to meet the particular needs of the user. Today specialized programs are available. They are developed either by commercial companies, which sell them to the users, or by open source communities. In the specialized programs often many models are already implemented which require just simple adjustments for particular application.

Since the user base is significantly growing also the CFD programs are getting more available and easier to use. Nowadays, there are available several programs dealing with CFD solution, both in commercial and open source domain. In commercial domain, probably the best know are FLUENT, CFX and COMSOL; in open source community the most widely used codes are OpenFOAM, SU^2 and Code_Saturn.

Since to solution is done numerically by a computer, it is necessary to discretize the domain in time and space. The time discretization is done by the solver code, of the space discretization is usually taken care by another software coupled with the solver during so called pre-processing. Pre-processing includes definition of the geometry and the mesh (grid) creation. After the mesh is created the problem is ready to be solved. Results of the simulation can be visualized and processed in the following step, so called post-processing. Some of the codes have pre- and post-processing tools incorporated in them, but for complex problems it is usually beneficial to use specialized software. Also this software can be found as open source as well as commercial package.

3.2 Mathematical basis of CFD

Three basic physical principles applied in CFD are:

- conservation of mass;
- conservation of energy;
- Newton's second law.

These principles are mathematically represented by governing equations (continuity equation, energy equation and momentum equation, respectively). The equations can be either differential or integral depending on the application and the used discretization scheme. Either one of the types can have conservation or non-conservation form depending on a frame of reference. From mathematical point of view all four equations are equivalent, even though the appearance differ significantly and it is possible to transform one into another with simple adjustments. Derivation of these equations is very nicely shown in [2, Chapter 2] and this section is going to give a brief overview of it.

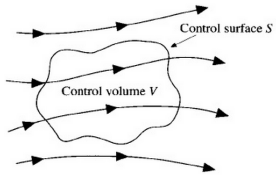
It is assumed that reader is well educated in substantial derivative, divergence and basics of calculus. (For revision see e. g. [2, pp. 43–49].)

Four basic models of flow are used while deriving the governing equations. For derivation one uses either a control volume fixed in space, where the fluid is moving through it (Figures 3.1a and 3.1c), or the control volume is moving along with the fluid, so that the same fluid particles are always in the same control volume (Figures 3.1b and 3.1d). Also, one uses either a finite volume (Figures 3.1a and 3.1b) or infinitesimally small volume (Figures 3.1c and 3.1d). According to which model is picked to derive the governing equations different forms are obtained. The forms and the models used to obtain them are summarized in Table 3.1.

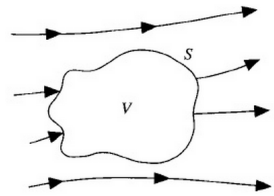
Even though most of the softwares are case oriented and basic user does not have to take care about actual form of the governing equations it is necessary to have an insight to the mathematics.

3.2.1 Continuity equation

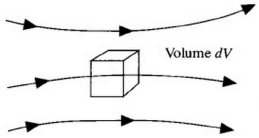
Continuity equation is the result of applying the **mass conservation principle**. For purposes of this report an infinitesimally small control volume moving with the flow was



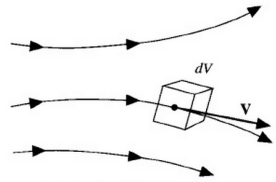
(a) Finite control volume fixed in space.



(b) Finite control volume moving with the fluid.



(c) Infinitesimal control volume fixed in space.



(d) Infinitesimal control volume moving with the fluid.

Figure 3.1: Model of a flow. *(Adapted from [2].)*

Table 3.1: Summary of flow model — equation form relations.

Used model	**Obtained form of governing equations**
Fixed finite volume	integral conservation form
Moving finite volume	integral non-conservation form
Fixed infinitesimal volume	differential conservation form
Moving infinitesimal volume	differential non-conservation form

chosen as the flow model for which the continuity equation will be derived. *Differential non-conservation form* will be obtained and then it will be shown how to transform it into different forms.

The model is pick such that mass of the control volume m is constant, the volume V and the density ρ are variable. Which are dependant as the basic physics define:

$$m = \rho V. \tag{3.1}$$

The fact that the mass is constant can be expressed so that the time rate of change of the mass of the fluid element is zero as the element moves along the flow, which reflects precisely the physical meaning of the substantial derivative. So in terms of equations it can be written:

$$\frac{\mathrm{D}m}{\mathrm{D}t} = 0. \tag{3.2}$$

By combining equations (3.1) and (3.2), we get:

$$\frac{\mathrm{D}(\rho\mathcal{V})}{\mathrm{D}t} = \mathcal{V}\frac{\mathrm{D}\rho}{\mathrm{D}t} + \rho\frac{\mathrm{D}\mathcal{V}}{\mathrm{D}t} = 0,$$

or

$$\frac{\mathrm{D}\rho}{\mathrm{D}t} + \rho\left[\frac{1}{\mathcal{V}}\frac{\mathrm{D}\mathcal{V}}{\mathrm{D}t}\right] = 0. \tag{3.3}$$

The term in the brackets is exactly the definition of the divergence of the velocity $\nabla \cdot \mathbf{V}$. Hence substituting this expression into equation (3.3) we obtain:

$$\boxed{\frac{\mathrm{D}\rho}{\mathrm{D}t} + \rho\nabla \cdot \mathbf{V} = 0}. \tag{3.4}$$

This is partial differential equation in non-conservation form describing the mass conservation, so called continuity equation. Another form of the continuity equation can be derived in two ways. First the direct derivation, as it was done above but using a different flow model. Second option is the indirect derivation, which means to take the form derived before, e. g. form in equation (3.4) and transform it. If one is interested in conservation form of differential equation, the manipulation is fairly simple. Mathematical definition of substantial derivative is applied on the first term in equation (3.4), so

$$\frac{\mathrm{D}\rho}{\mathrm{D}t} = \frac{\partial\rho}{\partial t} + (\mathbf{V} \cdot \nabla\rho)\,. \tag{3.5}$$

The chain rule of ∇ operator is

$$\nabla \cdot (\rho\mathbf{V}) \equiv (\mathbf{V} \cdot \nabla\rho) + (\rho\nabla \cdot \mathbf{V})\,. \tag{3.6}$$

By combining equations (3.4), (3.5) and (3.6) the partial differential conservation form of the continuity equation is obtained

$$\boxed{\frac{\partial\rho}{\partial t} + \nabla \cdot (\rho\mathbf{V}) = 0}. \tag{3.7}$$

In some cases it may be better (more convenient) to use the integral form. Again two ways are possible. Since we already derived two forms of the continuity equation, it is simpler to transform them into integral form, in contrast to performing a complete derivation. For mass to be conserved it is assumed that equation (3.7) has to be valid in every point of the system. For arbitrary volume V drawn in space it has to be valid:

$$\iiint\limits_{\mathcal{V}} \left[\frac{\partial\rho}{\partial t} + \nabla \cdot (\rho\mathbf{V})\right] \mathrm{d}\mathcal{V} = 0,$$

or

$$\iiint\limits_{\mathcal{V}} \frac{\partial\rho}{\partial t}\mathrm{d}\mathcal{V} + \iiint\limits_{\mathcal{V}} \nabla \cdot (\rho\mathbf{V})\,\mathrm{d}\mathcal{V} = 0. \tag{3.8}$$

The divergence theorem from vector calculus is applied to the second term of equation (3.8) and time derivative is taken out of the integral from the first term. The integral conservation form is obtained:

$$\boxed{\frac{\partial}{\partial t}\iiint_{\mathcal{V}} \rho \, \mathrm{d}\mathcal{V} + \iint_{\mathcal{S}} (\rho \mathbf{V}) \cdot \mathbf{dS} = 0}. \tag{3.9}$$

It would be possible to continue in transforming one form into another, but it is not necessary since the operations are the same.

3.2.2 Momentum equation

Physical principle involved in derivation of momentum equation is **Newton's second law**:

$$\mathbf{F} = m\mathbf{a}. \tag{3.10}$$

For derivation of momentum equation it was decided to use an infinitesimally small fluid element moving with the flow as shown on Figure 3.1d. This choice was made because then the derivation of momentum equation and energy equation is quite similar. Newton's second law is a vector relation, hence it can be split into scalar relations, one along each axes (cartesian coordination system is assumed). On the element several forces act. The forces are of two types — body forces and surface forces.

