



**TECHNISCHE
UNIVERSITÄT
WIEN**

Diplomarbeit

Simulation of Membrane Modules with OpenFOAM

ausgeführt zum Zwecke der Erlangung des akademischen Grades

Diplom-Ingenieur

unter der Leitung von

Ass. Prof. Dipl.-Ing. Dr.techn Michael Harasek

unter der Betreuung von

Dipl.-Ing. Christian Jordan

E166

Institut für Verfahrenstechnik, Umwelttechnik und Technische Biowissenschaften

eingereicht an der Technischen Universität Wien

Fakultät für Maschinenwesen und Betriebswissenschaften

von

Philipp Schretter

Matrikel Nummer 0717910

Wien, im Jänner 2016

(Philipp Schretter)

Abstract

With the help of computational fluid dynamics (CFD), the design of membrane modules, the operating conditions and the implementation into the process like the design of cascade connections can be analyzed beforehand. A couple of studies investigate membrane separation processes in hollow fiber membranes with CFD whereas most of them use commercially available software.

In this thesis, CFD simulations of the flow through hollow fiber membranes were used in order to examine the separation performance.

The open source software OpenFOAM[®] was used. As the standard package of OpenFOAM[®] does not support the modeling of mass and energy transfer between different regions, a new base solver, named *membraneFoam* is developed at TU Wien. The solver handles source and sink terms on the boundaries to calculate energy and mass transfer between the regions. For all simulations, dense polymer membranes were assumed with mass transfer based on the solution-diffusion model.

In the first step, a 3-D model of a module with seven fibers, five different permeate outlet positions and a membrane area of 0.011 m^2 was modeled. To mesh the geometry, the software Ansys GAMBIT[®] was used. With this model, the influence of the flow patterns inside the hollow fibers and the shell on the separation performance was analyzed.

In order to validate the new solver code, the influence of different inlet mass flows on the separation performance was analyzed. The results from the CFD simulations were compared to simulations which were carried out with an already validated membrane separation model. The program has been developed at TU Wien with the commercial process simulation software Aspen Custom Modeler[®]. The results of the validation data to show deviations which proved to be in an acceptable low error range.

In a next step, a single fiber 3-D membrane module with a membrane area of 0.00035 m^2 was created with the software Ansys GAMBIT[®]. The influence of different flow patterns, transmembrane pressures, permeances and inlet mass flows on the separation performance of the membrane module was analyzed.

In order to compare the newly developed base solver to laboratory experiments, an actually existing hollow fiber module with 30 fibers and a membrane area of 0.0013 m^2 was modeled. The membrane module geometry was created with a commercially available CAD software and meshed with the *snappyHexMesh* utility, which is supplied with OpenFOAM[®]. The results of the CFD simulation were compared to data from measurements with the module from lab experiments. The actual module has a membrane wall thickness of $55\text{ }\mu\text{m}$, whereas the membrane walls in the simulation were assumed to be infinitely thin.

Based on the results of the simulations, it can be stated that the developed solver *membraneFoam* enables to make reliable statements over the separation efficiency and flow patterns in membrane modules.

Implementing a porous layer to the solver to consider membrane wall thickness, analysis of mixing promoters of the flow and the analysis of other fluids than gases can be subject to further investigation.

Kurzfassung

Mit Hilfe von numerischer Strömungssimulation (CFD), können das Design von Membranmodulen, die Betriebsbedingungen und die Umsetzung in den Prozess wie zum Beispiel die Schaltung von Kaskaden vorab analysiert werden.

Einige Studien untersuchen Membrantrennprozesse mit Hohlfasermembranen mit CFD, wobei zumeist kommerzielle Software verwendet wurde.

In dieser Arbeit wurden CFD-Simulationen der Strömung durch Hohlfasermembranen verwendet um die Trennleistung zu untersuchen.

Dafür wurde die Open-Source-Software OpenFOAM[®] verwendet. Da das Standardpaket von OpenFOAM[®] keine Stoff- und Energieübertragung zwischen verschiedenen Regionen unterstützt, wird ein neuer Basislöser mit dem Namen *membraneFoam* an der TU Wien entwickelt. Der Löser verwendet Quell- und Senkenterme an den Grenzschichten um Energie- und Massentransfer zwischen den Regionen zu berechnen. Für alle Simulationen wurden dichte Polymermembranen modelliert mit Massentransfer basierend auf dem Lösungs-Diffusions-Modell.

In einem ersten Schritt wurde ein 3-D Modell eines Membranmoduls mit sieben Fasern, fünf verschiedenen Permeatauslass-Positionen und einer Membranfläche von 0.011 m^2 modelliert. Um die Geometrie zu vernetzen, wurde die Software Ansys GAMBIT[®] verwendet. Mit diesem Modell wurde der Einfluss der Durchflussrichtung in den Hohlfasern und im Gehäuse auf die Trennleistung untersucht.

Um den neuen Löser Code zu validieren, wurde der Einfluss von verschiedenen Einlassmassenströmen auf die Trennleistung untersucht. Die Ergebnisse der CFD Simulation wurden mit Simulationen verglichen, welche mit einem zuverlässigen Membrantrennmodell durchgeführt wurden. Das Programm wurde an der TU Wien mit der kommerziellen Prozess Simulationssoftware Aspen Custom Modeler[®] entwickelt. Die Ergebnisse der Validierungsdaten zeigen Abweichungen, die in einem akzeptabel niedrigen Fehlerbereich liegen.

Im nächsten Schritt wurde ein Einzelfaser 3-D Membranmodul mit einer Membranoberfläche von 0.00035 m^2 mit der Software Ansys GAMBIT[®] erstellt. Der Einfluss von verschiedenen Durchflussrichtungen, transmembranen Drücken, Permeanzen und Einlassmassenströmen auf die Trennleistung des Membranmoduls wurde analysiert.

Um den neu entwickelten Basislöser mit Laborexperimenten zu vergleichen, wurde ein real vorhandenes Hohlfaser Membranmodul mit 30 Fasern und einer Membranoberfläche von 0.0013 m^2 modelliert. Das Membranmodul wurde mit einer kommerziellen CAD-Software gezeichnet und das Netz wurde mit *snappyHexMesh* erstellt, einem Werkzeug welches Bestandteil von OpenFOAM[®] ist. Die Ergebnisse der CFD Simulation wurden mit Daten der Labormessungen mit dem Modul verglichen. Das tatsächlich vorhandene Modul hat eine Membranwanddicke von $55 \mu\text{m}$, wobei die Membranwand in der Simulation als unendlich dünn modelliert wurden.

Basierend auf den Ergebnissen der Simulationen lässt sich die Aussage treffen, dass der entwickelte Löser *membraneFoam* verlässliche Aussagen über die Trennleistung und Strömungsprofile in Membranmodulen.

Die Implementierung einer porösen Schicht in den Löser, um die Membranwand zu berücksichtigen, die Analyse von Mischpromotoren in der Strömung und die Analyse von anderen Fluiden außer Gasen können Gegenstand weiterer Untersuchungen sein.

Contents

Abstract	i
Kurzfassung	ii
1 Introduction	1
1.1 Motivation and Problem Statement	1
1.2 Previous studies of CFD for membranes	2
1.3 Aim of the work	4
2 Basics of membrane separation	6
2.1 Membrane types	6
2.2 Membrane configuration and module design	8
2.3 Terms and definitions of membranes	9
2.4 Module operation	12
2.5 Resistances to transmembrane flow	13
2.6 Concentration polarization	14
2.7 Flow regime in membrane fibers	15
2.8 Gas permeation and separation principles	16
2.8.1 Viscous flow	16
2.8.2 Knudsen diffusion	17
2.8.3 Surface diffusion	18
2.8.4 Diffusion in dense membranes	19
3 Basics of numerical simulation	24
3.1 Procedure in numerical simulation	24
3.2 Structure of CFD packages	25
3.3 Solution method for Navier-Stokes equations	26
3.3.1 SIMPLE algorithm	26
3.3.2 PISO algorithm	28
3.4 Finite volume method	28
3.5 Equations in fluid mechanics	29
3.5.1 Continuity equation	29
3.5.2 Momentum equation	30
3.5.3 Energy equation	31
3.6 OpenFOAM	32
3.6.1 membraneFoam solver	33
3.6.2 LTSmembraneFoam solver	34
3.7 Hardware for simulations	35
4 Test Cases	36
4.1 Hollow fiber membrane with seven fibers	36
4.1.1 Dimensions of the seven fiber module	36
4.1.2 Mesh creation	36
4.1.3 Simulation settings in OpenFOAM	37
4.1.4 Flow regime	38
4.2 LTSMembraneFoam validation test cases	38
4.2.1 Geometry and mesh for CFD cases	38
4.2.2 Simulation settings CFD and ACM	38
4.3 Hollow fiber membrane with one fiber	40
4.3.1 Dimensions of the test case	41
4.3.2 Mesh creation	41
4.3.3 Simulation settings in OpenFOAM	42
4.4 Hollow fiber membrane with 30 fibers	45
4.4.1 Membrane properties	45
4.4.2 Dimensions of the test module	45

4.4.3	Meshing with snappyHexMesh	47
4.4.4	Simulation settings in OpenFOAM	49
5	Laboratory measurements	50
5.1	Feed and volume flow	50
5.2	Pressure	50
6	Validation	51
6.1	Aspen Custom Modeler (ACM) model	51
6.2	Validation of the ACM model	53
6.2.1	Validation ACM model against an algorithm	53
6.2.2	Validation ACM model against lab experiments	54
6.3	Validation CFD solver LTSMembraneFoam against ACM model	55
6.3.1	Deviation between ACM and OpenFOAM	60
7	Convergence	64
7.1	Numerical errors	64
7.2	Physical errors	65
7.3	Convergence of CFD simulations with different feed velocities	67
7.4	Variation of Courant numbers	67
8	Results and discussion	70
8.1	Hollow fiber membrane with seven fibers	70
8.1.1	Flow regime	70
8.1.2	Flow pattern analysis	71
8.2	Hollow fiber membrane with one fiber	77
8.2.1	Variation of velocities with co-current flow	77
8.2.2	Variation of velocities with counter-current flow	77
8.2.3	Variation of transmembrane pressures with co-current flow	79
8.2.4	Variation of transmembrane pressures with counter-current flow	79
8.2.5	Variation of velocities with higher permeances	80
8.2.6	Variation of the permeance of hydrogen	81
8.2.7	Variation of permeances of all species	83
8.3	Hollow fiber membrane module with 30 fibers	85
8.3.1	Laboratry experiments	85
8.3.2	CFD simulations	86
8.4	Deviation between simulation and experiments	87
9	Conclusion and Outlook	89
	References	viii
	Appendix	ix
	List of Symbols	xi
	List of Figures	xii
	List of Tables	xii
	Acknowledgements	xiii
	Datasheet	xiv
	Flowsheet	xv
	Plots from CFD simulation	xvi
	Case setup	xix

1 Introduction

1.1 Motivation and Problem Statement

A critical issue that today's society must address is the shift from fossil fuels to regenerative energies. Fossil fuels are not only locally available but also a limited resource of energy.

Fuels that can be produced from biomass is generally referred to as biofuel. On the basis of application and feedstock utilization, biofuels can be classified into first generation biofuels like ethanol, fatty acid methyl ester or biodiesel and pure plant oil and second generation biofuels where the feedstock has lignocellulose material that can come from bi-products of agriculture or from residues from the forest. Important factors which have to be accounted for are the availability of raw material and the sustainability of resources, adverse effects over the biodiversity and the competition for farmlands. [Babu et al., 2014] Therefore, sustainable energy carriers must be identified and analyzed according to environmental emissions, energy security and costs. One pathway is the use of hydrogen as an energy carrier, extracted from renewable resources such as water or biomass. [Turner, 2004]

Another pathway is the upgrading of biogas to the quality of natural gas in order to substitute natural gas with the advantage of reduced emissions of carbon dioxide. Thus the upgraded biogas can be injected in the natural gas grid or used as a vehicle fuel. Biogas can be produced from sewage treatment plants, in landfills or from anaerobic digesters. In biogases, besides carbon dioxide and methane, several other components like hydrogen sulphide or ammonia can be found. There are a series of technologies which are used to upgrade biogas like pressurized water scrubbing, pressure swing adsorption, amine scrubbing, scrubbing with organic solvents or membrane permeation. The usage of membrane gas permeation systems has several advantages. The process is safe, easy to operate, the capital costs are comparatively low for small plants and no hazardous chemicals are used. [Makaruk et al., 2010]

Membranes for carbon dioxide removal are especially applicable for small plants where operating pressures are in the range of 5 - 7 bar and at ambient temperatures. Highly carbon dioxide selective membrane materials are used for this purpose like polysulfone, polyimide or polydimethylsiloxane. Other fields of application include the removal of carbon dioxide from natural gas for moderate volume gas streams and the removal of carbon dioxide from flue gas. [Pabby et al., 2009]

In the membranes which are usually used for gas permeation, the carbon dioxide permeates faster than methane and thus the product remains on the high pressure retentate side. The desired purity of the product gas can be achieved either by increasing the membrane surface or by recycling the product stream with a cascade connection. This leads to a trade off between yield and costs for compressing the feed stream. [Rautenbach and Melin, 2004]

The integration of membrane separation modules in biogas upgrading plants is therefore an important task. With the help of simulation software like one-dimensional membrane solvers or computational fluid dynamics (CFD), the design of the modules and the implementation into the process can be analyzed beforehand. Statements can be made in advance about the influence of different operating conditions like pressure, temperature and different process designs, like flow patterns or the number of recycle flows, on the separation efficiency.

The usage of CFD has several considerable advantages over simple one-dimensional pro-

cess simulation models:

- In CFD, also 3 dimensional flow can be modeled and analyzed. Therefore influences of laminar and turbulent regimes and of the flow pattern, co- counter- or mixed-current, on mass transport can be investigated. Parameters like velocities, mass fractions or temperatures can be analyzed for certain zones such as boundary layers.
- With a decent meshing tool, more complex module geometries can be meshed. This allows analysis and the detection of potential turbulence or dead wake areas.
- The concentration gradient parallel to the membrane surface is considered. This gives a more accurate representation of the concentration distribution. It is also possible to examine the effect of turbulent mixing induced by spacers on concentration polarization.
- The effect of concentration polarization near the membrane surface is considered and can be visualized.
- The pressure drop in the feed side region as well as on the permeate side region is calculated.

For the modelling of flow and concentration polarization in membrane systems, CFD is an important tool to investigate membrane fouling and subsequent permeate flux decline. Therefore, CFD evolved as being an effective tool for designing better membranes more rapidly and cost effectively. [Ghidossi et al., 2006]

1.2 Previous studies of CFD for membranes

To the present day, several studies can be found which deal with CFD to investigate mass transfer in membranes.