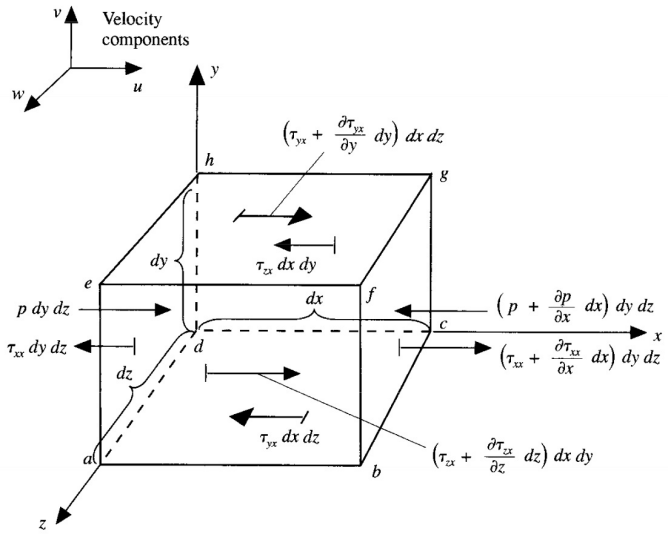


Figure 3.2: Acting forces on an infinitesimally small, moving element in x direction. *(Adapted from [2].)*

The body forces act directly on the mass of the element. These are e. g. gravitational, electric and magnetic. The surface forces act directly on the surface of the element.

Sources of the surface forces are pressure distribution and shear and normal stress. These forces are shown on Figure 3.2. When all the acting forces in x direction are summed up:

$$F_x = \left[-\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} \right] \mathrm{d}x\ \mathrm{d}y\ \mathrm{d}z + \rho f_x \mathrm{d}x\ \mathrm{d}y\ \mathrm{d}z. \tag{3.11}$$

Mass of the element can be expressed:

$$m = \rho\ \mathrm{d}x\ \mathrm{d}y\ \mathrm{d}z. \tag{3.12}$$

And the acceleration in x direction is rate of change of the x component of the velocity, so

$$a_x = \frac{\mathrm{D}u}{\mathrm{D}t}. \tag{3.13}$$

The same could be done for y and z directions accordingly. The set of three (one for each coordinate) partial differential equations in non-conservation form is then obtained.

$$\rho \frac{\mathrm{D}u}{\mathrm{D}t} = -\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + \rho f_x, \tag{3.14}$$

$$\rho \frac{\mathrm{D}v}{\mathrm{D}t} = -\frac{\partial p}{\partial y} + \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} + \rho f_y, \tag{3.15}$$

$$\rho \frac{\mathrm{D}w}{\mathrm{D}t} = -\frac{\partial p}{\partial z} + \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z} + \rho f_z. \tag{3.16}$$

These three equations (3.14 – 3.16) are called Navier-Stokes equations to honor M. Navier and G. Stokes — who both obtained the same solution independently on each other in the first half of 19th century. The conservation form and integral forms can be obtained in the same manner as was explained in section 3.2.1 where it is necessary to deal with the left hand side of Navier-Stokes equations only. The shear stress description is dependent on type of fluid (Newtonian/non-Newtonian). In practice most of the fluids are assumed to be Newtonian, which means that shear stress is proportional to the velocity gradients. For Newtonian fluids Stokes derived:

$$\tau_{xx} = \lambda(\nabla \cdot \mathbf{V}) + 2\mu \frac{\partial u}{\partial x}, \tag{3.17}$$

$$\tau_{yy} = \lambda(\nabla \cdot \mathbf{V}) + 2\mu \frac{\partial v}{\partial y}, \tag{3.18}$$

$$\tau_{zz} = \lambda(\nabla \cdot \mathbf{V}) + 2\mu \frac{\partial w}{\partial z}, \tag{3.19}$$

$$\tau_{xy} = \tau_{yx} = \mu \left[\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right], \tag{3.20}$$

$$\tau_{xz} = \tau_{zx} = \mu \left[\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right], \tag{3.21}$$

$$\tau_{yz} = \tau_{zy} = \mu \left[\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right]. \tag{3.22}$$

Where μ is molecular viscosity and λ is second viscosity, for which Stokes made hypothesis:

$$\lambda = -\frac{2}{3}\mu. \tag{3.23}$$

3.2.3 Energy equation

The energy equation reflects the **energy conservation principle**. Again the derivation is focused on infinitesimally small fluid element moving with the flow, so it is similar to the derivation of the momentum equation. Change of energy inside the element is caused by two sources — net flux of heat into the element and rate of work due to body and surface forces.

Energy of the element has two contributors, *internal energy* due to random motion of the molecules and *kinetic energy* due to motion of the element. The total energy is the sum of unit internal and kinetic energy multiplied with mass defined as equation (3.12). Then the rate of change of total energy of the element is substantial derivative of the total energy.

$$\frac{\mathrm{D}E}{\mathrm{D}t} = \rho \frac{\mathrm{D}}{\mathrm{D}t}\left(e + \frac{V^2}{2}\right) \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}z. \tag{3.24}$$

Energy fluxes associated with the moving element are captured on Figure 3.3. To keep the figure simple only energy fluxes in x direction are shown. The energy fluxes influencing the total energy of the element are also in y and z direction.

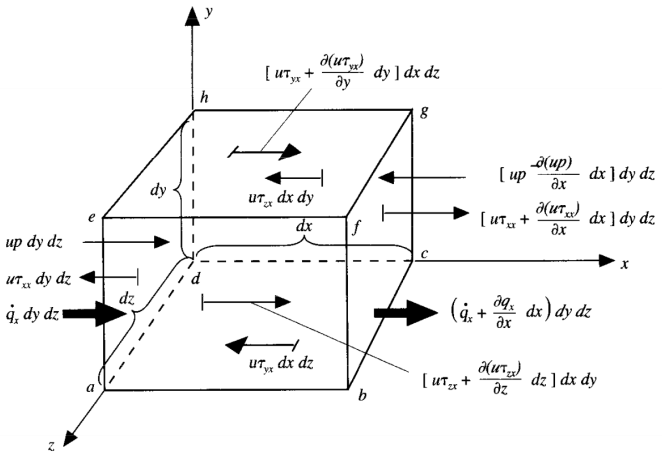


Figure 3.3: Energy fluxes associated with an infinitesimally small, moving element in x direction. *(Adapted from [2].)*

Net flux of heat into element has two sources. Volumetric heating $\dot{q}$ is caused by absorption or emission of heat radiation. And conduction causes heat transfer across the surfaces ($\dot{q}_x$, $\dot{q}_y$, $\dot{q}_z$). On conductivity is applied Fourier's law

$$\dot{q}_i = -k\frac{\partial T}{\partial i} \qquad i = x, y, z. \tag{3.25}$$

Net flux of heat into element is then sum of these two effects (described mathematically

on Figure 3.3) in all the directions.

$$\dot{Q} = \left[\rho\dot{q} + \frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(k\frac{\partial T}{\partial y}\right) + \frac{\partial}{\partial z}\left(k\frac{\partial T}{\partial z}\right)\right] \mathrm{d}x\ \mathrm{d}y\ \mathrm{d}z. \tag{3.26}$$

Rate of work originates from body and surface forces exerted on the element. It can be shown (see [1]) that work caused by a force acting on a moving element is equal to the product of the force and the component of velocity in the direction of the force. Hence for the body forces acting on the element moving with velocity **V** in vector form one can write:

$$\rho\mathbf{f} \cdot \mathbf{V}(\mathrm{d}x\ \mathrm{d}y\ \mathrm{d}z).$$

The work done by the surface forces is simply the particular force multiplied by according velocity component. E. g. pressure work in x direction:

$$\left[up - \left(up + \frac{\partial(up)}{\partial x}\mathrm{d}x\right)\right] \mathrm{d}y\ \mathrm{d}z = -\frac{\partial(up)}{\partial x}\mathrm{d}x\ \mathrm{d}y\ \mathrm{d}z.$$

The total net flux of the work is then summary of all particular works in all the directions.

$$\begin{aligned} W = \Bigg[&-\left(\frac{\partial(up)}{\partial x} + \frac{\partial(vp)}{\partial y} + \frac{\partial(wp)}{\partial z}\right) + \frac{\partial(u\tau_{xx})}{\partial x} + \frac{\partial(u\tau_{yx})}{\partial y} \\ &+ \frac{\partial(u\tau_{zx})}{\partial z} + \frac{\partial(v\tau_{xy})}{\partial x} + \frac{\partial(v\tau_{yy})}{\partial y} + \frac{\partial(v\tau_{zy})}{\partial z} + \frac{\partial(w\tau_{xz})}{\partial x} + \frac{\partial(w\tau_{yz})}{\partial y} \\ &+ \frac{\partial(w\tau_{zz})}{\partial z}\Bigg] \mathrm{d}x\ \mathrm{d}y\ \mathrm{d}z + \rho\mathbf{f} \cdot \mathbf{V}(\mathrm{d}x\ \mathrm{d}y\ \mathrm{d}z). \end{aligned} \tag{3.27}$$

The energy equation can be written as:

$$\frac{\mathrm{D}E}{\mathrm{D}t} = \dot{Q} + W. \tag{3.28}$$

Now the equations (3.24), (3.26) and (3.27) are substituted into equation (3.28) the final partial differential non-conservation form of energy equation is obtained.

$$\begin{aligned} \rho\frac{\mathrm{D}}{\mathrm{D}t}\left(e + \frac{V^2}{2}\right) &= \rho\dot{q} + \frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(k\frac{\partial T}{\partial y}\right) + \frac{\partial}{\partial z}\left(k\frac{\partial T}{\partial z}\right) \\ -\left(\frac{\partial(up)}{\partial x} + \frac{\partial(vp)}{\partial y} + \frac{\partial(wp)}{\partial z}\right) &+ \frac{\partial(u\tau_{xx})}{\partial x} + \frac{\partial(u\tau_{yx})}{\partial y} + \frac{\partial(u\tau_{zx})}{\partial z} + \frac{\partial(v\tau_{xy})}{\partial x} + \frac{\partial(v\tau_{yy})}{\partial y} \\ &+ \frac{\partial(v\tau_{zy})}{\partial z} + \frac{\partial(w\tau_{xz})}{\partial x} + \frac{\partial(w\tau_{yz})}{\partial y} + \frac{\partial(w\tau_{zz})}{\partial z} + \rho\mathbf{f} \cdot \mathbf{V}. \end{aligned} \tag{3.29}$$

3.2.4 Summary

The above derived equations are base of CFD and even though user does not have to put them into most programs it is necessary to have at least a little overview of them to get a better insight into the problems. The governing equations at given form are not considering any reaction occurring as the fluid flows. To consider the reaction simply source

reaction terms are added only to the energy and continuity equation since the momentum equation is not significantly influenced by the reaction.

In ordinary physics the three equations for momentum (3.14 – 3.16) are called Navier-Stokes equations. In the CFD practice on the other hand all the equations for viscous flow (derived above) are called Navier-Stokes. Equations describing the inviscid flows are simplification of the above derived equations. Terms for diffusion, conduction, dissipation and transport phenomena of viscosity terms are neglected. This set of simplified equation is in CFD community called Euler equations. Continuity equation does not contain any of these terms so it stays the same as equation (3.4). The momentum equations (3.14 – 3.16) get much simpler, as

$$\rho\frac{\mathrm{D}u}{\mathrm{D}t} = -\frac{\partial p}{\partial x} + \rho f_x, \tag{3.30}$$

$$\rho\frac{\mathrm{D}v}{\mathrm{D}t} = -\frac{\partial p}{\partial y} + \rho f_y, \tag{3.31}$$

$$\rho\frac{\mathrm{D}w}{\mathrm{D}t} = -\frac{\partial p}{\partial z} + \rho f_z. \tag{3.32}$$

And also energy equation (3.29) gets much simpler.

$$\frac{\partial}{\partial t}\left[\rho\left(e+\frac{V^2}{2}\right)\right] + \nabla\cdot\left[\rho\left(e+\frac{V^2}{2}\right)\mathbf{V}\right] = \rho\dot{q} - \left(\frac{\partial(up)}{\partial x} + \frac{\partial(vp)}{\partial y} + \frac{\partial(wp)}{\partial z}\right) + \rho\mathbf{f}\cdot\mathbf{V}. \tag{3.33}$$

3.3 Discretization schemes

For complex problems the direct analytical solution of the governing equations is not possible, so the numeric solution is done. Numeric solutions are done by computers which operate on discrete domain and are not able to handle continuous problems. The discretization of partial differential equations is then of a great importance.

The discretization is in very simple terms transformation of the partial differential equation into set of algebraic equations. Three basic schemes are used in todays practice. For simple geometries all three give the same solution matrix, for more complicated or irregular geometries the solution can differ very significantly.

Finite difference method (FDM) was derived and published first. [19] FDM substitutes the partial derivatives by series expansion (usually Taylor's expansion), where the high order terms are neglected. This method is easy to use for simple geometries but is not well suitable for complicated irregular geometries. And more importantly it does not conserve mass, momentum and energy.

Most widely used is **finite volume method** (FVM). In this scheme the domain is split into finite number of small volumes. The governing equations are integrated over the volumes assuming linear variation of dependent variables. Results are representation of each volume element in the center of the volumes and the values at edges and faces are obtained by interpolation between neighboring volumes. The only limitation for the volumes is that they must not overlap, otherwise they can have any shape and size. For irregular cell shapes

(e. g. tetrahedral, polyhedral) is the bookkeeping demanding, but since the scheme is used for some time, the solvers are well developed and are reliable. This scheme conserves mass, energy and momentum by definition, hence the very wide use (approximately 80% of the available codes). On the other hand, the method suffer from numerical diffusion, when simple numerics are used.

Finally the **finite element method** (FEM) is gaining importance. From physical point of view it is not very beneficial, because the discretized terms do not have a physical importance. On the other hand the scheme is very beneficial from programming point of view, since it is required the same effort for any geometry. It offers the highest accuracy and it is excellent for diffusion problems. Although it is not well suited for fast flows and it is relatively slow for large problems. This method integrates the governing equations over element or volume after having been multiplied by a weight function (Galerkin's method of weighted residuals). [23] Dependent variables are represented by shape functions which differ depending on used cell shapes.

3.3.1 Finite volume vs. Finite element

Both methods are using integral forms of the governing equations but integrating them over different control volumes. The finite volume method uses weighted residual procedure, where the residual over whole domain is required to be zero, hence to close the balances (conserve energy, momentum and mass). Finite element does not guarantee the conservation for all applications, since the solution criteria is to optimize for other quantities than that to make the residuals zero.

Finite volume method takes artificial volume with desired node as center and then uses Gauss's (divergence) theorem for transforming the volume integrals into surface integrals. Which are then split in terms of fluxes, hence the 3D cases are transfer to infinite number of 1D problems [11] which are then solved.

Finite element method on the other hand uses linear or quadratic shape functions for the relations between two nodes for which the 'exact' solution is then found.

Idelsohn and Oñate [11] are showing that in simple cases both methods are actually equal and give the same results. Fallah *et al.* [7] makes a point that finite volume method can be viewed as special case of finite element with non-Galerkin weighting. The Galerkin's approach for finite elements uses weighting function W which is then defined as $W_i = N_i$ where N_i are shape functions corresponding to node i. The finite volume procedure can be viewed as special form with weighting function

$$W = \begin{cases} 1 & \text{in the control volume,} \\ 0 & \text{elsewhere.} \end{cases} \tag{3.34}$$

In general it can be said that one method is better for some application and the second for other applications. Finite elements perform better for mechanical properties problems and incompressible flows at low Reynolds numbers. Finite volumes deal better with compressible flows and with flows at high Reynolds numbers. There are new finite elements solvers and algorithms developed in commercial sector which should be able to deal with high Reynolds numbers and still conserve quantities and have better accuracy then finite volume methods.

3.4 Mesh

A mesh (or grid) is basically a discretized representation of the geometry for which the calculations are done. In finite volume method the mesh cells represent the volumes for which the discretized solution is made. The properties of the mesh influence the speed of the convergence, time needed to solve the problem and the accuracy of the solution. It also influences the usage of the RAM memory. In general it can be said, that the finer (denser) the mesh is, the more accurate, time consuming and RAM using the solution is. Few properties are used to describe the mesh.

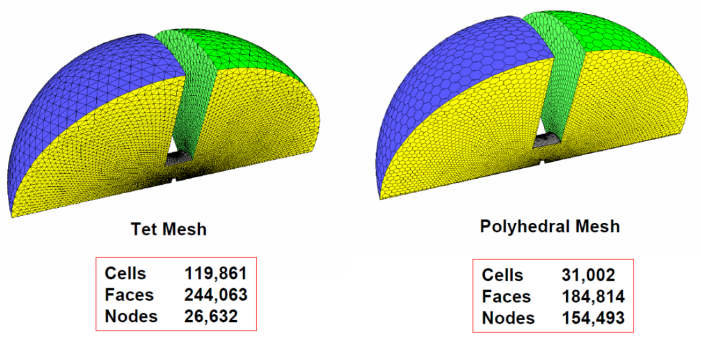


Figure 3.4: Comparison of two meshes. *(Taken from [12].)*

Density (granularity) of the cells describes how many cells are in an unit volume. Density can be constant throughout the whole mesh, but it is very common to vary the density. Near the walls or at regions where big gradients are expected is used a finner mash, than in the fluid body. However change of the size of the cells has to be smooth. The neighbor cells size variation should not exceed 20%.

Other properties are related to the size and shape of the cells. When only two dimensional calculations are done triangular, convex quadrilateral or hexagonal shapes are used. In three dimensional cases the most common shapes are tetrahedron, hexahedron and half-hexahedron or polyhedrons (honeycombs). These shapes and more are shown on Figure 3.5.

Skewness (asymmetry) of the cells is in ideal case 0 in worst case 1. There are two methods to obtain skewness. First is based on equilateral shape of the cells.

$$\text{skewness} = \frac{\text{optimal size of cell} - \text{actual size of cell}}{\text{optimal size of cell}}, \tag{3.35}$$

where size means area (in 2D cases) or volume (in 3D cases). To illustrate each term see Figure 3.6. This calculation method is applicable only on triangles and tetrahedrons.