Some of the papers which were published about this topic deal with the commercial software COMSOL Multiphysics[®] [Santafe-Moros and Gozalvez-Zafrilla, 2013], [Tahvildari et al., 2015] or with CFX4 [Wiley and Fletcher, 2002]. Marriott and Sørensen present a mathematical model to describe membrane separation in spiral-wound modules and radial-flow hollow-fiber modules with the software *gProms* [Marriott and Sorensen, 2003]. The thesis from [Nanduri, 2011] from the university of Toledo deals with the simulation of crimped hollow fiber membranes with ANSYS FLUENT[®]. To account for mass transfer flux across surfaces, a user defined function was created whereas the mass transfer was modeled in analogy to the heat transfer. The work studies a single straight and a single crimped fiber whereas only the flow on the feed side was modeled. An overview of earlier work about modeling of membrane separation processes, e.g. the linking of Navier-Stokes equations to the Darcy equations for the description of mass transfer through porous walls with MATLAB or the formation of concentration polarization from water with finite element analysis, is given in [Ghidossi et al., 2006]. The authors also give an overview of earlier studies, where simulation results could show the problems of laminar flow of water and growing concentration boundary layers on mass transfer and on fouling on the permeable wall. Further, simulation results of the implications of air injection, spacers, vortices and the geometry on the hydrodynamic conditions are summarized.

The work of [Koch, 2015] deals with the implementation of a genetic algorithm into OpenFOAM, version 2.3.0, to investigate the placement of circular spacers, known as baffles,

in forward osmosis asymmetric membrane channels. The solver models the effects of concentration polarization on the rejection and on the draw sides of the membrane on the osmotic pressure loss. The influence of baffles as spacers on the water flux and on the pressure loss was investigated. For this work, a solver was developed which is based on the weakly-compressible equations where the density is not coupled with the pressure but with the viscosity through the solute mass fraction. Therefore an additional equation, the diffusion-convection equation, is solved to account for the mass transport between the control volumes. The developed solver *SaltPimpleFoam* is based on the *twoLiquidMixingFoam* solver which comes with OpenFoam and uses the PIMPLE algorithm. The solver is capable of modeling the effects of cross-flow in asymmetric membranes and the genetic algorithm was proved to produce reliable results. For validation, two 2-D geometries were created with one membrane and two membranes respectively with circular baffles as spacers.

[Gruber et al., 2012] validate a CFD model, which is developed to simulate membrane separation via forward osmosis. The model is based on previous work from [Gruber et al., 2011]. In forward osmosis, the driving force is based on an osmotic pressure gradient rather than the hydraulic pressure. The membrane has an asymmetric design with a dense active layer on the feed side and a porous layer for mechanical support on the permeate side. The simulations were carried out with the OpenFOAM library, version 1.7. The solver uses the PISO algorithm whereas the continuity equation, the Navier-Stokes equation and the convection-diffusion equation are solved in the PISO loop. The density is expressed as a function of solute mass fraction and considered independent of the pressure, which is denoted as weakly compressible. The simulations were run until steady state conditions could be reached, whereas no turbulence models were used. As test cases, two 3-dimensional geometries were used whereas each grid consisted of approximately 500 000 cells.

The simulation results show a good prediction of water fluxes at different cross-flow velocities. The influences of feed and draw concentrations on the effective osmotic pressure were analyzed. These results were also compared to analytical calculations of the mass transfer. It could be shown that the analytical model underestimates the significant of concentration polarization which is due to the higher spatial resolution of the CFD model and the simplifications of the governing equations in the analytical model. It is expected that for final applications, eddy-promoting spacers could be used to reduce the negative effects of concentration polarization on the mass transfer.

The proceedings of the 12th international conference on membrane science and technology showed, among others, a 3-dimensional CFD simulation of hydrogen separation in palladium hollow fiber membranes whereas the software used is not specified.

[Sharafpoor et al., 2015]

Also a 2-dimensional mathematical model was presented for hydrogen separation in Pd hollow fiber membranes. CFD was used to solve the model equations.

[Mansourpour et al., 2015]

Further, in two publications, palladium membrane reactor performance during ethanol steam reforming was investigated with the Comsol-Multiphysics software version 5. [Andalib et al., 2015], [Zeynali et al., 2015]

One work deals with the CFD simulation of the separation of carbon dioxide and methane with a Silico-Alumino Phosphate-34 (SAPO-34) tubular membrane with the software ANSYS FLUENT. [Banitaba et al., 2015]

Another model which is mentioned in the proceedings simulated gas permeation with mixed matrix membranes based on the solubility-diffusion mechanism. The software which was used for the CFD simulations is not mentioned. [Izadiar et al., 2015]

In previous studies at the institute of chemical engineering at TU Wien CFD was used to investigate flow behavior in membranes.

In [Staudacher, 2000] a dialysis module for the cleaning of blood was simulated with FLUENT. A main part of the work was the implementation of a user defined function, written in the programming language C, to model the mass flow through the dialysis membrane. Therefore, the pressure and the concentration on the membrane boundaries were used as input parameters and the mass transfer was considered with source and sink terms. The dialysis membrane module had around 7000 capillary fibers but in order to reduce the complexity, only one fiber was modeled. As a further simplification, the mass transfer inside the membrane was neglected. The pressure loss and the point of back filtration were analyzed and compared to experimental data.

In a paper from [Staudacher et al., 2002], the flow of gases and vapors in a membrane test module was simulated with FLUENT and validated with Laser Doppler Anemometry. Analysis of the simulation data showed dead zone and a stagnation point in the module. With the experiments only certain regions of the membrane module could be measured. Though the measurement and the simulation results were in good accordance. It could be shown that CFD can provide useful information regarding the construction and design of membrane modules.

A work that was conducted to find the optimal configuration of a hollow-fiber membrane module with the application of CFD was published by [Costa et al., 2005]. The membrane module was used for humidification and dehumidification of air and the fibers were placed perpendicular to the flow direction. An absorbent circulated through the lumen of the fibers. In order to simulate the mass transfer, the concentration profile was investigated with the help of the heat transfer equation. Thus, the analogy between heat and mass transfer was used. With the help of CFD, the best arrangement of the fibers in the module with the highest mass transfer and minimal energy consumption could be determined.

In general, several studies are available investigating mass transfer of gaseous mixtures and concentration polarization of membranes with CFD. Up to the present day, none of the models can be found in public literature which uses the open source software OpenFOAM for modeling gas separation.

1.3 Aim of the work

The aim of the work done for this thesis is to perform CFD simulations of the flow through hollow fiber membranes and to examine the separation properties. Therefore, the open source software OpenFOAM[®] is used. As the standard package of OpenFOAM[®] doesn't support the modeling of mass and energy transfer between different regions, a new transient base solver, named *membraneFoam* is developed at TU Wien. The solver handles source and sink terms on the boundaries and energy and mass transfer between the regions. Further, also a steady state solver using the local time stepping (LTS) method is developed and named *LTSMembraneFoam*.

In this work the simulations were carried out for membrane modules with nonporous, dense membranes for gas separation.

The goal was to create and mesh different membrane module geometries in order to investigate the reliability of the *membraneFoam* and the *LTSMembraneFoam* solver.

The influence of flow patterns like co-current or counter-current on the separation efficiency was analyzed. Also, the influence of different transmembrane pressures and permeances on the separation performance was analyzed.

Further, test cases were set up with different inlet mass flows in order to investigate the separation properties. The results were used to validate and compare the *membraneFoam* and the *LTSMembraneFoam* solver to literature data and to data from a process simulation software.

In order to validate the solver code to laboratory experiments, an already available hollow fiber module with 30 fibers and with a reverse-selective PDMS membrane was modeled. The results were compared to data from lab experiments.

In the end, with the simulations which are run, statements should be made, if the new solver can reliably predict and model membrane separation processes.

2 Basics of membrane separation

A membrane is a separating layer which is permeable for at least some of the liquids or gases which are in contact with it. For some other components, the membrane may be impermeable. The separation of molecular and particulate mixtures are the main applications of membranes. Other applications involve chemical or electrochemical synthesis, energy storage and conversion and the controlled release of active agents.

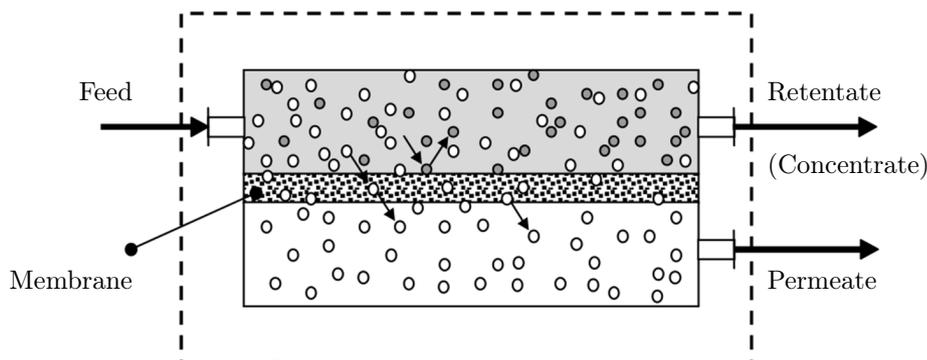


Figure 1: Membrane separation principle, adapted from [Rautenbach and Melin, 2004]

2.1 Membrane types

Membranes are categorized either by the size of the permeating components, by the characteristics of separation or by the condition of aggregation on both sides of the membrane. [Rautenbach and Melin, 2004]

Membranes which are used for separation can also be classified according to the driving forces of the processes. [Strathmann, 2011]

- Hydrostatic pressure: reverse osmosis, nano-, ultra, microfiltration, gas separation
- Concentration gradient or chemical potential gradient: dialysis, pervaporation, membrane contactors, scrubbers
- Electrical potential gradient: electrolysis, electrodialysis

According to [Mulder, 1996], membranes can be classified into three basic types in regard to the structure, morphology and the separation principle.

- Porous membranes
Separation in porous membranes is based on the particle size. The selectivity is enhanced if the particle size is large compared to the membrane pore size. In comparison to dense membranes, the separation properties are more influenced by the pore size than by the membrane material.
- Nonporous membranes
Separation in dense membranes is based on differences in solubility and the diffusivity. Thus the intrinsic properties of the polymeric material influence the selectivity and permeability.

- Carrier membranes

The separation is determined by carrier molecules. These molecules are either fixed to the membrane matrix or dissolved in the fluid in the pores of a porous membrane. The properties of the carrier molecules have the main influence on the permselectivity towards a component.

As described in table 1, porous membranes are used for microfiltration and ultrafiltration. The membranes can be characterized by the pore size into macroporous > 50 nm, mesoporous 2 nm $<$ pore size < 50 nm and microporous < 2 nm.

Nonporous or dense membranes are used for gas separation, pervaporation, vapour permeation and dialysis. For dense pervaporation or gas separation membranes, the choice of the polymer material is important and can be classified into crystalline, amorphous, glassy and rubbery. Glassy and rubbery polymers are distinguished by the glass transition temperature, which is affected by structural factors like chain flexibility, chain interaction and molecular weight. This temperature indicates the transition of the non-crystalline, amorphous polymer from a glassy to a rubbery state. The modulus of elasticity, which indicates the force, necessary to obtain deformation is three to four orders of magnitude lower in rubbery state than in glassy state.

Apart from polymers, membranes can also be composed of inorganic materials which generally possess a higher thermal, chemical and mechanical stability. The inorganic materials refer to the porous membranes and may be distinguished into ceramic membranes, glass membranes, metallic membranes and zeolitic membranes. [Mulder, 1996].

Two main types of membranes, porous and dense membranes, are depicted in figure 2.

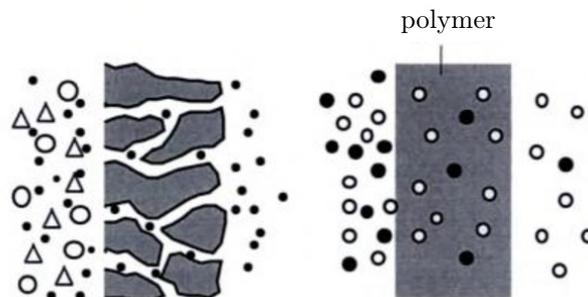


Figure 2: Membrane classification, [Mulder, 1996]

Not all membranes can be classified into one of the above types. Reverse osmosis, for instance, is found between porous and dense membranes. Another possibility to classify membranes is according to the manufacturing process into asymmetric and symmetric membranes as depicted in figure 3. For symmetric membranes, the separation properties are influenced by the layer which accumulates during the separation.



Figure 3: Symmetric (left) and asymmetric membrane (right)[Rautenbach and Melin, 2004]

Asymmetric membranes have a thin active layer on top of the membrane polymer which has a significant influence on the separation properties. This membrane type allows

a relatively high flux and stability with a thin structure. The active layer generally represents only a small part of the membrane thickness. According to [Mulder, 1996] the active layer is in the range of 0.1 to 0.5 μm whereas the porous sublayer is in the range of 50 to 150 μm . The thin active layer has the main influence in the selectivity and on the transmembrane flux.

2.2 Membrane configuration and module design

In order to use membranes for separation processes, they have to be implemented in a module configuration. With a membrane module, a feed stream is divided into retentate and a permeate stream. The driving force can be the difference of the pressure or the chemical potential across the membrane, depending on the application of the membrane module. Generally, the membrane modules have one inlet- and two outlet streams (3-end-module). In case the retentate outlet is closed, the module is in dead-end-operation (2-end module). Occasionally, on the permeate side, the module is operated with an additional sweep stream (4-end-module). The membrane types can be classified into two types of configuration, flat and tubular.

- Modules with flat membranes

Modules with flat membranes can be classified into plate modules, frame modules and spiral-wound modules.

In flat membranes, the separation barrier is on the surface of the membrane. The modules can be small like in stirred batch cells or in dead-end filter devices for laboratory use. Flat membranes are also versatilely applicable in different scales in plate-and-frame modules for ultrafiltration, reverse osmosis or gas separation. In larger scale industrial applications, flat membranes are used in pleated filter membrane cartridges and in spiral-wound modules. The modules are therefore very different in their design, manufacturing costs and lifetime. [Rautenbach and Melin, 2004]

- Modules with tubular membranes

Tubular membranes have an inner diameter between 5 and 25 mm and are generally several meters long. The feed flows only on the inside of the tubes. In relation to the volume, these membranes have a low surface $< 200 \text{ m}^2/\text{m}^3$ and the manufacturing costs are high. The advantages of this module type is that plugging, fouling and the concentration polarization can be controlled. The module can be used for highly viscous fluids and for solutions with high solids loading.