Other scheme is applicable to all shapes (necessary for pentahedrons) and it is based on difference from normalized angle.

$$\text{skewness} = \max\left[\frac{\theta_{\max} - \theta_e}{180° - \theta_e}; \frac{\theta_e - \theta_{\min}}{\theta_e}\right], \tag{3.36}$$

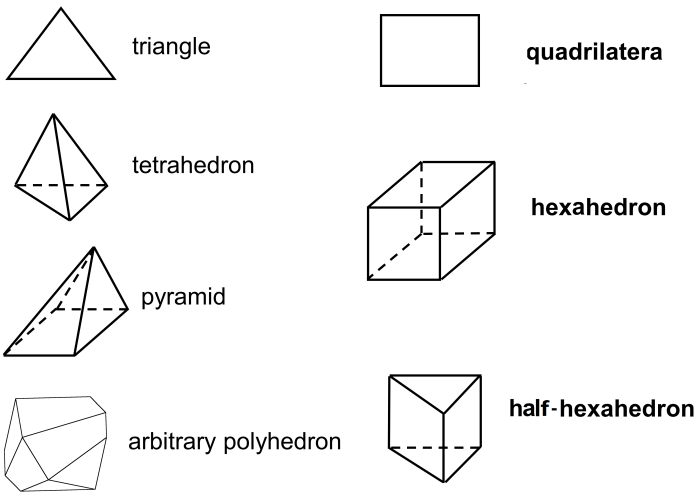


Figure 3.5: Shape of the cells. *(Adapted from [12].)*

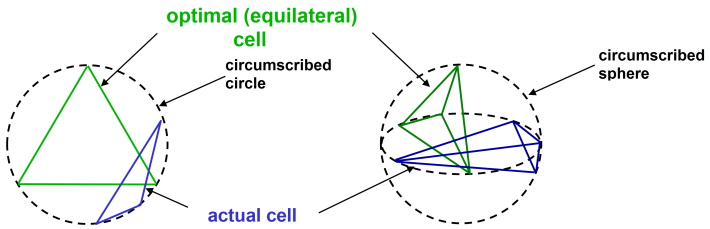


Figure 3.6: Skewness based on equilateral shape of the cells. *(Adapted from [12].)*

where $\theta_{\max}$ and $\theta_{\min}$ are maximal and minimal angle in the cell respectively and θ_e is angle in equiangular cell ($60°$ for triangle, $90°$ for cube).

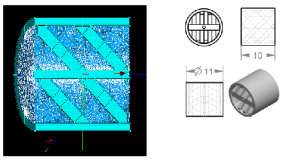
Another property of the mesh is aspect ratio, which is ratio between edges/faces of the cells. Ideal value is 1 which means equilateral shape. For one-dimensional flows it is not necessary to keep the aspect ratio at 1, but for poly-dimensional flow it is highly recommended.

3.5 Pre-processing

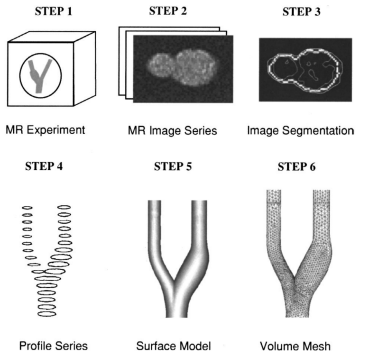
In theory pre-processing is everything that is preparing the numerical problem, though in practice only geometry and mesh definition is referred to as pre-processing.

In technical practice it is usual that the geometry of a model is defined from the design drawings of the equipment. This requires the design drawings to be done in a form treatable by a computer, e. g. drawn in designing software such as AutoCAD, Pro/Engineer

or other (e. g. Figure 3.7a). The image analyses is the newest development in geometry definition domain. This approach is applicable in e. g. medicine (see Figure 3.7b).



(a) Pro/Engineer drawing of a desired equipment.



(b) Summary of steps used to construct a mesh from magnetic resonance images. *(Taken from [16].)*

Figure 3.7: Geometry representation.

Next stage of the pre-processing is the mesh creation from the given geometry. For very simple geometries it is usually possible to write the source code of the mesh directly in the modeling software language. Things get more complicated with more complex or irregular geometries. For this purpose there is usually a software tool implemented which is capable of generating a mesh by itself while given e. g. surface representation of the geometry. Alternatively there is possibility to import the mesh from the software specialized in mesh generation. For very complicated geometries the last option is usually the best. The meshing software is in some cases part of the CFD package (Gambit in Fluent package) or it is the third party software (Netgen, Salome, etc.).

3.6 Residence time distribution

The residence time distribution (RTD) of a chemical reactor is a probability distribution function that describes the amount of time that a fluid element could spend inside the reactor. There are two main functions describing the residence time distribution as proposed by Danckwerts [6].

The first function is **E-function**. Integral of the function $E\mathrm{d}t$ between t_1 and t_2 gives the fraction of material that has "age" between t_1 and t_2. The graphical representation of the E-function is called differential residence time distribution curve and it is defined as:

$$\int_0^\infty E(t)\mathrm{d}t = 1. \tag{3.37}$$

In fact it is a response at the outlet of a system to an impulse of a tracer at the inlet.

F-function is called cumulative distribution and it is defined as the fraction of material that spends less than time t_1 in the system [24]:

$$F(t_1) = \int_0^{t_1} E(t)\mathrm{d}t = 1. \tag{3.38}$$

Graphical representation of the dependence of the F-function value on time (or normalized time by the average residence time) is referred to as integrated residence time distribution curve. It can be obtained as response at the outlet of a reactor to the step change at the inlet.

It was found by Danckwerts [6] that for ideal reactors with perfect mixing the F- and E-functions obey relations:

$$F(t) = 1 - e^{-t/\theta}, \tag{3.39}$$

$$E(t) = \frac{1}{\theta} e^{-t/\theta}, \tag{3.40}$$

where θ is the ideal plug flow residence time defined by Eq. 3.42. These equations are applicable to ideal continuous stirred tank reactors. When series of CSTRs are considered the residence time distribution curves tend to become closer to the plug flow as the number of tanks increases (see Figure 3.8. This suggests that pipe reactors can be approximated by series of CSTRs. Péclet number and Bodenstein number are tightly connected with this phenomenon. The definition of the dimensionless quantities will be given further in the text.

The average residence time of the fluid element is given as a first moment of the age distribution and it can be calculated:

$$\bar{t} = \frac{\int_0^\infty tE(t)\mathrm{d}t}{\int_0^\infty E(t)\mathrm{d}t} = \int_0^\infty tE(t)\mathrm{d}t. \tag{3.41}$$

For ideal plug flow is the average residence time equal to θ, where:

$$\theta = \frac{V}{\dot{V}}, \tag{3.42}$$

where V is the volume of the reactor and $\dot{V}$ is the volumetric flow of the fluid.

So called second central moment of the residence time distribution is very important for comparison of the RTD curves for various systems. The second central moment indicates the variance, σ^2, of the RTD around the mean value:

$$\sigma^2 = \int_0^\infty (t-\theta)^2 E(t)\mathrm{d}t. \tag{3.43}$$

3.7 Important dimensionless quantities

Dimensionless quantities appear through out the engineering science. When it comes to the fluid flow the most important and most generally known is **Reynolds number**. Reynolds

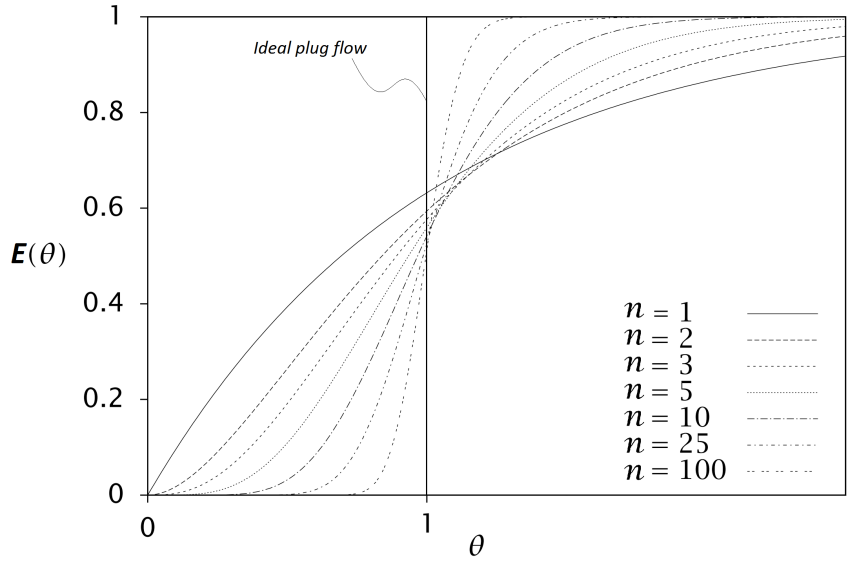


Figure 3.8: Integrated residence time distribution curves for increasing number of CSTRs in series.

number as used in this work is defined as:

$$Re = \frac{U d_H \rho}{\eta} = \frac{U d_H}{\nu}, \tag{3.44}$$

where U is the mean velocity, ρ is the density, η is the dynamic viscosity, ν is kinematic viscosity and d_H is a hydraulic diameter which can be found as:

$$d_H = \frac{4V}{A} = \frac{4 \cdot \text{free volume}}{\text{wetted surface}}. \tag{3.45}$$

Another very important measure is the **Schmidt number** which compares the effects of viscous diffusion and mass diffusion. In another words it shows which phenomenon — diffusion of mass or diffusion of momentum — is dominating. It is defined as:

$$Sc = \frac{\eta}{\rho D} = \frac{\nu}{D}, \tag{3.46}$$

where D is the mass diffusivity.

For heat transfer it used the **Prandtl number**, which is an equivalent to the Schmidt number. This number is a ratio of diffusion of momentum and diffusion of heat. It is defined as:

$$Pr = \frac{\eta}{\rho \alpha} = \frac{\nu}{\alpha} = \frac{c_p \eta}{k}, \tag{3.47}$$

where α is the thermal diffusivity, c_p is the specific heat and k is the thermal conductivity.

Péclet number is a comparison between the advection and diffusion of desired physical property. The axial Péclet number for tube is defined as:

$$Pe_{ax} = \frac{Ud_R}{D_{ax}}, \tag{3.48}$$

where d_R is the diameter of a tube and D_{ax} is mass diffusivity in axial direction. For tube reactors it was found that the axial Péclet number can be approximated as a function of Reynolds and Schmidt numbers [5]:

$$\frac{1}{Pe_{ax}} = \begin{cases} \dfrac{1}{Re \cdot Sc} + \dfrac{Re \cdot Sc}{192} & \text{for } 1 < Re < 2000 \text{ and } 0.23 < Sc < 1000, \\ \dfrac{3 \cdot 10^7}{Re^{2.1}} + \dfrac{1.35}{Re^{1/8}} & \text{for } 3 \cdot 10^3 < Re < 10^5. \end{cases} \tag{3.49}$$

In European literature, especially German, the Péclet number is often related to the **Bodenstein number** and it has its benefits [5]. As reader can see in Equation 3.48 the axial Péclet number is related to the diameter of the tube. On the other hand Bodenstein number compares the same quantities with respect to the length of the tube, L, so it can be calculated as:

$$Bo = \frac{UL}{D_a x} = Pe_{ax} \cdot \frac{L}{d_R}. \tag{3.50}$$

The extreme cases of the Bodenstein number are:

- continuous stirred tank reactor (CSTR) $Bo = 0$;
- plug flow reactor (PFR) $Bo \to \infty$.

The main benefit of the Bodenstein number is, that it was found to be very simple parameter which can be used to find number of CSTRs is series which approximate a tube reactor in order to obtain same residence time distribution. The number of CSTRs can be found:

$$n = \frac{Bo}{2}. \tag{3.51}$$

In literature [5] are also presented simple correlations for the variance of the residence time distribution normalized by the average residence time distribution, σ_θ^2 related to the Bodenstein number. The approximation depends on the system. Three types of the systems are proposed — opened system, closed system and semi-opened system. These three systems are captured on Figure 3.9.

The approximations for opened, semi-opened and closed systems are respectively:

$$\sigma_\theta^2 = \frac{2}{Bo} + \frac{8}{Bo^2}, \tag{3.52}$$

$$\sigma_\theta^2 = \frac{2}{Bo} + \frac{3}{Bo^3}, \tag{3.53}$$

$$\sigma_\theta^2 = \frac{2}{Bo} - \frac{8}{Bo^2}[1 - \exp(-Bo)]. \tag{3.54}$$

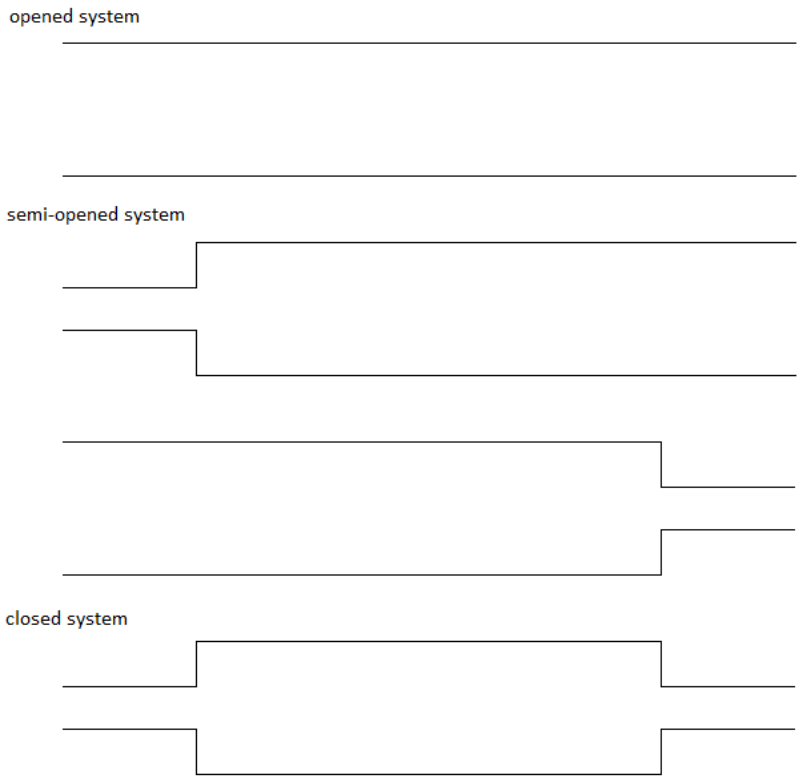


Figure 3.9: System types for approximation of the RTD variance.

Chapter 4

Modeling

This chapter will take a brief insight into the modeling part of the solution. It will describe the software used for solution and also the modeled geometry and conditions.

4.1 OpenFOAM

"The OpenFOAM® (Open Field Operation and Manipulation) CFD Toolbox is a free, open source CFD software package produced by OpenCFD Ltd. It has a large user base across most areas of engineering and science, from both commercial and academic organisations. OpenFOAM has an extensive range of features to solve anything from complex fluid flows involving chemical reactions, turbulence and heat transfer, to solid dynamics and electromagnetics." [13]

OpenFOAM® is Linux based toolbox which includes pre- and post-processing, meshing and solution of the problems. It is based on C++ routines, which makes it very powerful tool since anybody can adjust the routines for their particular need, only requirement is a knowledge of C++ programing language. It does not have any graphical interface and it runs via Linux terminal so a special folder structure is required. The description of all the folders and files is very precisely done in the user guide [14] and it is unnecessary to get into details on this level. But brief description is needed to get at least a small insight into problem definition.

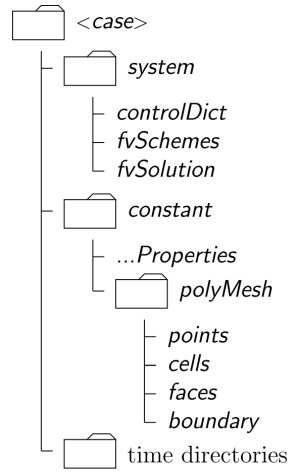


Figure 4.1: Required folder structure of a case in OpenFOAM®. [14]

In folder `system` three main files are located. In these files the application is defined. The file `controlDict` contains general information about the solution, such as application (solver), initial and end time and time step of solution, interval

in which results are written, etc. There are many standard solvers available in OpenFOAM, the difference between them is in system they can apply to [14, p. U-83–U-87]. In dictionary `fvSchemes` user has to define numerical schemes for terms like derivatives, divergences, Lapacians, etc. File `fvSolution` controls the equation solvers, tolerances and algorithms used for solution. Also a convergence criteria is set here.

Folder `constant` contains several properties, e. g. when turbulent modeling is done, it contains properties for Reynolds averaging, transportation properties, etc. Most importantly in this folder is located directory `polyMesh` which contains information about the mesh.

The time directories are filled with results by the solver itself in intervals defined in `controlDict` as the solution proceeds. The directory `0` contains information about the initial and boundary conditions and it has to be defined by the user.

This is the basic structure, which may not vary for different application. There might be some additional files and folders, but the basic ones are always there.

The case is executed by series of commands in the terminal. The sequence is dependent on the application and command to actually execute the solution is same as the name of the solver used. So e. g. when solver `icoFoam` is chosen to be used, user just opens the terminal in the case directory (after setting all the necessary properties and defining the mesh) and executes command `icoFoam` and the problem is solved.