Capillary membranes have an inner diameter between 0.2 and 3 mm. In general, several capillaries are packed in a shell with the selective layer on the inside of the capillary. The feed flows either in the lumen or in the shell of the membrane. The operating pressure is relatively low in the range of 2 - 10 bar. Therefore, the application is limited to processes with low membrane pressures such as dialysis, microfiltration and ultrafiltration.

Hollow fiber membranes have an outer diameter between 0.05 and 0.5 mm and the wall thickness is between 0.01 and 0.1 mm. Generally, the selective layer is on the

outside of the hollow fiber and the membrane structure is asymmetric. Up to several thousand fibers can be bundled, usually curled in a half loop. Both ends of the fibers are potted with an epoxy resin.

Basically, the feed stream can pass the membrane either from inside of the fiber or from the outside. In case the feed flow is on the inside of the fibers, the active layer can be either on the inside or on the outside of the hollow fiber membrane. The advantage of an active layer on the outside is the increased surface area and consequently an increased flux. For feed flow inside the fiber, a pressure of up to 15 bar is allowed and the pressure drop on the permeate side is negligibly low. In contrast to inside flow, for outside flow, a pressure difference from feed to permeate of up to 70 bar is allowed, whereas the pressure drop on the permeate side is considerable.

The advantages of hollow fiber modules are the high surface area in relation to the volume with up to $30\,000\text{ m}^2/\text{m}^3$, its low manufacturing costs and its high packing density. The main disadvantages are that concentration polarization and fouling can occur. This is especially the case, if the feed flow is on the inside of the fibers and if the feed flow regime is laminar. The feed, especially liquids, needs to be very clean and possibly pretreated. Its main applications are in reverse osmosis, pervaporation and in gas permeation.[Rautenbach and Melin, 2004]

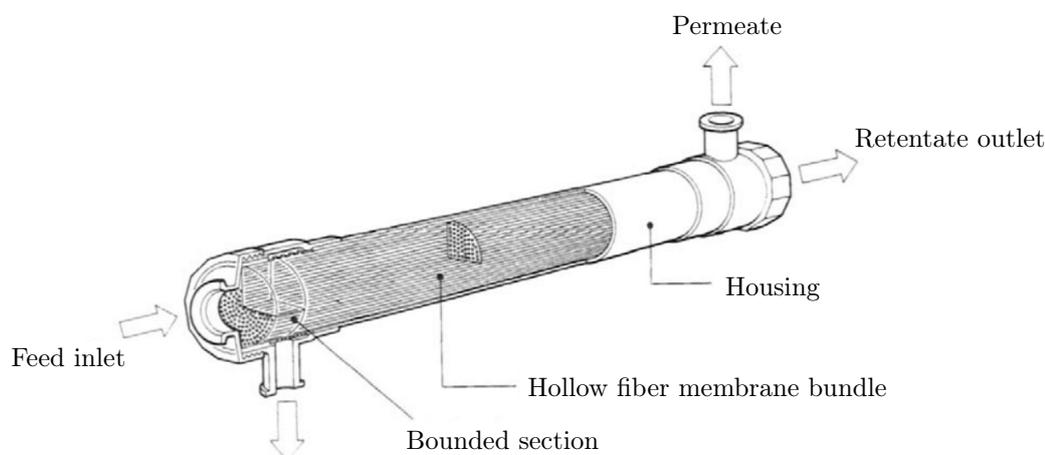


Figure 4: Hollow fiber membrane [Scott, 1996]

Figure 4 shows the structure of a hollow fiber membrane module. The feed enters the module through a port, where the stream is manifolded into the fibers. Similarly the retentate exits the tube side. The permeate exits the shell through one or more ports at the beginning or at the end of the module.

2.3 Terms and definitions of membranes

The capability of membranes to separate a specific component from a mixture can be expressed by the selectivity coefficient, the rejection and the permeability. The **selectivity** of a membrane describes the capability to differentiate between components in a mixture.

Depending on the selectivity, the product can be on the permeate side as well as on the retentate side. For a binary mixture, indicated by the indices i and j , the selectivity $S_{i,j}$ is defined by the composition of the product stream and the feed stream. For a binary mixture and with the mole fractions of feed x and permeate y the **selectivity**, which is generally also referred to as separation factor, is defined by

$$S_{ij} = \frac{\frac{y_i}{y_j}}{\frac{x_i}{x_j}} = \frac{\frac{y_i}{1-y_i}}{\frac{x_i}{1-x_i}}. \quad (2.1)$$

For a binary mixture, the selectivity is also defined by the mass fractions of the feed w_F and of the permeate w_P .

$$S_{ij} = \frac{\frac{w_{iP}}{w_{jP}}}{\frac{w_{iF}}{w_{jF}}} = \frac{\frac{w_{iP}}{1-w_{iP}}}{\frac{w_{iF}}{1-w_{iF}}}. \quad (2.2)$$

The permselectivity is given with the permeability P_i for component i and P_j for component j .

$$S_{ij} = \frac{P_i}{P_j} \quad (2.3)$$

Besides the selectivity, another coefficient is the **rejection**, that indicates the separating efficiency of a membrane.

$$R_i = \frac{w_{iF} - w_{jP}}{w_{jF}} = 1 - \frac{w_{iP}}{1 - w_{iF}} \quad (2.4)$$

In gas separation, the product of the diffusion coefficient D and the sorption coefficient k defines the **permeability coefficient** :

$$P_i = D_i k_i \quad (2.5)$$

The sorption coefficient or solubility coefficient for a membrane for component i is given by

$$k_i = \frac{\varphi_i \rho_i^m}{\bar{\gamma}_i p_i^0 M_i} \quad (2.6)$$

where φ is the fugacity coefficient, a correction factor for the actual pressure, ρ is the density, $\bar{\gamma}$ is the average activity coefficient, a correction factor for the actual concentration, p_i^0 is the saturation pressure and M is the molecular weight of the component i in the membrane.

The sorption coefficient determines the solubility of a component in a membrane. It generally increases with the size of the dissolving molecules. For an ideal mixture or gas, the coefficient is constant and equal to the Henry coefficient K_i^H of the dissolving substance. The sorption coefficient is directly proportional to the concentration in the ideal gas mixture which is in contact with the membrane. Henry's law is

$$C_i^m = k_i^h p_i \quad (2.7)$$

with C_i^m the concentration of the component in the membrane, k_i^h the Henry constant and p_i the partial pressure of the component in the phase above the membrane. The sorption increases linearly with the partial pressure and is inversely proportional to the saturation pressure. [Strathmann, 2011]

The diffusion coefficient decreases if the molecular weight of the diffusing component increases. The relation between diffusivity and molecular weight is described by the Stokes-Einstein equation for liquids and by the Einstein-Smoluchowski equation for solid materials.

The **permeance** is defined as the permeability related to the thickness of the active layer of the membrane. Whereas the permeability defines the material properties of the membrane, the permeance depends on the dimensions as well as on the material of the membrane. It is inversely proportional to the thickness of the membrane

$$Q_i \sim \frac{D_i k_i}{d} \quad (2.8)$$

Thus, the permeance defines a specific membrane. Generally, the permeance is determined with experiments. Therefore, the permeate flow is related to the membrane area and to the transmembrane pressure. The permeance of a certain specie has the unit

$$Q_i = \left[\frac{\text{m}^3 (\text{STP})}{\text{Pa s m}^2} \right] \quad (2.9)$$

The **permeability** is occasionally given in *barrer* which has the unit

$$1 \text{ barrer} = 10^{-10} \left[\frac{\text{cm}^3 (\text{STP}) \text{ cm}}{\text{cmHg s cm}^2} \right] \quad (2.10)$$

according to [Robb, 1968].

The conversion factor of the unit *barrer* to SI units is defined in good approximation by

$$1 \text{ barrer} \sim 7.5006 \cdot 10^{-18} \left[\frac{\text{m}^3 (\text{STP}) \text{ m}}{\text{Pa s m}^2} \right] \quad (2.11)$$

In order to describe the dependency of the permeability on the temperature, the Arrhenius equation is used. For lower temperatures, surface diffusion is predominant whereas for higher temperatures the gas diffusion becomes dominant. In polymer membranes the solubility and the diffusion coefficient depend on the temperature. Thus, also the permeability and the selectivity of components of a mixture depend on the temperature, which can be described by the Arrhenius-van't Hoff equations [Yampolskii and I. Pinnau, 2007]

$$\begin{aligned} D_j(T) &= D_{j,0} \exp\left(-\frac{E_D}{R T}\right) \\ S_j(T) &= S_{j,0} \exp\left(-\frac{\Delta H_s}{R T}\right) \\ P_j(T) &= P_{j,0} \exp\left(-\frac{E_P}{R T}\right) \end{aligned} \quad (2.12)$$

where T is the temperature, R is the gas constant, ΔH_s is the enthalpy of sorption which has two contributions, the enthalpy of condensation ΔH_c and the partial molar enthalpy ΔH_m for dissolving the penetrants in the permeate phase. E_D and E_P are the activation energies for a gas to diffuse and to permeate through a membrane. In the equations 2.12, the activation energies are defined as $E_P = E_D + \Delta H_s$. The term E_D is always positive, but E_P depends on the magnitude of ΔH_s .

For permanent gases, the enthalpy of sorption is small and the permeance increases with the temperature. For easily condensable gases, the enthalpy of sorption can be highly negative and therefore the permeability decreases with the temperature.

The **flux** across the membrane is defined as the amount of a certain unit per membrane area and per time as a result of a driving force. This can be written as

$$J = -L\nabla X \quad (2.13)$$

with J the flux and X the driving force. L is a coefficient that changes, depending on the flux property. L can stand for the permeability (P), the diffusion coefficient (D_i), the electrical conductivity (κ) or the heat conductivity (λ). X can stand for the pressure p , the concentration C , the electrical potential ϕ or the temperature T . The mass flux through a membrane with a dense matrix is based on diffusion, whereas through a membrane with pores or permanent channels, the flux is based on viscous flow. The following flux properties can occur across the membrane

$$\begin{aligned} J_v &= -P\nabla p && \text{Volume flux (Darcy's law)} \\ J_A &= -D_{AB}\nabla C_A && \text{Molar flux (Fick's law)} \\ J_e &= -\kappa\nabla\phi_i && \text{Ion flux (Ohm's law)} \\ J_e &= \lambda\nabla T && \text{Heat flux (Fourier's law)} \end{aligned} \quad (2.14)$$

In equation 2.14, Darcy's law states that the volume flux is related to the hydrostatic pressure difference. Fick's law relates the flux to the concentration gradient. Ohm's law relates the electrical current to the electrical potential gradient. Fourier's law relates the heat transport to the temperature gradient.

The **stage-cut** Θ gives the amount of permeate volume flow in relation to the feed volume flow.

High transmembrane fluxes are achieved with a high pressure difference, whereas for high purities of the preferred component, the pressure difference should be low. Usually, the target is to obtain a highly concentrated permeate stream whereas the residue stream has a low content of the permeable component. As both targets are difficult to reach at the same time, a trade-off is made which is expressed in the stage-cut.

$$\Theta = \frac{\dot{V}_p}{\dot{V}_f} \quad (2.15)$$

If the desired product concentration cannot be reached in one stage, several stages can be connected in series or in parallel to form a cascade of membrane modules.

Generally the feed stream and the product stream have different pressures. Therefore, in order to compare both volume streams for the stage cut, the streams are scaled to standard conditions for temperature and pressure (STP) of 273.15 K (0 °C) and an absolute pressure of 101 325 Pa (1 atm).

2.4 Module operation

Apart from the module design also the module operation like the flow conditions or the direction of the currents are significantly important for the gas separation process. For the gas separation modules, three flow patterns can be differentiated, cross-flow, co-flow

and counter-flow. The flow patterns are depicted in figure 5 along with the change in the concentration gradients across the membranes. The figure on the right side shows that the concentration gradient of the preferentially permeating component is higher for counter-flow than for co-flow. In order to see an improved separation process for counter-current flow, the effect of concentration polarization must be small on the permeate side. Therefore, fluids with a high diffusion coefficient are required, which restricts the advantages of counter-flow to pervaporation and gas separation. [Baker, 2004]

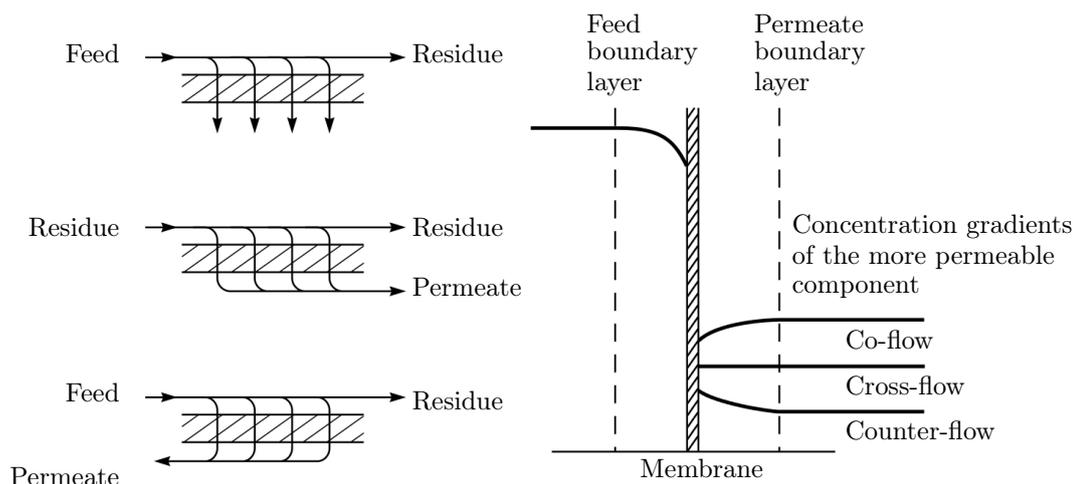


Figure 5: Flow patterns (left), cross flow (top), co-flow (middle), counter-flow (bottom), concentration gradient (right) [Baker, 2004]

On the permeate side, a lateral flow of sweep gas can be used to sweep the permeating components away from the membrane surface. This reduces the partial pressure of the permeating component and proportionally increases the partial pressure difference and consequently the transmembrane flux. The disadvantage of this method is that the sweep gas like helium, nitrogen or vapor has to be separated from the permeating components. Usually the permeating component is removed via condensation. In pervaporation, the feed state is liquid whereas on the permeate side, a partial vacuum is generated by cooling the permeating vapor stream, causing it to condense.

2.5 Resistances to transmembrane flow

According to [Rautenbach and Melin, 2004], besides the desired selective resistance of the active membrane layer, four local resistances to the transmembrane flow are relevant

- Concentration polarization on the feed side
- Concentration profile in the porous layer of asymmetric membranes
- Pressure drop in the porous layer
- Concentration polarization in the boundary layer on the permeate side

In general, the largest effects are attributed to the concentration polarization on the feed side and to the pressure drop in the porous layer.

2.6 Concentration polarization

The formation of concentration polarization decreases the separation process in two ways. One effect is the decline of the concentration towards the membrane surface of the component with the higher permeability. On the other side, the concentration of the components with lower permeability is increased. This has a direct influence on the driving force of the transmembrane flux as described in chapter 2.8.4. As a result, the total flow as well as the purity of the preferably permeating component decreases.

Two cases for mass transfer across the membrane can be differentiated. Firstly, the flux is controlled by the membrane itself which is the case for solution diffusion membranes. Secondly, the flux is controlled by the surface layer of particles that accumulates on the membrane. In the first case, the mass transfer coefficient can be introduced. Therefore, the following assumptions are made. [Rautenbach and Melin, 2004]

- Stationary flow
- Fick diffusion
- No chemical reactions
- The concentration gradient normal to the membrane is large compared to the parallel gradient

The course of the concentration of a specific component across the membrane wall is depicted in figure 6. The thickness δ_F is the area of concentration polarization. δ_M is the active layer of on the membrane, δ_S is the porous membrane, δ_P is the area of concentration polarization on the permeate side.

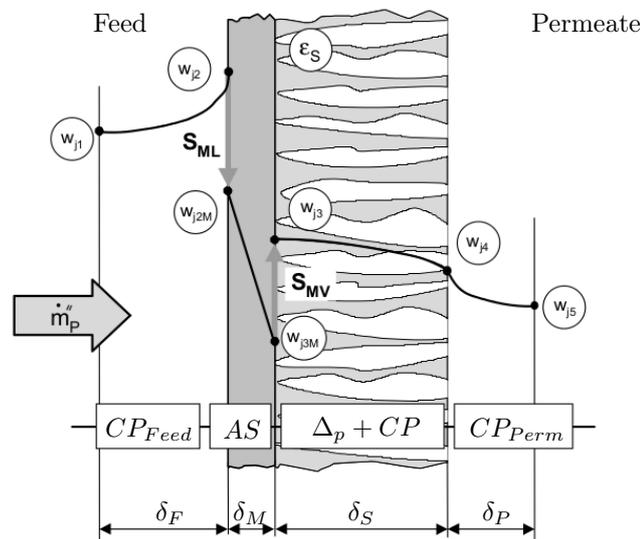


Figure 6: Course of the concentration [Rautenbach and Melin, 2004]

The components of the feed stream travel to the membrane surface via convection and diffusion. As a result of the selectivity of the membrane, the mass fraction of some components increases near the surface. The amount of some other components decreases near the membrane as they travel back to the core of the stream via diffusion. The effect

of concentration polarization on the feed side can be written as

$$\frac{w_{j1} - w_j^*}{w_{j2} - w_j^*} = \exp\left(-\frac{\dot{m}_P \delta_F}{\rho_F D_{ij}}\right) \quad (2.16)$$

which is sometimes also referred to as film-theory equation. From figure 6 as well as from equation 2.16 it can be seen that the concentration of a component j increases exponentially towards the membrane surface. The increase of the concentration is influenced by the mass transfer across the membrane as well as by the mass flow parallel to the membrane. Usually, a mass transfer coefficient $k_F \equiv \frac{D_{ij}}{\delta_F}$ is introduced. The mass transfer coefficient can be determined experimentally.

In order to quantify the magnitude of concentration polarization, the *Peclet-Number* can be used. It is defined as

$$Pe = \frac{\dot{m}_P d_{hyd}}{\rho_F D_{ij}} \quad (2.17)$$

whereas a low Pe-Number indicates a good diffusion in the boundary layer, a low increase of the concentration and a low polarization. The following equation shows the relation between the Peclet number, the Sherwood number and the concentration.

$$\frac{w_{j1} - w_j^*}{w_{j2} - w_j^*} = \exp\left(-\frac{Pe D_{ij}}{k_F d_{hyd}}\right) = \exp\left(-\frac{Pe}{Sh}\right) \quad (2.18)$$

The Sherwood number depends on the flow regime and on the geometry of the flow channel. Basically, the Sherwood number is defined as

$$Sh = \frac{k_F d_{hyd}}{D_{ij}} \quad (2.19)$$

which shows the relation of the of mass transfer and the diffusion.

For pervaporation and gas permeation the transmembrane fluxes are very low and according to [Rautenbach and Melin, 2004], the Peclet numbers are in the range of 10^{-3} to 1. The effect of concentration polarization increases with the transmembrane flux and with the selectivity. This effect can be reduced by adding turbulence to the feed-flow via flow patterns or channel spacers and usually also by increasing the feed flow rate. [Baker, 2004]

2.7 Flow regime in membrane fibers

A rough estimation of the flow regime can be done by the Reynolds number which is defined as

$$Re = \frac{v * L}{\nu} \quad (2.20)$$

where v is the velocity and d the characteristic length or diameter of the fiber in the current case.

The critical Reynolds number for transition from laminar to turbulent was found to be 2300. Nevertheless, the critical Re number highly depends on the conditions at the tube inlet and the inflow. Thus, the critical Re number increases for smaller disruptions at the inlet. This could be confirmed with experiments, where critical Re numbers of 40 000 could be found. [Schlichting and Gersten, 2005] The velocity profile can be derived from

the Navier-Stokes-equation and the resulting equation for the velocity profile is known as *Hagen-Poiseuille-flow*

$$v(r) = v_{max} \left(1 - \frac{r^2}{R^2} \right) \quad (2.21)$$

where v is the velocity, r is the distance from the center of the pipe to the wall and R is the radius of the pipe. By integrating the parabolic velocity profile, the volume flow through the pipe can be calculated.

$$\dot{V} = \int_A \mathbf{u} \cdot d\mathbf{A} = \int_0^{2\pi} \int_0^R v(r) r dr d\phi = \pi R^2 \frac{v_{max}}{2} \quad (2.22)$$

with the mean velocity $\bar{v} = v_{max}/2$.

The volume flow is proportional to the pressure drop along the pipe. The results of the pressure drop in the membrane fibers is described in chapter 8.1.2. Along the pipe with length L and with a constant pressure gradient it is given by

$$\Delta p = 8\mu \frac{\bar{v}L}{R^2} = \frac{\rho \bar{v}^2 L}{2} \frac{64\nu}{2R\bar{v}} = \underbrace{\frac{\rho \bar{v}^2}{2}}_{\text{scale}} \times \underbrace{\frac{L}{d}}_{\text{geom. form}} \times \underbrace{\frac{64}{Re}}_{\lambda} \quad (2.23)$$

where the pressure drop is a product of a specified scale, the geometry and a flow factor. [Kuhlmann, 2007]

The term λ is defined as the Darcy friction factor which is $64/Re$ for laminar flow.

2.8 Gas permeation and separation principles

The following figure 7 shows the mechanisms of gas permeation in porous and dense membranes. The flux across the membrane can be described by convective flow, Knudsen diffusion, surface diffusion and diffusion in dense membranes.

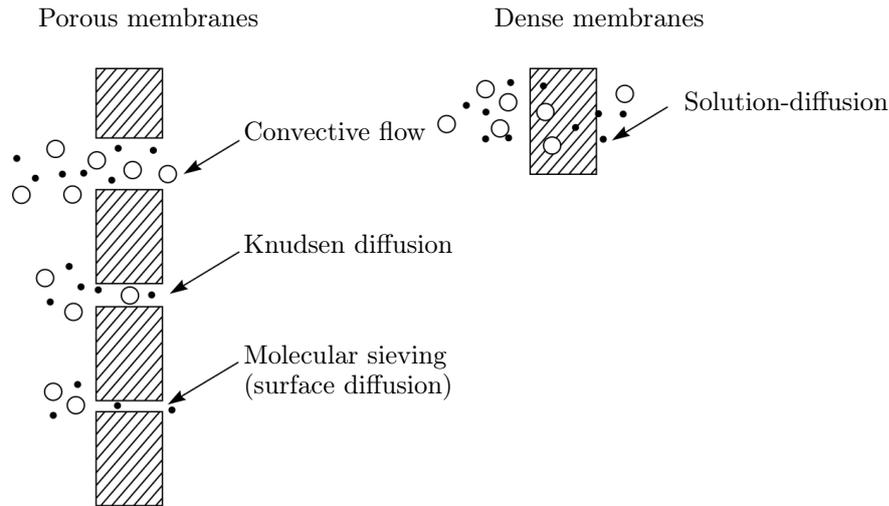


Figure 7: Transport of gases through membranes [Baker, 2004]

2.8.1 Viscous flow

In case of viscous flow or convective flow through a porous membrane, the driving force of the flux is a hydrostatic pressure gradient and the flux, J_v , can be described with the

Hagen-Poiseuille equation.

$$J_v = -\frac{r^2}{8\mu} \frac{\Delta p}{L} \quad (2.24)$$

with r , the radius, Δp the pressure difference between both ends of the capillary, μ the dynamic viscosity and L the length of the capillary. Convective flow occurs in membranes with a pore size from 0.1 to 10 μm . In case of viscous flow, the permeating components interact considerably more with each other than with the walls of the membrane pores. Another formulation for the Hagen-Poiseuille law is

$$J_v = \frac{\pi n r_p^4}{8\mu A_m \tau} \frac{\Delta p}{\Delta x}. \quad (2.25)$$

In equation 2.25 J_v is the flux, n the number of pores, r_p the pore radius, A_m the membrane area, μ the viscosity, τ the tortuosity factor, which relates the pore length to the thickness of the membrane. Δp is the pressure difference and Δx the membrane thickness.[Strathmann, 2011]

The dynamic viscosity μ of gases is primarily a function of the absolute temperature. This relationship is expressed by the Sutherland law.

2.8.2 Knudsen diffusion

Knudsen diffusion occurs in membrane pores which are smaller than 0.1 μm , which corresponds to the mean free path length of the molecules. In this case, the ratio of the pore radius to the gas mean free path (r/λ) is less than one and the molecules interact considerably more with the walls of the membrane pores than with each other. It can be seen from figure 8 that if the relation of pore radius to the mean free path length is greater than one, viscous flow predominates.

The ratio of the mean free path length, to the pore size, λ/r , is referred to as *Knudsen number*.

It can be shown that for Knudsen diffusion, the efficiency of the separation process depends on the square root of the ratio of the molecular weights of the gases that are separated.

$$\alpha_{i,j} \propto \sqrt{\frac{M_i}{M_j}} \quad (2.26)$$

The phenomenon depicted in equation 2.26 states, that when the difference of the molecular masses is high, also the separation factor is high. It was first observed by Graham and therefore is called Graham's law of diffusion.[Baker, 2004]

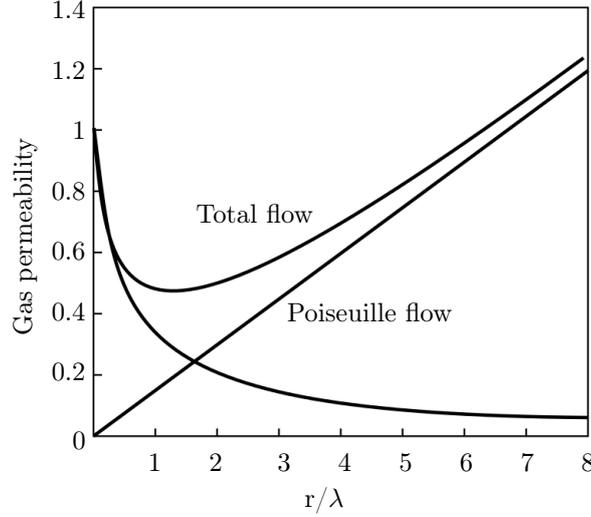


Figure 8: Relation Knudsen to Poiseuille flow [Baker, 2004]

The *mean free path length* is the average distance a gas molecule can travel before it collides with another gas molecule.

$$\lambda = \frac{kT}{\pi d_{gas}^2 p \sqrt{2}} \quad (2.27)$$

with k the Boltzmann constant, d_{gas} the diameter of the molecule and p the pressure. The *diffusion coefficient* for gases which pass pores that are smaller than their free path length is described as Knudsen diffusion coefficient. It is given by

$$D_i^k = 0.66r \sqrt{\frac{8RT}{\pi M_i}} \quad (2.28)$$

where M_i is the molecular weight of the gaseous component i .

The following equation describes the flux of a component i for Knudsen diffusion

$$J_i = -\frac{\pi n r^2 D_i^k \Delta p}{RT \tau \Delta z} \quad (2.29)$$

with n the number of pores in a unit area of the membrane, r the pore radius, Δp the pressure difference across the membrane, Δz the thickness of the membrane, τ the tortuosity factor and D^k the Knudsen diffusion coefficient. [Strathmann, 2011]

2.8.3 Surface diffusion

For gas fluxes across porous systems, which are governed by viscous flow and Knudsen diffusion, the pore diameters and the molecules have the same range of magnitude. Surface diffusion governs the membrane transport in small pores with a size from 0.5 to 2 nm.

Figure 9 shows the gas separation phenomena molecular sieving, capillary condensation and surface diffusion.

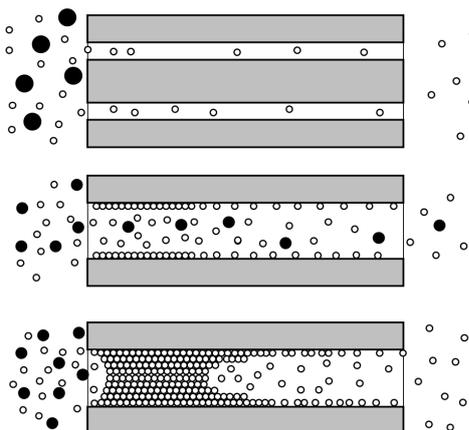


Figure 9: molecular sieving (top), surface diffusion (middle), capillary condensation (bottom)
[Rautenbach and Melin, 2004]

In membranes with **molecular sieving** the parameter that effects the transport rate is the molecular size. Some molecules don't get adsorbed on the pore walls as their diameter is too large. Other components can get into the pores, where they are adsorbed.

In membranes with **surface diffusion**, the flux is a result of two effects, the surface diffusion and the Knudsen diffusion. Surface diffusion is determined by the size of the components and the concentration on the surface whereas Knudsen diffusion is determined by the concentration of the components in the gas phase of the membrane pores.