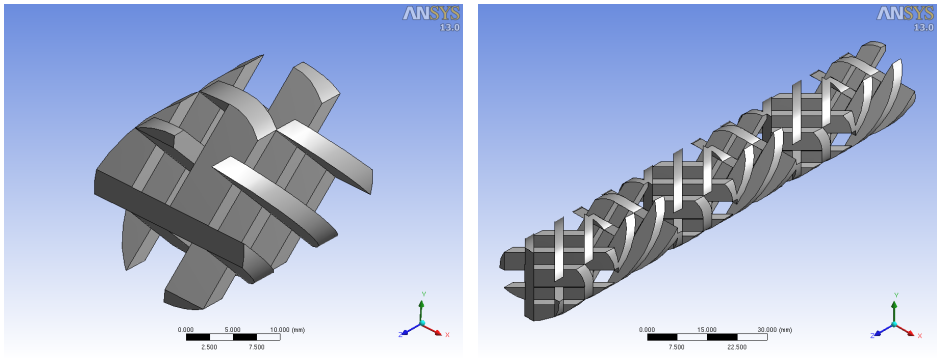
4.2 Geometry

The considered geometry was based on Fluitec CSE-X4 mixing elements. There are two main reasons for this choice. First, the CSE-X4 mixing element is the simplest mixing element supplied by Fluitec and second the experimental data for residence time distribution are available.

The CSE-X4 mixing element contains four cross-bars and three parallel cross-bars. The modeled geometry has a diameter of 21 mm and the baffle width of 2.2 mm (see Figure 4.2a). This geometry was constructed in Ansys® DesignModeler. Since this elements are not supplied individually but as a sequence of six elements, the modeled geometry was constructed accordingly (see Figure 4.2b).

Since technical drawings were not available for the CSE-X4 mixing element, it was necessary to measure the dimensions by caliper and than reproduced into the drawing. The drawn element contained so called short edges, sliver faces and discontinuities which were required to be repaired. Hence cleanup and repair tools provided by DesignModeler were used. These tools automatically detect and repair the problematic parts of geometry. The short edges could cause some artificial obstacle in flow and could cause false flow changes.

As the first set of models for various properties was solved, it was found that flow detachment and backflow appears. Description of the backflow issue and why it is necessary to eliminate it is given in Chapter 5. In order to avoid this problem several adjustments of the model as well as of the modeled geometry were tested. It appeared to be the best option to add a tube with a small diameter at the end of the reactor which unifies the flow direction and the volumue addition is insignificant compare to the reactor, so the RTD is not influenced. Also this modeled design is a better representation of the experiments,



(a) One CSE-X4 mixing element. **(b)** Six CSE-X4 mixing elements.

Figure 4.2: Geometry of Fluitec CSE-X4 mixing elements.

since in laboratory the measurements are also done in a thinner tube behind the reactor and not directly after the last mixing element. The modeled geometry is shown on Figure 4.3. The figure shows a cutting plane through the center of the reactor. The modeled geometry consists of 6 mixing elements which have a 0.8 mm gap between each other. After the last baffle 2 mm free space is left and then a 10 mm tube with a diameter of 5 mm is added. The evaluation of the residence time distribution was done at the end of the thin tube.

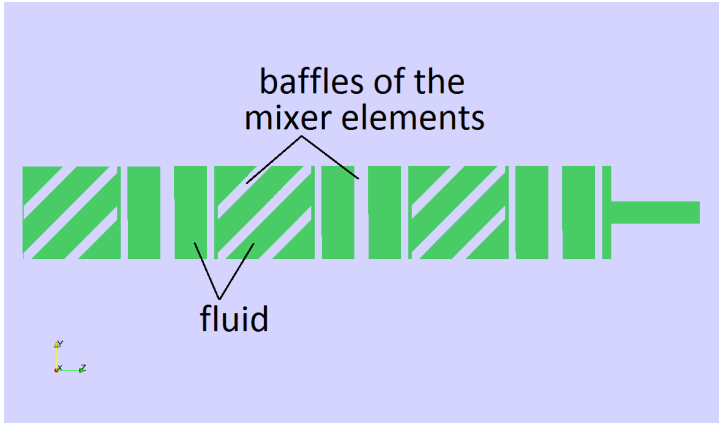


Figure 4.3: Geometry used for the modeling.

4.3 Mesh

The mesh of the desired geometry was constructed in Ansys® Gambit. There were used size functions ensuring denser mesh near walls and around corners where big changes of variables were expected.

The final mesh contained around 28 millions of tetrahedral cells. This resolution was chosen in order to avoid grid dependent solution. In order to decrease the solution time, the mesh was transfered from tetrahedral cells to polyhedral. This transfer resulted in decrease of number of cells, so the final mesh has 6.3 millions of polyhedral cells. This resolution gives in average 1 million of cells per mixing element which is reasonable, it guarantees grid independent solution (according to grid dependence analysis performed during earlier research in industry) in reasonable solution time.

For laminar flows it might be possible to take advantage of the symmetry of the geometry. The geometry has two symmetry planes, XZ- and YZ-plane. It should be possible to use only one quarter of the geometry, hence of the mesh, which would cause the decrease the number of the calculated cells to one quarter. For a lack of time it was not possible to test if the solution while using symmetry planes would be in agreement with complete solution without the symmetry planes. So it was decided to use the full geometry.

4.4 Model description

The residence time distribution analysis was performed in two steps. First the steady state flow field was calculated for given flow conditions and subsequently a passive scalar transport was calculated on the obtained velocity field and evaluated at the outlet boundary. This procedure copies the experimental method, where a tracer (passive scalar) is injected into a flowing liquid and its concentration in time is measured on the outflow from the reactor.

The flow field was obtained for wide range of Reynolds numbers in laminar flow regime. The laminar regime was picked because the reactors are operated in laminar regime so the experimental data are available only for laminar flow. For the scalar transport range of Schmidt numbers was chosen such that desired range of Bodenstein numbers were investigated.

In the following text settings of the numerical solution will be described briefly. For solution of the flow field and the scalar transport was necessary to use two different settings, so two sets of files `fvSolution` and `fvScheme` were used. These files are showed in Appendix A and B respectively.

For solution of the flow field was used solver `simpleFoam`. This solver is a steady-state solver for turbulent flow, according to the user guide [14]. But it is possible to use it also for laminar modeling, when the turbulence is switched off in the solver. This solver deals with pressure `p` and velocity `U`.

The pressure equation is solved with generalized geometric-algebraic multi-grid solver (`GAMG`) with Gauss-Seidel smoother with tolerance 10^{-7} and relative tolerance 10^{-2} which specifies convergence of one iteration. The convergence criteria is specified to be 10^{-7}. One corrector for pressure is applied for non-orthogonal cells. This option is set to be on the safe site since the mesh orthogonality is very good.

The preconditioned bi-conjugate gradient solver (`PBiCG`) with (asymmetric) diagonal incomplete-LU preconditioner was chosen for velocity solution. The tolerance for one iteration was set to 10^{-8} and relative tolerance to 10^{-3}. The solution is considered converged when residuals of each direction component of velocity becomes lower than 10^{-7}.

Relaxation of the solution is set to be 0.3 for pressure and 0.7 for velocity. These values were taken according to advice from more experienced users and seemed to be reasonable.

The solution of the tracer concentration T was performed also by preconditioned bi-conjugate gradient solver (`PBiCG`) with (asymmetric) diagonal incomplete-LU preconditioner. Since tracer concentration requires transient solution the relative tolerance is specified to be 0 forcing the solution to converge at each time step, where convergence criteria for one time step is specified to be 10^{-6}. Also for tracer concentration there is set one non-orthogonal corrector to avoid eventual issues with mesh non-orthogonality.

Chapter 5

Results and discussion

As the first set of models for various properties was solved, it was found that slight flow detachment appears directly after the last baffle, even for small Reynolds numbers, which causes a back-flow problem. It is necessary to avoid the back-flow at the outlet plane to be able to calculate representative residence time distributions. If one would calculate RTD with back-flow one would not know the residence time of the volumes that are flowing in at the outlet boundary. This issue is displayed on Figure 5.1 where the red zone behind the mixing element represents the region of pressure lower than pressure at the outlet boundary. This pressure distribution causes that the fluid flows into the red region.

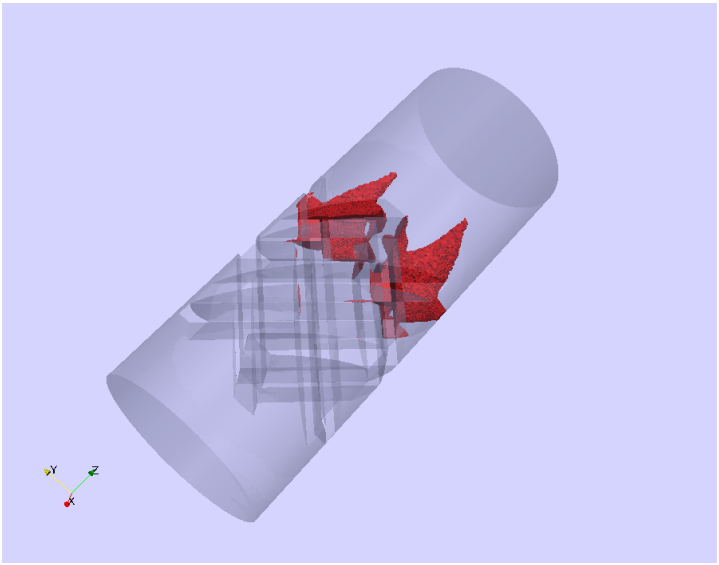


Figure 5.1: Flow detachment zone illustration.