Capillary condensation occurs if gases with a larger molecular weight have a high critical temperature, such as carbon dioxide, propane, butane or vapors, they interact with the membrane material and get adsorbed at the surface of the membrane. As a result, the concentration of the adsorbed gas increases in the pores of the membrane and the flux of the adsorbed components in the membrane increases. Thus, small inert gases such as hydrogen, nitrogen or helium get blocked for transport and have a lower permeability.

2.8.4 Diffusion in dense membranes

The gas transport in a dense polymer matrix is a result of solution and diffusion of the gas components in the membrane. The model to describe transport through dense membranes is the solution-diffusion model. It states that the permeating components pass the membrane in three steps.

- Sorption on the membrane or solution in the membrane according to the partition coefficient between gas and membrane
- Diffusion in the membrane depending on the activity gradient
- Desorption from the membrane on the permeate phase

The three steps for transport in dense membranes are depicted in figure 10 where μ is the chemical potential, X is the molar fraction, C is the concentration and p is the partial pressure. [Strathmann, 2011]

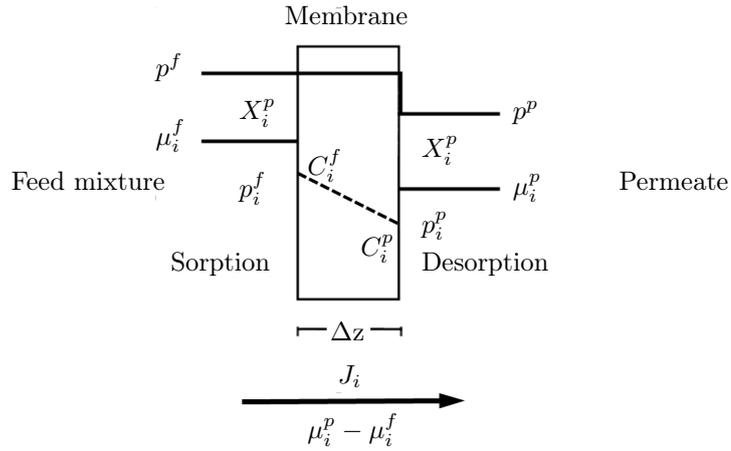


Figure 10: Solution diffusion model, adapted from [Strathmann, 2011]

In case of diffusion in dense membranes, the flux can be described by Fick's first law of diffusion depicted in equation 2.14 which is written for a binary mixture with the diffusion coefficient $D_{AB} = D_{BA} = D$. This assumes an ideal binary liquid solution, where the components have no electrical charges under the driving force of a concentration gradient.

The *diffusion coefficient* in dense materials is described by the Einstein-Smoluchowski equation

$$D = \frac{\lambda u_{th}}{2} \exp\left(-\frac{E_D^{ex}}{RT}\right) \quad (2.30)$$

with λ the free path length, u_{th} the mean thermal velocity and E_D^{ex} the excess energy that is necessary to create an empty space in solid material and for the component to reach the space. The excess energy and also the diffusion coefficient are influenced by the structure of the polymer and the size of the permeating component. With an increase of the molecular weight of the permeating component, the diffusion coefficient and thus the permeability decreases. The excess energy and the diffusion coefficient are higher for rubbery polymers such as elastomers with a low glass transition temperature than for glassy, amorphous and crystalline polymers with a high glass transition temperature.

The mass transport in dense membranes mainly depends on the difference between the chemical potential on the feed and on the permeate side and can be described by the following equation

$$J_i = -L_i \frac{d\mu_i^m}{dz} \quad (2.31)$$

with the phenomenological coefficient L_i . The driving force for the mass transport is the activity gradient of the component permeating through the membrane. The change of the chemical potential of a component i is related to the pressure p and the activity a_i .

$$d\mu_i = V_i dp + RT d\ln(a_i) \quad (2.32)$$

The chemical potential of the gas in the membrane is given by

$$d\mu_i^m = \mu_i^0 + RT d\ln(a_i^m) + V_i^m(p^m - p^0) \quad (2.33)$$

with μ_i^0 the chemical potential of the pure component at standart conditions, p^0 the saturation pressure and V_i^m the partial molar volume in the membrane.

The solution-diffusion model states that the fluids and the membrane are in equilibrium

state and that the potential gradient is continuous: $\mu_i^m = \mu_i^g$.

Further, it is assumed, that the pressure across the membrane is constant and thus the potential gradient is expressed as a concentration gradient. Another assumption is that the fugacity coefficient in the feed and the permeate is equal. The flux across dense membranes is

$$J_i = -D_i \frac{\varphi_i \rho_i^m}{\bar{\gamma}_i p_i^0 M_i} \left(\frac{p_i^p - p_i^f}{\Delta z} \right) \quad (2.34)$$

with D_i the diffusion coefficient, φ the fugacity coefficient, ρ the density, $\bar{\gamma}$ the average activity coefficient of the component i in the membrane, p_i^0 the pressure and M_i the molecular weight. Equation 2.34 shows that a high permeability is obtained for a high diffusion coefficient and a low saturation pressure. A low activity coefficient indicates a high affinity of the component to the membrane.

With the sorption coefficient defined in equation 2.6 and with the permeability defined in equation 2.5, $P_i = D_i k_i$, equation 2.34 can be written as

$$J_i = -P_i \frac{p_i^p - p_i^f}{\Delta z} \quad (2.35)$$

with p_i^p and p_i^f the partial pressure of a component i on the permeate and on the feed side. [Strathmann, 2011]

The partial pressure p_i is related to the molar fraction X_i of a component i in the gas phase by

$$X_i^g p = p_i \quad (2.36)$$

The concentration C_i^m in the membrane is related to the molar fraction of a component i in the membrane by

$$C_i^m = \frac{\rho_i^m \varphi_i p_i}{M_i \gamma_i p_i^0} \quad (2.37)$$

In dense membranes, the sorption and the diffusion coefficient have the most influence on the separation process.

Generally, the diffusion coefficient decreases with the increase of the size of the permeating molecule. On the other side, the sorption coefficient is proportional to the size of the permeating molecules. Therefore, the permeability can increase or decrease for larger molecules, depending on the influence of k and D . [Ohlrogge and Ebert, 2006]

For easily condensable vapors, like carbon dioxide and low molecular weight hydrocarbons, the **diffusion coefficient** depends strongly on the concentration and pressure of the permeating substance. This dependence is based on the glass transition temperature or the free volume in the membrane polymer structure.

The **sorption coefficient** of nonideal gases can be written as

$$k = k_0 \exp \left(-\frac{\Delta H}{RT} \right) \quad (2.38)$$

with k_0 as a pre-exponential factor and ΔH the enthalpy of mixing. Equation 2.38 shows that the enthalpy can have a considerable influence on the sorption behaviour. The sorption of gases in the polymer membrane depends on the type of gas and on the type of polymer. Consequently, the relationship between concentration of the component in the membrane and the applied pressure can be linear or nonlinear. A linear relationship occurs for the sorption of inert gases in rubbery polymers and is described by Henry's law.

For a glassy polymer, the Henry and the Langmuir type of absorption occur and thus the sorption isotherm shows a nonlinear relationship between pressure and concentration.

For permanent gases, like hydrogen and nitrogen, it is assumed, that both, the diffusion and sorption coefficient are independent of concentration and pressure. The concentration in the membrane can be expressed by equation 2.37. [Strathmann, 2011]

Table 1 gives an overview of the commercial membrane processes and the driving forces.

Table 1: Membrane separation processes, structures, materials and driving forces [Strathmann, 2011]

Separation process	Membrane structure	Separation and transport mode	Driving force
Microfiltration	Symmetric macroporous ∅ 0.1-20 μm	Size exclusion, convection	Applied Pressure 0.05-0.2 MPa
Ultrafiltration	Asymmetric macroporous ∅ 2-10 nm	Size exclusion, convection	Applied Pressure 0.1-0.5 MPa
Nanofiltration	Asymmetric mesoporous ∅ 1-5 nm	Size exclusion, diffusion, Donnan exclusion	Applied Pressure 0.3-3 MPa
Reverse osmosis	Asymmetric dense, microporous	Solution and diffusion	Applied Pressure 1-10 MPa
Dialysis	Symmetric porous, dense, ion-exchange	Solution and diffusion	Activity gradient
Electrodialysis	Symmetric ion-exchange	Donnan exclusion ion migration	Electrical potential gradient
Gas and vapor separation	Asymmetric dense or porous ∅ 1-5 μm	Solution, diffusion Knudsen diffusion	Pressure, vapor pressure
Pervaporation	Asymmetric dense	Solution and diffusion	Vapour pressure
Membrane contactor	Various	Diffusion and solution	Chemical potential
Other membranes (fuel cells, etc.)	dense		Electrochemical potential gradient

Microfiltration and *Ultrafiltration* (MF, UF) are used to separate macromolecules, slurry particles or droplets from a solvent. MF separates small particles, viruses, bacteria, etc. UF separates macromolecules with molecular weights in the range of $0.5-500 \times 10^3$ Da. The transport in MF and UF is based on viscous flow.

The membranes used for *Nanofiltration* (NF) can separate molecules with molar masses greater than 300 g/mol.

In NF, and *reverse osmosis* (RO), the separation process is predominated by the solubility of different components in the membrane matrix. The transport is based on diffusion.

The most important application of *reverse osmosis* (RO) is the desalination of seawater. In RO, the applied hydrostatic pressure must be higher than the osmotic pressure.

For *gas separation*, dense and porous membranes are used. In dense membranes, the

transport is based on the solution and diffusion of the species in the membrane, whereas in porous membranes, the transport is determined by the Knudsen diffusion. The Knudsen diffusion depends on the pore size of the membrane and the free path length of the molecules.

The most important application of *dialysis* is to remove substances from the blood and the removal of acids and bases from aqueous solutions using the concentration difference. *Electrodialysis* is the principle behind fuel cells, where the concentration gradient is responsible for the transport of H⁺-Ions from the anode side through a cation-exchange membrane to the cathode side. The electrical potential is the driving force for the mass transport of ions in electrodialysis and chlorine alkaline production. [Strathmann, 2011]

3 Basics of numerical simulation

In fluid mechanics, flow phenomena can be described by partial differential equations which generally cannot be solved analytically. Therefore the equations are discretized into a system of algebraic equations which can be solved numerically by a computer.

3.1 Procedure in numerical simulation

In the following, the components of a numerical solution method are discussed as outlined in [Ferziger and Peric, 2002]. The procedure is categorized into seven steps.

1. Mathematical model

A mathematical model is set up with a set of partial differential equations and boundary conditions. The set of equations depends on the fluid properties like compressibility, viscosity, turbulence, two- or three-dimensional flow, etc.

2. Discretization

The mathematical model needs to be discretized by a system of algebraic equations. Different approaches exist for discretization like finite difference, finite volume, finite element and spectral methods.

3. Coordinate System

The coordinate system and the basis vectors and tensors need to be defined, depending on the formulation of the conservation equations. Usually, cartesian components are used.

4. Numerical Grid

A numerical grid is created to define the locations for solving the discretized equations. The grid divides the computational domain into a finite number of subdomains. The grid or mesh structure can be classified into three types.

- Structured grid

A regular or structured grid is the simplest type of grid. Each grid point can be described by two indices in 2-dimensional or by three indices in 3-dimensional domains. The matrix of the algebraic equation system has a regular structure and allows the application of efficient solvers. Though, it can be difficult to use structured grids in complex geometries.

- Block-structured grid

In case the structured grid in a computational domain is divided into subdomains, the grid is referred to as block-structured grid. Usually, parts of the domain have a coarse grid with large segments and some parts which are of special interest, have a fine, structured grid.

- Unstructured grid

With this type of grid also complex geometries can be meshed. The computational domain is divided into triangles or quadrilaterals in 2-dimensional and into tetrahedra or hexahedra in 3-dimensional shapes. Often such grids can be generated automatically by algorithms. The matrix of the algebraic equation has usually no regular, diagonal structure and consequently the computational effort is higher.

Figure 11 shows two examples of a block-structured and an unstructured 2-dimensional grid.

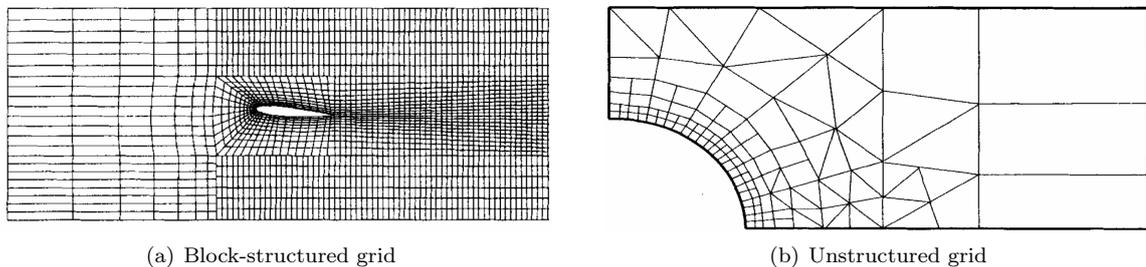


Figure 11: Discretization of the computational domain with grids,[Ferziger and Peric, 2002]

5. Approximation for the discretization

Several approximation methods exist for each discretization type. For the finite difference method, the derivatives are approximated at the grid points. For the finite volume method, the surface and the volume integrals need to be approximated for each control volume. The fluid properties are calculated at a computational node at the center of each control volume. The values at the surface of the control volume are then interpolated.

For the finite volume method the most commonly used interpolation schemes are

- Upwind Interpolation
- Linear Interpolation
- Quadratic Upwind Interpolation (QUICK)

For the finite element method the equations are multiplied by a weight function and then integrated over the computational domain. Linear shape functions are used in each element to ensure continuity across its boundaries.

6. Solution method for equation systems

The discretization results in a large system of non-linear algebraic equations which is usually solved iteratively. For unsteady flows, an elliptic problem has to be solved for each time step with methods which are used for initial value problems. For steady flows, pseudo-time marching iteration schemes can be used. Two solution methods for Navier-Stokes Equations are described in the section 3.3.

7. Convergency

Convergence criteria are necessary to define the end of the iterative solving process. The criteria must ensure that the solution is reached efficiently and has an adequate accuracy. Commonly, the difference between two successive iterations, known as residuals, is measured and the simulation stops when the error is less than the chosen value.

3.2 Structure of CFD packages

A CFD package includes user interfaces to define the problem parameters before the simulation and to examine the results afterwards. Following the definition of [Versteeg and Malalasekera, 1995], three main elements of a CFD package can be distinguished.

- Pre-processor
Pre-processing encompasses the definition of the geometry which is examined, the computational domain. Further the grid is generated where the domain is divided into smaller control volumes or cells.
The physical and chemical phenomena and the fluid properties are defined. For the domain boundaries, appropriate boundary conditions are defined.
- Solver
Basically, the solver performs the steps discretization, approximation and the solution of the equation system as outlined in steps 2, 5 and 6 in the previous section 3.1.
- Post-processor
Post-processing covers tools for data manipulation and visualization of the simulation data, external to the solver code. This includes, among others, depiction of the domain geometry and the grid, vector- and contour plots or animation of dynamic results.

3.3 Solution method for Navier-Stokes equations

The most widely used applications are the pressure-implicit split-operator algorithm (PISO) and the semi-implicit method for pressure linked equations algorithm (SIMPLE). PISO is a transient solver, whereas SIMPLE is a steady state solver. Both algorithms iteratively solve the equations for velocity and pressure which is referred to as pressure-velocity coupling. The process is defined by the iterative solution of the unsteady governing differential equations until a convergent steady-state is reached.

3.3.1 SIMPLE algorithm

The iterations in one time step are categorized into outer iterations in which the coefficients and source matrices are updated and inner iterations where the coefficients are fixed. On each outer iteration the following momentum equations are solved

$$A_P^{u_i} u_{i,P}^{m*} + \sum_l A_l^{u_i} u_{i,l}^{m*} = Q_{u_i}^{m-1} - \left(\frac{\delta p^{m-1}}{\delta x_i} \right)_P \quad (3.1)$$

whereas m stands for the number of the outer iteration. Inner iterations refer to the solution of linear equations like the pressure correction equation 3.6. At the beginning of each outer iteration, the coefficients Q and p are taken from the previous step. P stands for an arbitrary velocity node, l describes the neighbor points of the discretized momentum equation. Q represents all values which are computed explicitly in terms of u_i^{m-1} and other values at the new time level. Equation 3.1 contains non-linear terms and is coupled to other differential equations. The coefficients A and possibly also the source term Q can depend on the solution u_i^m and thus an iterative solution is necessary. [Ferziger and Peric, 2002] Equation 3.1 is solved for the velocity. Then the velocity is corrected so that it satisfies the continuity equation 3.15.

$$u_{i,P}^m = \tilde{u}_{i,P}^{m*} - \frac{1}{A_P^{u_i}} \left(\frac{\delta p^m}{\delta x_i} \right)_P \quad (3.2)$$

This gives a discrete Poisson equation for the pressure.

$$\frac{\delta}{\delta x_i} \left[\frac{\rho}{A_P^{u_i}} \left(\frac{\delta p^m}{\delta x_i} \right) \right]_P = \left[\frac{\delta(\rho \tilde{u}_i^{m*})}{\delta x_i} \right]_P \quad (3.3)$$

After solving the Poisson equation 3.3, the velocity field for the new iteration is obtained from equation 3.2. This gives a velocity field which is satisfying for the continuity equation but not for the momentum equations 3.1. Therefore another outer iteration is started until the velocity field satisfies also the momentum equation.

In the SIMPLE algorithm, the correct pressure p is obtained from

$$p = p^* + p' \quad (3.4)$$

where p^* is the guessed pressure and p' is the pressure correction. The goal is to improve p^* so that the starred velocity field, \mathbf{v}^* , satisfies the continuity equation. The resulting velocity components are composed of

$$u = u^* + u' \quad v = v^* + v' \quad w = w^* + w' \quad (3.5)$$

These values are substituted in the momentum equations 3.2 and inserted into the discretized continuity equation which leads to the pressure correction equation

$$\frac{\delta}{\delta x_i} \left[\frac{\rho}{A_P^{u_i}} \left(\frac{\delta p'}{\delta x_i} \right) \right]_P = \left[\frac{\delta(\rho u_i^{m*})}{\delta x_i} \right]_P + \left[\frac{\delta(\rho \tilde{u}'_i)}{\delta x_i} \right]_P \quad (3.6)$$

Important SIMPLE operations are as follows according to [Pantakar, 1980]

- Guess the pressure field p^*
- Solve the discretized momentum equations 3.1 to obtain the velocities u^* , v^* , w^*
- Solve the discretized equation 3.6 for the pressure correction p'
- Calculate p from equation 3.4.
- Calculate u , v , w from their starred values u^* , v^* , w^* and from the pressure corrections p' , using the velocity-correction formulas 3.5
- Solve the discretization equations for other Φ 's like temperature, concentration or turbulence if they influence the flow field.
- Set the corrected pressure p as the new guessed pressure p^* and repeat the procedure until convergence.

It turned out that the SIMPLE algorithm shows a better convergence, if the pressure is not fully but rather partially corrected in each loop with an under-relaxation parameter α_p .

$$p = p^* + \alpha_p p' \quad (3.7)$$

In this case, only a fraction of p' is added to p^* whereas α_p is set to around 0.7. Also the velocities \mathbf{u}^* are usually underrelaxed with a parameter α of around 0.3 for solving the momentum equations.

Revised versions of the SIMPLE algorithm have been worked out, called SIMPLE revised (SIMPLER) which have shown even faster convergence. Basically SIMPLER starts the loop with a guessed velocity field and therefore extracts the pressure field from a given velocity field. Also other variations of the algorithm, named SIMPLE corrected (SIMPLEC) are described in the literature which promise faster convergence than SIMPLE. [Pantakar, 1980].

In the SIMPLE algorithm, step 2 refers to the *momentum predictor* step which is always executed. In the PISO algorithm, the momentum predictor can be explicitly specified in the OpenFOAM fvSolution dictionary.

3.3.2 PISO algorithm

Two characteristics distinguish PISO from SIMPLE. Firstly in PISO, no under-relaxation is applied. Secondly, whereas SIMPLE makes one correction, for the PISO algorithm, a second pressure-correction equation is solved. Both velocities and pressure are corrected again in another step with u'' and p'' .

Thus the corrected fields in the PISO algorithm can be written as

$$u_i^m = m_i^{m*} + u_i' + u_i'' \quad p^m = m^{m-1} + p' + p'' \quad (3.8)$$

In OpenFOAM the number of outer iterations is defined as *nOuterCorrectors* and the number of inner iterations is defined as *nCorrectors* in the fvSolution dictionary. The number of inner iteration defines how often the pressure correction equation 3.6 is solved.

Non Orthogonal Corrector

Both solvers, SIMPLE and PISO, allow to account for non-orthogonal meshes. In an orthogonal mesh, each face normal is parallel to the vector between the centres of the cells that the cell connects. [OpenFOAM, 2014] The following equation shows the procedure when a non orthogonal corrector is used.

$$\nabla^2 p = f(U, \nabla p) \rightarrow p_{new} \rightarrow \nabla^2 p_{new} = f(U, \nabla p_{new}) \rightarrow \text{repeat } n \text{ times} \quad (3.9)$$

The pressure field is calculated more often from the pressure equation 3.3 with the new pressure value, according to the number specified in *nNonOrthogonalCorrectors* in the fvSolution dictionary. [openfoamwiki.net, 2016a]

In case the *momentum predictor* is set to *on*, the pressure field from the previous time step is used to solve the discretized momentum equation 3.1. Otherwise, if the momentum predictor is set to *off*, solving the momentum equation is omitted in each time step of the PISO algorithm.

3.4 Finite volume method

To obtain the equations which describe fluid motion, generally a small control volume is considered. Fluid flows through the volume with the requirement that mass and energy are conserved. The governing equations in fluid mechanics are mathematical statements of the following conservation laws. [Bird et al., 2002]

- The mass of the fluid is conserved
- The rate of change of momentum equals the sum of the forces on a fluid particle (Newton's second law)
- The rate of change of energy is equal to the sum of the rate of heat addition to and the rate of work done on a fluid particle (first law of thermodynamics)

The fluid is regarded as a continuum, where the molecular structure and motion is ignored. The behavior of the fluid is described in macroscopic properties, such as velocity, pressure, density and temperature and their space and time derivatives. In the finite volume method, small elements of fluid as in figure 12 are considered.

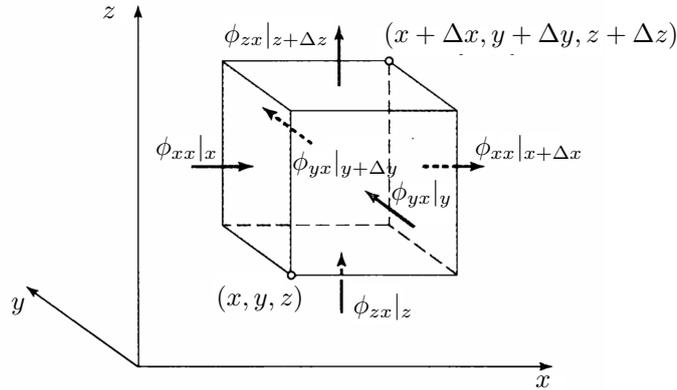


Figure 12: Finite volume, [Bird et al., 2002]

The six arrows indicate the flux ϕ through the faces of the fluid element. The changes of the mass, momentum and energy of these fluid elements result either from the fluid flow across the boundary faces or from sources inside the element.

The momentum and energy equation make statements regarding the changes of properties of the fluid in space and time. The variable ϕ represents the value of a property per unit mass. Hence, the total or substantive derivative of ϕ is

$$\frac{D\phi}{Dt} = \frac{\partial\phi}{\partial t} + \frac{\partial\phi}{\partial x} \frac{\partial x}{\partial t} + \frac{\partial\phi}{\partial y} \frac{\partial y}{\partial t} + \frac{\partial\phi}{\partial z} \frac{\partial z}{\partial t} = \frac{\partial\phi}{\partial t} + \mathbf{u}\nabla\phi \quad (3.10)$$

As equation 3.10 defines the rate of change of property ϕ per unit mass, it needs to be multiplied by the density for the change per unit volume.

3.5 Equations in fluid mechanics

The finite volume method for fluid motion generally gives five partial differential equations, the mass conservation equation, three momentum equations and the energy equation. Together with equations of state, like the ideal gas equation for gases,

$$p = \rho RT \quad (3.11)$$

and the equation of state for internal energy

$$e = C_V T \quad (3.12)$$

these equations determine the values of the pressure p , the density ρ , the temperature T , the velocity \mathbf{v} and the internal energy i , whereas thermodynamic equilibrium is assumed. In case of compressible fluids, the equations of state link the energy equation to the mass- and momentum equations because the density is linked to the pressure and temperature. In case of incompressible fluids, without density variation, the equations of state can't link the energy equation to the mass- and momentum equations. Often the flow field is solved by considering only the continuity and the momentum equation. The energy equation is only solved if heat transfer occurs. [Versteeg and Malalasekera, 1995]

3.5.1 Continuity equation

The mass balance for the fluid element states that the rate of increase of mass in the fluid element equals the net mass flow rate into the fluid element across the surfaces S of the

volume V .

$$\frac{d}{dt} \int_V \rho \, dV = - \int_S \rho \mathbf{v} \cdot \mathbf{n} \, dS \quad (3.13)$$

where \mathbf{n} is the unit outward normal vector and \mathbf{v} the velocity vector. With the Gauss theorem, the unsteady, three dimensional mass balance or continuity equation in a compressible fluid can be written in vector notation

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (3.14)$$

The first term is the rate of change in time of the density and the second term is the net mass flow across the boundaries of the element. In case of an incompressible fluid with constant density equation 3.14 becomes

$$\nabla \cdot \mathbf{u} = 0 \quad (3.15)$$

3.5.2 Momentum equation

According to Newton's second law, the rate of change of momentum of a fluid equals the sum of the forces on the particle. The rate of increase of the momentum in x-, y- and z-direction per unit volume of a fluid particle are expressed by

$$\rho \frac{Dx}{Dt}, \quad \rho \frac{Dy}{Dt}, \quad \rho \frac{Dz}{Dt}. \quad (3.16)$$

With the total time rate of change of \mathbf{v} Newton's second law can be expressed by

$$\int_V \rho \frac{D\mathbf{v}}{Dt} \, dV = \int_V \rho \mathbf{f} \, dV - \int_S p \mathbf{n} \, dS = \sum F. \quad (3.17)$$

The fluid particles are exposed to two types of forces

- surface forces
 - pressure forces
 - viscous forces
- body forces
 - gravity forces
 - centrifugal forces
 - Coriolis force
 - electromagnetic forces

In the momentum equation, generally the surface forces are expressed as separate terms and body forces as source terms. [Versteeg and Malalasekera, 1995]

The momentum equation per unit volume for an inviscid fluid is expressed by the Euler equation which is valid for both compressible and incompressible flow.

$$\frac{\partial}{\partial t} \rho \mathbf{v} + \nabla \cdot \rho \mathbf{v} \mathbf{v} = -\nabla p + \rho \mathbf{f} \quad (3.18)$$

In case of viscous flow, the viscosity is expressed with the general stress tensor $\underline{\sigma} = -p\mathbf{I} + \underline{\tau}$, which considers stress in any direction. Therefore, the momentum equation for a viscous fluid is

$$\underbrace{\rho \frac{\partial}{\partial t} \mathbf{v}}_{\text{rate of increase of momentum}} + \underbrace{\nabla \cdot \rho \mathbf{v} \mathbf{v}}_{\text{rate of momentum addition by convection}} = \underbrace{-\nabla p - \nabla \cdot \underline{\tau}}_{\text{rate of momentum addition by molecular transport}} + \underbrace{\rho \mathbf{f}}_{\text{external force on fluid}} \quad (3.19)$$

where ∇p denotes the gradient of pressure and $\nabla \cdot \underline{\tau}$ denotes the divergence of the tensor $\underline{\tau}$. [Fletcher, 1988]

In cartesian coordinates, the momentum equations for a viscous fluid are

$$\begin{aligned} \rho \frac{Du}{Dt} &= -\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + \rho f_x \\ \rho \frac{Dv}{Dt} &= -\frac{\partial p}{\partial y} + \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} + \rho f_y \\ \rho \frac{Dw}{Dt} &= -\frac{\partial p}{\partial z} + \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z} + \rho f_z \end{aligned} \quad (3.20)$$

In the Navier-Stokes equations the viscous stresses are related proportionally to the rates of deformation or strain which is distinctive for Newtonian fluids. The rate of deformation involves two constants of proportionality. The linear deformation rate, expressed by the dynamic viscosity μ , to relate stresses to linear deformation and the volumetric deformation rate, expressed by the value $-\frac{2}{3}\mu$ to relate stresses to volumetric deformation.