As was mentioned above first the velocity fields for various flow conditions were calculated. Since the flow conditions were chosen such that the flow is in the laminar regime with low Reynolds numbers, the solution is linear in the dependent variables. This means that when one parameter is changed, the solution changes linearly. That results in self similarity of the results for different modeled cases while normalized by the maximal velocity. One representative result is presented in Figure 5.2. The result is shown on a cutting plane through the reactor. The fluid flows in positive z-direction (i. e. from left to right). The thinning at the end of the reactor is removed from the view in order to maximize the displayed region.

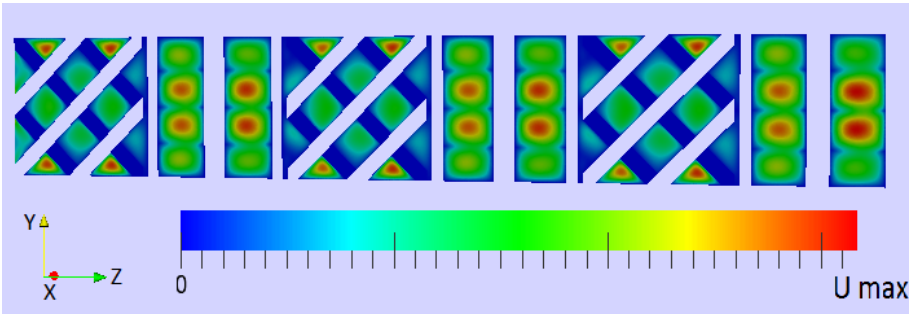


Figure 5.2: Isocontours of the velocity magnitude in the cutting plane through the middle of the reactor. Flow direction from left to right.

On Figure 5.3 is captured comparison between experimentally measured RTD curve and RTD curve computed by CFD model in OpenFOAM. The computed curve is in agreement with the experimental data. The slight difference is very small and can be caused by various factors. The factors can be such as:

- uncertainty of the measurement;
- numerical diffusion;
- inaccuracy of the Bo number correlation.

The experimental data were provided without further specification of the experimental system so the only comparison measure was the Bodenstein number. The range of Bodenstein numbers covered by the experiment was very narrow. Since all the data were provided very late, it was impossible (for the lack of time) to do further calculations in order to compare the model for more modeled cases. So only one modeled case fit in the experimental range.

The longer tail in the experimental data suggests that there is a dead volume inside of the static mixer. Even though the model was developed to perform RTD analyses it is possible to spot dead volume in some regions of the geometry.

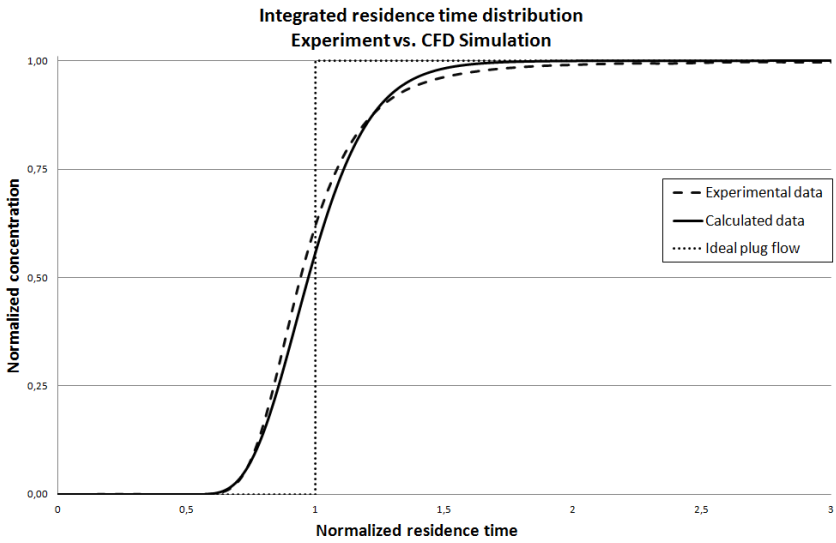


Figure 5.3: Comparison of experimental data and results of CFD model.

The Figure 5.4 shows how the reactor is filled with tracer after 1 residence time (time normalized by residence time of ideal plug flow). The blue region shows the trace concentration above 95 %.The red circle indicates the region where the mixing is delayed.

Although, the dead volume was found the extension of the tail part of the integrated RTD curve suggests that there is more dead volume in the real system. In order to find all of the dead volume it would be necessary to obtain more information of the experimental system and develop a more precise model of the system. Since all of this information were not available by the end of the project, it was impossible to obtain more precise data.

The results showed that the residence time distribution is dependent only on the product $Re \cdot Sc$ regardless on what particular values Re and Sc have which is in agreement with the theory. It can be assumed that similar correlation as Eq. 3.49 between the Péclet number and the product $Re \cdot Sc$ can be found also for the static mixers. Since it was possible to solve the model for narrow range of Bodenstein numbers it would not be reliable to find this correlation with the available calculations.

On Figure 5.5 are displayed the integrated residence time distribution curves calculated by the CFD model. It can be observed the general trend of narrower residence time distribution for higher Bodenstein numbers ($Bo_1 > Bo_2 > Bo_3 > Bo_4$). This trend is also in agreement with the theory [5, p. 334] and with the experimental data.

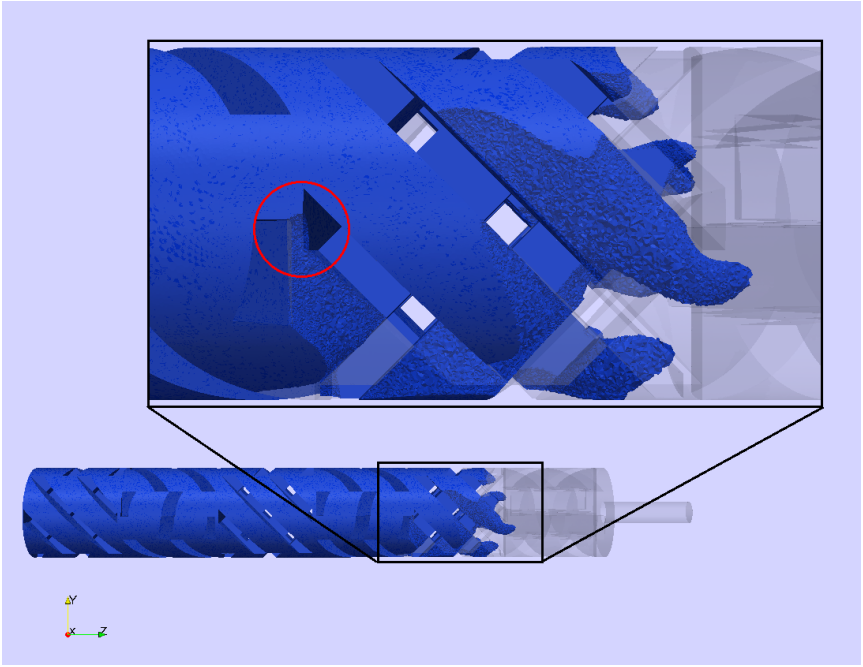


Figure 5.4: Reactor filled with tracer after 1 residence time.

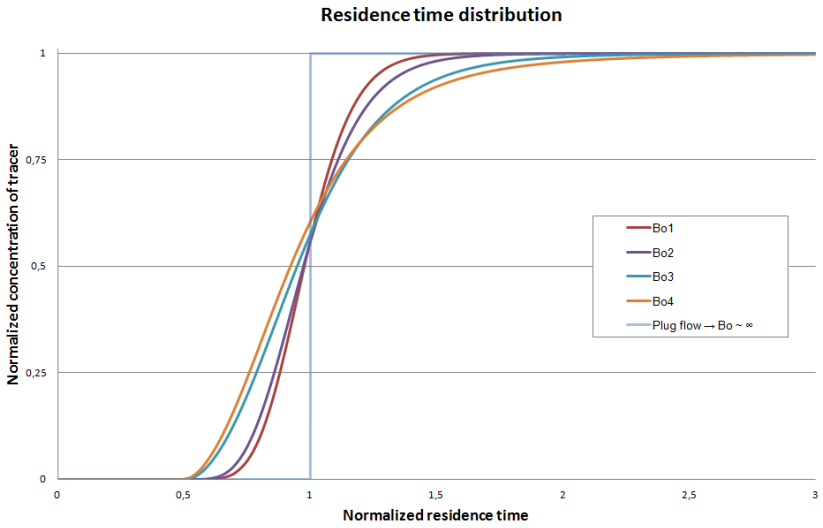


Figure 5.5: Residence time distribution curves for different Bodenstein numbers.
$Bo_1 > Bo_2 > Bo_3 > Bo_4$

Chapter 6

Conclusion

In this thesis a residence time distribution analysis was performed for Fluitec static mixer reactor with CSE-X4 mixing elements. The tracer injection experiment was reproduced by computation fluid dynamics modeling. The model contained of two parts, first velocity fields were calculated for various laminar flow conditions and then transport of the passive scalar was evaluated. The passive scalar change (tracer injection) was performed as a step change at the inlet boundary and it was evaluated at the outlet from the reactor. It gave integrated residence time distribution curves which were compared with the experimental data.

CFD modeling revealed that even for flows with low Reynolds numbers the flow detachment occurs at the leaving edge of the mixing element. In order to correctly calculate the residence time distribution it was necessary to avoid the backflow at the outlet boundary. The geometry modification, by adding a thin tube at the end of the geometry, was chosen to be the best solution for unifying the flow direction through the outlet boundary.

The resulting integrated residence time distribution curve of one case was compared with available experimental data. The results showed good agreement between the experiment and the model. The slight difference between the curves suggest that the model did not capture all the dead volume occurring in the reactor. More information about the experiment would be needed in order to refine the model accordingly to obtain better agreement.

It was found that the width of the RTD is dependent only on product $Re \cdot Sc$ and not on the particular values of the two. This suggests that basic theory, such as Equations 3.49 and 3.52 – 3.54, of the pipe flow is applicable also to the static mixer reactors and similar correlations between variance of the RTD and the Bodenstein number, or the product $Re \cdot Sc$, could be found.

The time schedule and computation capacity did not allow to calculate the RTD for broader range of Bodenstein numbers. It would not be then reliable to find a correlation between the product $Re \cdot Sc$ and variance for the static mixer.

In future work it may be beneficial to investigate possibility of using symmetry planes in the modeled geometry, hence decreasing significantly the computation time and com-

putation power demands. In order to use the model in future research, it is necessary to obtain a better description of the experimental procedure to setup the modeled conditions accordingly. Then it would be possible to verify the model confidently.

When the flow model is well developed and verified, it will be possible to implement reactions and population balances. This would result in complex model of the whole reaction process, which could be used to predict the behavior of the reactive mixture under various reaction conditions.

Bibliography

[1] J. D. Anderson. *Fundamentals of aerodynamics*. McGraw-Hill, 2 edition, 1991.

[2] J. D. Anderson. *Computational fluid dynamics: The basics with applications*. McGraw-Hill, 1995.

[3] J. Aubin, D. F. Fletcher, J. Bertrand, and C. Xuereb. Characterization of the mixing quality in micromixers. *Chemical Engineering & Technology*, 26(12):1262 – 1270, 2003.

[4] J. Aubin, D. F. Fletcher, and C. Xuereb. Design of micromixers using CFD modelling. *Chemical Engineering Science*, 60(8 – 9):2503 – 2516, 2005.

[5] M. Baerns, H. Hofmann, and A. Renken. *Chemische Reaktionstechnik*. Thieme, Stuttgart, 2nd edition, 1992.

[6] P. V. Danckwerts. Continuous flow systems: Distribution of residence times. *Chemical Engineering Science*, 2(1):1 – 13, 1953.

[7] N.A. Fallah, C. Bailey, M. Cross, and G.A. Taylor. Comparison of finite element and finite volume methods application in geometrically nonlinear stress analysis. *Applied Mathematical Modelling*, 24(7):439 – 455, 2000.

[8] J. H. Ferziger and M. Peric. *Computational methods for fluid dynamics*. Springer, 3rd edition, 2002.

[9] E. Fourcade, H. C. J. Hoefsloot, G. van Vliet, W. Bunge, S. M. P. Mutsers, and P. D. Iedema. The influence of micromixing on molecular weight distribution during controlled polypropylene degradation in a static mixer reactor. *Chemical Engineering Science*, 56(23):6589 – 6603, 2001.

[10] E. Fourcade, R. Wadley, H. C. J. Hoefsloot, A. Green, and P. D. Iedema. CFD calculation of laminar striation thinning in static mixer reactors. *Chemical Engineering Science*, 56(23):6729 – 6741, 2001.

[11] S. R. Idelsohn and E. Oñate. Finite volumes and finite elements: Two 'good friends'. *International Journal for Numerical Methods in Engineering*, 37(19):3323 – 3341, 1994.

[12] M. Jahoda. Lectures in cfd course. Technical report, Institute of chemical technology, Prague, CZ, 2005.

[13] OpenCFD Ltd. Official webpage (http://www.openfoam.com/), November 2012.

[14] OpenCFD Ltd. Openfoam - user guide, May 2012.

[15] E. S. Mickaily-Huber, F. Bertrand, P. Tanguy, T. Meyer, A. Renken, F. S. Rys, and M. Wehrli. Numerical simulations of mixing in an SMRX static mixer. *The Chemical Engineering Journal and the Biochemical Engineering Journal*, 63(2):117 – 126, 1996.

[16] J.A. Moore, D.A. Steinman, D.W. Holdsworth, and C.R. Ethier. Accuracy of computational hemodynamics in complex arterial geometries reconstructed from magnetic resonance imaging. *Annals of Biomedical Engineering*, 27:32 – 41, 1999.

[17] F. J. Muzzio, M. M. Alvarez, S. Cerbelli, M. Giona, and A. Adrover. The intermaterial area density generated by time- and spatially periodic 2D chaotic flows. *Chemical Engineering Science*, 55(8):1497 – 1508, 2000.

[18] D. Rauline, P. A. Tanguy, J. M. Le Blevec, and J. Bousquet. Numerical investigation of the performance of several static mixers. *The Canadian Journal of Chemical Engineering*, 76(3):527 – 535, 1998.

[19] L. F. Richardson. The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 210(459 – 470):307 – 357, 1911.

[20] M. K. Singh, P. D. Anderson, and H. E. H. Meijer. Understanding and optimizing the SMX static mixer. *Macromolecular Rapid Communications*, 30(4 – 5):362 – 376, 2009.

[21] M. K. Singh, T. G. Kang, H. E. H. Meijer, and P. D. Anderson. The mapping method as a toolbox to analyze, design, and optimize micromixers. *Microfluidics and Nanofluidics*, 5(3):313 – 325, 2008.

[22] A. D. Stroock, S. K. W. Dertinger, A. Ajdari, I. Mezic, H. A. Stone, and G. M. Whitesides. Chaotic mixer for microchannels. *Science*, 295(5555):647 – 651, 2002.

[23] Autodesk WikiHelp. Finite element vs finite volume, September 2012.

[24] D. Wolf and W. Resnick. Residence time distribution in real systems. *Industrial & Engineering Chemistry Fundamentals*, 2(4):287 – 293, 1963.

Appendix A

fvSolution

Settings for simpleFoam

```
solvers
{
  p
  {
    solver                  GAMG;
    tolerance               1e-7;
    relTol                  0.01;
    smoother                GaussSeidel;
    nPreSweeps              0;
    nPostSweeps             2;
    cacheAgglomeration      true;
    nCellsInCoarsestLevel   10;
    agglomerator            faceAreaPair;
    mergeLevels             1;
  }

  U
  {
    solver                  PBiCG;
    preconditioner          DILU;
    tolerance               1e-08;
    relTol                  0.001;
  }
}

SIMPLE
{
  nNonOrthogonalCorrectors 1;
```

```
  residualControl
  {
    p                       1e-7;
    U                       1e-7;
  }
}

relaxationFactors
{
  p                         0.3;
  U                         0.7;
}
```

Settings for scalarTransportFoam

```
solvers
{
  T
  {
    solver                  PBiCG;
    preconditioner          DILU;
    tolerance               1e-06;
    relTol                  0;
  }
}

SIMPLE
{
  nNonOrthogonalCorrectors 1;
}
```

Appendix B

fvScheme

Settings for simpleFoam

```
ddtSchemes
{
  default                steadyState;
}

gradSchemes
{
  default                Gauss linear;
  grad(p)                Gauss linear;
  grad(U)                Gauss linear;
}

divSchemes
{
  default                none;
  div(phi,U)             Gauss linear;
}

laplacianSchemes
{
  default                none;
  laplacian(nuEff,U)     Gauss linear corrected;
  laplacian((1|A(U)),p)  Gauss linear corrected;
}

interpolationSchemes
{
  default                linear;
  interpolate(U)         linear;
```

```
}

snGradSchemes
{
  default                 corrected;
}

fluxRequired
{
  default                 no;
  p                       ;
}
```

Settings for scalarTransportFoam

```
ddtSchemes
{
  default                 Euler;
}

gradSchemes
{
  default                 Gauss linear;
}

divSchemes
{
  default                 none;
  div(phi,T)              Gauss limitedLinear 1;
}

laplacianSchemes
{
  default                 none;
  laplacian(DT,T)         Gauss linear corrected;
}

interpolationSchemes
{
  default                 linear;
}

snGradSchemes
{
  default                 corrected;
```

```
}

fluxRequired
{
  default                 no;
  T                       ;
}
```