$$\begin{aligned} \tau_{xx} &= -\frac{2}{3}\mu \operatorname{div} \mathbf{u} + 2\mu \frac{\partial u}{\partial x}, \\ \tau_{yy} &= -\frac{2}{3}\mu \operatorname{div} \mathbf{u} + 2\mu \frac{\partial v}{\partial y}, \\ \tau_{zz} &= -\frac{2}{3}\mu \operatorname{div} \mathbf{u} + 2\mu \frac{\partial w}{\partial z}, \\ \tau_{xy} = \tau_{yx} &= \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right), \\ \tau_{xz} = \tau_{zx} &= \mu \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right), \\ \tau_{yz} = \tau_{zy} &= \mu \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) \end{aligned} \quad (3.21)$$

Inserting equations for the viscous stresses 3.21 into the momentum equations 3.20 gives the Navier-Stokes equations. In vector notation, they are given by

$$\rho \frac{D\mathbf{v}}{Dt} = \rho \mathbf{f} - \nabla p - \frac{2}{3} \nabla (\mu \nabla \cdot \mathbf{v}) + 2 \nabla \cdot \left(\mu \frac{1}{2} \frac{\partial v_i}{\partial x_j} + \mu \frac{1}{2} \frac{\partial v_j}{\partial x_i} \right) \quad (3.22)$$

The Navier-Stokes equations 3.22 are applicable to viscous compressible flow of Newtonian fluids. [Versteeg and Malalasekera, 1995]

3.5.3 Energy equation

According to the first law of thermodynamics, the rate of change of internal energy and kinetic energy is equal to the rate of heat addition and the rate of work done. The thermal

energy equation for a control volume is

$$\int_V \rho \frac{D}{Dt} \left(e + \frac{v_i v_i}{2} \right) dV = \int_V \rho \mathbf{f} \cdot \mathbf{v} dV + \int_S \mathbf{n} \cdot (\mathbf{v} \underline{\boldsymbol{\sigma}} - \dot{\mathbf{q}}) dS \quad (3.23)$$

where e stands for the internal energy and the second term on the left side for the mechanical energy. The first and the second term on the right side express the work resulting from volume and surface forces and $\dot{\mathbf{q}}$ is the surface rate of heat transfer. From 3.23 the mechanical energy is removed and the Gauss theorem is applied. This gives the thermal energy equation for an arbitrarily small volume.

$$\rho \frac{De}{Dt} + p \nabla \cdot \mathbf{v} = \Phi - \nabla \cdot \mathbf{q} \quad (3.24)$$

with the dissipation function from irreversible viscous work $\Phi = \underline{\boldsymbol{\tau}} \cdot \mathbf{v}$. In the Fourier's law, the heat transfer $\dot{\mathbf{q}} = -k \nabla T$ is a linear function of the temperature gradient. In case of airflow, the fluid is modeled as ideal gas and the internal energy and enthalpy are related to the temperature

$$\begin{aligned} e &= c_v (T - T_{ref}) \\ h &= c_p (T - T_{ref}) \end{aligned} \quad (3.25)$$

with c_v the specific heat at constant volume and c_p the specific heat at constant pressure. [Fletcher, 1988]

3.6 OpenFOAM

The Open Source Field Operation and Manipulation (OpenFOAM) C++ library is designed to create executables or applications such as solvers and utilities. The results which are presented in this work are based on calculations and simulations from the OpenFOAM version 2.3.1. OpenFOAM is a free software which can be modified under the terms of the GNU General Public License. The majority of the solvers are used to solve different problems in fluid mechanics. Different solvers can be used to account for compressible or incompressible, single-phase or multi-phase, transient or steady-state, isothermal or non-isothermal, trans- or super-sonic and for other fluid properties. [OpenFOAM, 2014] Utilities encompass, among others, tools for pre-processing, mesh generation, mesh conversion, mesh manipulation and for post-processing of simulation data. An overview of the utilities which are provided by OpenFOAM is documented by CFD direct. [CFDDirect, 2016]

Different mesh manipulations applications like blockMesh or snappyHexMesh can be used to create the base mesh inside or around geometries where the differential equations are solved. Also several utilities allow the mesh conversion from third party software like ANSYS FLUENT[®], CFX[®] or Star-CCM+[®] to the OpenFOAM format. Also it is possible to convert the OpenFOAM mesh to data formats which can be used by third party software. The utility *checkMesh* checks for several parameters of the mesh quality like cell openness, topology, aspect ratio, cell volume, non-orthogonality or skewness. Other important post processing tools are the utilities *yPlusRAS* and *yPlusLES* which allow the calculation of y^+ values at walls.

An utility which is especially helpful for cases with a big computational effort is the possibility to decompose the geometry and the associated fields in order to run cases on several

CPUs at the same time. If cases are decomposed, OpenFOAM distributes the whole mesh domain into a predefined number of domains, which corresponds to the number of processors used. Thus each processor only calculates the flow field in a certain subdomain. After finalization of the simulation, the whole mesh domain is reconstructed from the single processor solutions. In order to gain an advantage in calculation time by decomposing the case, the decomposed mesh on each CPU must be large enough in order to compensate for the overhead costs expressed in time needed for the data exchange between processors.

In order to post-process and analyze the simulation data, the program ParaView version 4.1 is used. With OpenFOAM, a utility allows the conversion of data to the VTK format which is readable by ParaView. The application is built on top of the Visualization ToolKit (VTK) C++ class libraries. ParaView is a freely available, open source, multi-platform software system for two- or three-dimensional computer graphics. An advantage of ParaView is its good documentation in the internet. [ParaView, 2015].

Especially for the post processing of fluid dynamic problems, ParaView allows the 2-D and 3-D visualization of geometries and its meshes. Certain domains of the mesh like volumes, cells, points and faces can be examined. Vector field data can be visualized and colored through contour plots, streamlines, vector arrows, cones, glyphs etc. Also ParaView allows the extraction and analysis of data like field variables from certain areas of the calculated domain. Subregions can be extracted by cutting or clipping the geometry. The tool *plot over line* allows the extraction of data along a line through the geometry. ParaView allows the user to reuse and to automatize some post processing tasks like visualization of certain domains or extraction of certain data with the Python programmable filter by recording actions and repeating them automatically. The integration of the python programming language makes Paraview scriptable.

For computationally and memory expensive data processing, ParaView can be run in parallel on several computers using MPI.

3.6.1 membraneFoam solver

The newly developed base solver, named *membraneFoam* is based on the solvers *chtMultiRegionFoam* and *reactingFoam*, whereas source and sink terms on the boundaries are added. Thus the new solver can handle heat and mass transfer from one region to another. The solver *chtMultiRegionFoam* is a combination of *heatConductionFoam* and *buoyantFoam* for solving transient heat transfer between a solid region and a fluid region. *reactingFoam* is a solver that accounts for reactions and the resulting changes of enthalpy taking place in the computational domain. [openfoamwiki.net, 2016b]

Both, *chtMultiRegionFoam* and *reactingFoam* as well as *membraneFoam* are basically transient solvers and use the PIMPLE algorithm.

PIMPLE algorithm

The PIMPLE algorithm is a combination of the properties of SIMPLE and PISO. It uses the PISO algorithm, which has a transient solver for the momentum equation, with more than one correction of the pressure field. Also it uses the idea of relaxation factors from the SIMPLE approach which limits the influence of new values Φ^n .

$$\Phi^n = \Phi^{n-1} + \alpha_\Phi (\Phi^{new} - \Phi^{n-1}) \quad (3.26)$$

A new value Φ^n is the sum of the old value and the value which is calculated in the current step. The introduction of underrelaxation slows down the time to convergence

but also makes the simulation more stable. In *membraneFoam*, by default, the momentum predictor is set to *on*, the number of correctors is set to 2 and the number of non orthogonal correctors is set to *zero*.

3.6.2 LTSmembraneFoam solver

In the solver settings, by default, the momentum predictor is set to *off*, the number of correctors is set to *one* and the number of non orthogonal correctors is set to *zero*.

membraneFoam is a transient solver but in membrane processes often steady state conditions need to be analyzed. Therefore, the local-time stepping method is applied to the membrane solver. The resulting new solver is named *LTSMembraneFoam*.

In the **Local Time Stepping (LTS)** method, in contrast to global-time stepping, the time step is set as high as possible for every cell so that the steady-state is reached faster. The LTS method is already implemented in OpenFOAM for the solvers *interFoam* and *reactingFoam*, where they are known as *LTSInterFoam* and *LTSReactingFoam*.

Basically the LTS version of the solver makes the time step as large as possible by simultaneously satisfying the Courant number. The Courant number according to [Courant et al., 1928] is defined as

$$c = \frac{u}{\frac{\Delta x}{\Delta t}} \quad (3.27)$$

and states that an observed value Φ may not move on further than one cell in one time step. Thus the condition $c < 1$ should hold for all time steps so that the algorithm is stable.

In the LTS version, first the time steps are maximized and then smoothed in order to reduce sudden sharp changes in the time step field. Also a damping coefficient can be set to prevent oscillations.

Permeance

The permeances in *membraneFoam* as well as in *LTSMembraneFoam*, following equation 2.12, are defined as a source term

$$Q_i = \frac{Q_{STP}}{T^n} e^{\frac{E}{RT}} \quad (3.28)$$

which gives an Arrhenius relation between the permeance and the temperature. By default, and for all simulations, the activation energy E and the temperature exponent are set to *zero* so that the permeance corresponds to the permeance at STP conditions. The permeances at STP conditions must be taken from lab experiments or from the literature.

By default, the reference temperature and pressure conditions in the *LTSmembraneFoam* solver code which is used for the simulations are defined at a pressure of 10^5 Pa and a temperature of 298.15 K.

The permeate flow, following equation 2.35, across cell i is defined as

$$J_k = -Q_k \cdot (p_{k,i}^p - p_{k,i}^f) \cdot Area \quad (3.29)$$

The permeate flow J_k of a component k is corrected with the ideal gas equation according to

$$J_{corr} = J_k \cdot \frac{p_{ref}}{p_i} \cdot \frac{T_i}{T_{ref}} \quad (3.30)$$

With the following equation the transmembrane flow is converted from a volume flow to a mass flow as a source term

$$J = J_{corr} \cdot \frac{\rho(k, p_i, T_i)}{V_i} \quad (3.31)$$

with the density depending on the specie k , the pressure and the temperature and the volume of cell i .

3.7 Hardware for simulations

All simulations are run on multi-core processors. Therefore, three kinds of servers are used which have different processor types. The nodes the server *Navier* have AMD FX™-8350 Eight-Core processors with 1.4 GHz and 4.0 GHz respectively. The nodes the server *Foam*, have AMD FX™-8320 Eight-Core processors with 3.2 GHz. The nodes on the server *Caelv* have AMD Opteron™-8356 processors with 1.2 GHz. The majority of the simulations is run on the *Foam* server. Table 2 shows the specifications of both servers.

Table 2: Server hardware

Server name	Processor	RAM [GB]	Clock rate [GHz]	Cache [KB]
Caelv	AMD Opteron™-8356 4-Core	128	1.2	512
Navier	AMD FX™-8350 8-Core	32	1 or 4	2048
Foam	AMD FX™-8320 8-Core	16	3.2	2048

4 Test Cases

4.1 Hollow fiber membrane with seven fibers

As a first step, a small hollow fiber membrane with seven fibers and five exits along the shell is created. The module has five exits for the permeate, to investigate five different module operations like fully co-current flow, fully counter-current flow and combinations of both flow patterns.

4.1.1 Dimensions of the seven fiber module

The module has a total length of 540 mm and an outer shell diameter of 6 mm. The feed inlet and retentate outlet diameter are 3 mm and the permeate outlets have a square cross section with a lateral length of 1 mm.

Each of the fibers has a length of 500 mm and a diameter of 1 mm. As a simplification, the fibers are modeled with no wall thickness and thus, inner and outer diameter are the same.

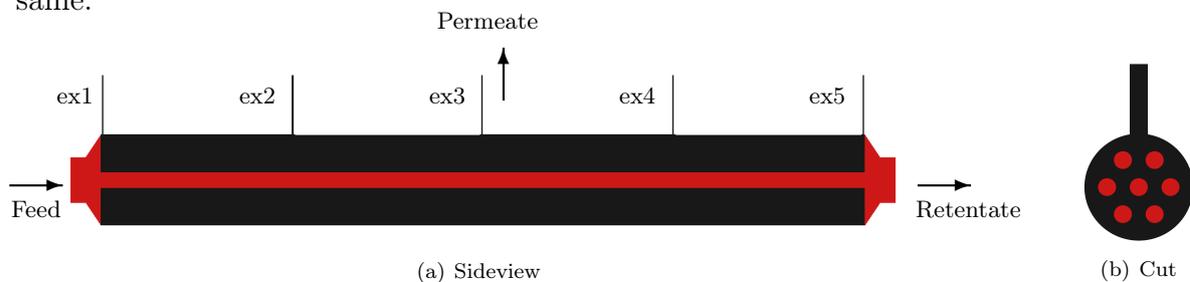


Figure 13: Membrane module with 5 exits, scaled 1:10

The module and the positioning and naming of the five exits are depicted in figure 13. The first exit is near the feed inlet and the last exit is near the retentate outlet.

4.1.2 Mesh creation

The geometry was created and meshed with the commercial software ANSYS GAMBIT 2.4.6. Subsequent to the geometry generation, the module is meshed with hexahedral cells. Each fiber has 43 cells in the cross section and 349 cells along the whole length, 80 cells between each exit, 5 cells below each exit and 1 cells on the far left and on the far right end of the fibers. For a better resolution of the flow, the mesh is refined near the exits as depicted in figure 14(a), which is scaled by 1:20 for a better visibility.

Figure 14(b) shows the cells distribution in the cross section of the middle of the module. Both, the inlet domain on the left, before the feed stream is split into the seven fibers, and the outlet domain, where the seven retentate streams are merged, have 30 cells in flow direction.

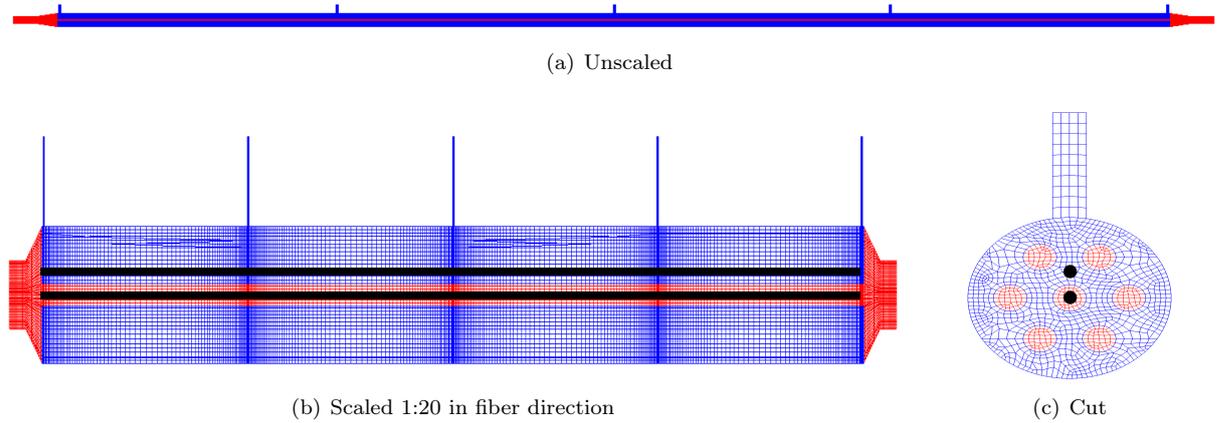


Figure 14: Membrane module with 7 fibers, [Haddadi et al., 2015a], [Haddadi et al., 2015b]

The geometry consists of two regions, the hollow fibers region with 170 809 cells and the shell region with 278 455 cells. Thus, the total cell count of the module is 449 264 cells. The feed stream enters the module from the inlet on the left and is split up into 7 streams in order to increase the surface area between retentate and permeate.

Two black lines in figure 14(b) and two black points in (c) indicate the location where the data is extracted for post-processing. The flow and the fluid characteristics such as velocity and pressure are evaluated.

The examination of the mesh quality with the OpenFOAM tool *checkMesh* resulted in no errors for both regions. Parameters like aspect ratio, min. cell volume, mesh non-orthogonality, max. skewness are in an acceptable low range for the hollow fiber and shell region.

The mesh check with GAMBIT shows no inverted elements and passes the skewness test and all elements have a skewness lower than 0.9.

4.1.3 Simulation settings in OpenFOAM

For the test case, the feed consists of three species, carbon dioxide, methane and oxygen. On the retentate side, the pressure is set to 9 bar at the outlet and the permeate pressure is 1.1 bar. Thus the transmembrane pressure difference is 8 bar, whereas no gravity is considered in the model.

The temperature at both, the retentate and permeate side is 316.5 K. The velocity on the inlet is set to 0.1 m/s, which results in a feed mass flow of 6.2×10^{-6} kg/s. The module is configured to separate carbon dioxide from the mixture and therefore carbon dioxide has the highest permeance. The feed composition and the permeances are listed in table 3.

Table 3: Feed composition

Gas name	Formula	Mass fraction	Mole fraction	Permeance $\left[\frac{m^3(STP)}{Pa \cdot s \cdot m^2} \right]$
Carbon Dioxide	CO ₂	0.582	0.3391	5.91×10^{-10}
Methane	CH ₄	0.406	0.6512	1.59×10^{-11}
Oxygen	O ₂	0.012	0.0097	1.36×10^{-10}

In each case which is investigated only one exit was set as open and the other four exits

are set as closed. In case one, only exit one on the shell side is open, which results in a counter-current flow in the module. On the contrary, in case five, only exit five was set as open in order to obtain co-current flow. For the cases 2,3 and 4, both flow pattern, co- and counter current occur.

Table 4: Input parameters for test cases

Case number	Open exit	Flow pattern counter-current : co-current [%]
1	ex 1	100:0
2	ex 2	75:25
3	ex 3	50:50
4	ex 4	25:75
5	ex 5	0:100

4.1.4 Flow regime

With the operating conditions specified above for the feed, the density is around 10 kg/m^3 and the dynamic viscosity of ambient air is around $19.4 \times 10^{-6} \text{ Pa}\cdot\text{s}$. With equation 2.20 this results in a rough estimation of the Reynolds number of around 50.

Thus, the flow regime is assumed to be laminar and the usage of a turbulence model is omitted. Both, the turbulence model and the RAS model are set to *laminar*.

As the flow in the hollow fibers is laminar in all cases, the refinement of boundary layer cells to account for the viscous sublayer is omitted.

The only case, where a refined mesh is used is for visualizing and comparison of the influence of concentration polarization with coarser meshes. The refinement is only based on expectations to accurately display the effect of concentration polarization. The refinement according to the calculation of y^+ values is not considered as no turbulence models are used in the simulations.

4.2 LTSMembraneFoam validation test cases

The LTSMembraneFoam solver code is validated against a process simulation model with the software Aspen Custom Modeler[®] (ACM V7.3) distributed by Aspen Technology, Inc., Burlington, USA, 2011. The ACM model is explained in detail in chapter 6.

4.2.1 Geometry and mesh for CFD cases

The seven fiber module, introduced in section 4.1 is also used for the validation of the LTSMembraneFoam solver.

The geometry for the validation cases, depicted in figure 13, is the same module as the one which was used for the examination of the permeate outlet positioning. Also the mesh is the same as depicted in figure 14.

4.2.2 Simulation settings CFD and ACM

The CFD simulation settings for pressure, temperature, species and mass fractions, for both, the retentate and the permeate side, are identical to the values which are illustrated in chapter 4.1.3.

The permeances for the species CO_2 , CH_4 and O_2 are the same as listed in table 3 for the ACM model as well as in the CFD model. The influence of diffusion and sorption coefficients on the permeance as described in equation 2.8 is neglected.

For validation of the *LTSMembraneFoam* solver, five fully counter-current cases with five different inlet velocities are investigated. Only exit 1 was set as open and all other exits are set as closed.

The settings for each case are listed in table 5. Thus, the influence of different velocities and mass flow on the separation efficiency and on the permeate composition is investigated.

Table 5: Validation cases CFD and ACM

Case number	Inlet Velocity [m/s]	Mass flow [kg/s]
1	0.01	6.21×10^{-7}
2	0.02	1.24×10^{-6}
3	0.05	3.11×10^{-6}
4	0.1	6.21×10^{-6}
5	0.5	3.11×10^{-5}

For the validation of the CFD cases, the highest velocity is 0.5 m/s. With equation 2.20, this results in a Reynolds number of roughly 290.

It is assumed, that the flow regime is laminar and that the usage of a turbulence model can be omitted. Both, the turbulence model and the RAS model are set to *laminar*.

In order to ensure that the simulations of the CFD validation cases are convergent, the numerical errors and the physical errors are checked. The numerical errors are examined with the trend of the residuals over each iteration step. The physical errors are analyzed with the course of the mass balance for the species over each simulation. The convergence is checked for all cases with different velocities. The convergence check is described exemplary for two cases in chapter 7.

4.3 Hollow fiber membrane with one fiber

In order to investigate further cases and validate the results with literature values, a simple hollow fiber membrane module with one fiber is created. The single fiber module is depicted in figure 15.

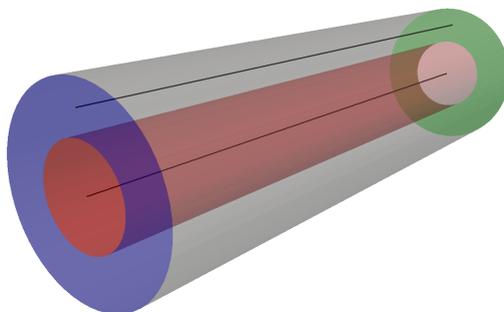


Figure 15: Single fiber membrane module

In case the flow pattern is set to co-current in the single fiber membrane module, the permeate outlet is on the right side, colored as green and the left blue side is set as a wall. In case the flow pattern is set to counter-current, the permeate outlet is on the left side, colored as blue and the green surface is set as a wall.

In both flow patterns, co- and counter-current, the feed inlet is on the left side of the inner fiber, colored as red. The retentate outlet is on the right side of the inner fiber as depicted in figure 15.

The data for the retentate zone is extracted from the center line in the middle of the module. For the permeate zone the data is extracted from the middle of the permeate area, along a radius of 0.0003 m. Both lines are colored black and depicted in figure 15.

The following table 6 shows the examined inlet velocities and the corresponding mass flow in a 35 fiber module.

Table 6: Standard values for single fiber membrane

velocity [m/s]	mass flow [Nl/h]
3.79	60
5.68	90
7.58	120
9.47	150
11.37	180

The goal with this simple single fiber module is to make further studies of co and counter-current flow at different velocities and pressures. Further, the effects of different permeabilities and courant numbers are analyzed to investigate the reliability of the solver.

All cases which are investigated and listed below in table 8 use the same membrane module with a single fiber.

4.3.1 Dimensions of the test case

The length of the module is 0.28 m, the inner fiber diameter is 0.0004 m and the shell diameter is 0.0008 m. As the CFD solver is not capable of handling mass transport through porous layers like a membrane wall, the wall thickness is set to zero.

4.3.2 Mesh creation

In a first step, an attempt was made to create the mesh with the OpenFOAM tool *snappyHexMesh* was used which is supplied with the OpenFOAM distribution 2.3.1. This tool can generate 3-dimensional hexahedra meshes inside or outside of predefined surfaces. The meshing region can be an object with one single region or an object with several regions. Compared to other meshing software packages, *snappyHexMesh* has the following advantages:

For creating and meshing objects, no commercial software package is ultimately necessary. The geometry can be created with any CAD program like CATIA, FreeCAD, etc. As for the geometry, only the surface data is required, the files need to be in .stl, .nas or .obj format. For the meshing, the OpenFOAM environment is sufficient and no further software is necessary. Furthermore, the meshing process can be run in parallel mode. If high computational capabilities are available, high quality meshes can be generated in little time.

The structure of the folders is similar to the OpenFOAM case directories, whereas for the meshing, only the folders *constant* and *system* are used. Meshing with *snappyHexMesh* can be divided into three basic steps:

1. **Castellating:** The region is defined by its surrounding surfaces. The mesh around each surface can be split up to a defined refinement level. Also, special surface areas and regions can be defined and refined separately. The cells which are beyond a region set by a predefined point are deleted. In the *snappyHexMeshDict* file in the system folder, the refinement around the surfaces can be set. In figure 16, the refinement level of one cell is depicted. As the cells are 3-dimensional, the refinement of one cell in each level results in 8 smaller cells.

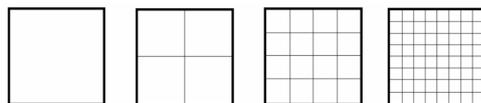


Figure 16: Refinement level 0, level 1, level 2, level 3

2. **Snapping:** Moves the edges of the cells to the surface boundary
3. **Layering:** Creates additional layers in the boundary region.

Also the *blockMeshDict* in the folder *polyMesh* is necessary in order to define the base mesh around the geometry. Starting from the base mesh, created with *blockMesh* the mesh is castellated, snapped or layered around the surfaces.

It is possible to execute either only step 1 or step 1 and step 2 or all three steps together.

With *snappyHexMesh* a considerably bigger mesh must be created to avoid corrupted cells on the boundaries. The basic version of *snappyHexMesh* needs low aspect ratio, almost cubic, cells. Therefore it is not possible to apply high grading along the length of

the fiber. For the dimensions as described in section 4.3.1 the retentate cross section was created with 32 cells and the permeate cross section with 92 cells. With the basic settings of *snappyHexMesh* the fiber needs to have about 2000 cells along the length for a good cell quality on the walls and on the boundaries. This resulted in an overall mesh count of 246 456 cells which was considered to be too high for the simple geometry. Exceeding an aspect ratio of around two, which is necessary in order to keep the total mesh count low, results in corrupted cells on the walls.

In order to obtain a good mesh resolution in the cross section, considerably more cells would be needed for the overall module. It is expected that applying a higher mesh resolution in the cross section results in a too high computational effort for the test cases. Therefore another approach is adopted for the meshing.

In a second step, the mesh which is used for all single fiber membrane module test cases is created with the commercial software GAMBIT 2.4.6. The mesh has 10 cells along the radius of the retentate domain and 10 cells along the radius of the permeate domain. The permeate cross section has 560 cells and the retentate cross section has 404 cells. Along the fiber, the domain has 200 cells with a grading on both ends of 1.03. The overall cell count is 192 800 cells with 80 800 cells in the retentate domain and 112 000 cells in the permeate domain. The mesh quality is examined with *checkMesh*. For both domains, no errors are detected.

A cut along the length of the fiber and a cut of the cross section is depicted in figure 17.

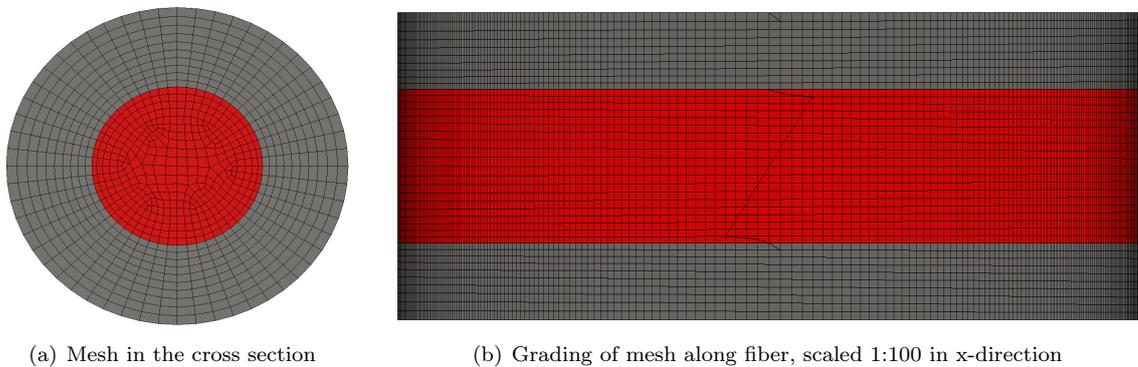


Figure 17: Mesh of the single fiber membrane

4.3.3 Simulation settings in OpenFOAM

The following table 7 shows the standard values, which are applied to all cases. The temperature in all cases is 303.14 K. By default, the pressure is 10 bar on the feed side and 1 bar on the permeate side. This results in a default transmembrane pressure of 9 bar.

Table 7: Standard values for single fiber membrane

Species	mole fraction [mol/mol]	mass fraction [kg/kg]	Permeance [m ³ (STP)/(m ² s bar)]
CO ₂	0.3033	0.531585	2.396×10^{-5}
N ₂	0.3984	0.444467	1.3×10^{-6}
H ₂	0.2983	0.023948	5.864×10^{-5}

Similar to the hollow fiber membrane module with seven fibers, also in this case with one fiber, the flow regime is analyzed as described in chapter 2.7. The highest Reynolds number of around 2360 results from case 5 which has the highest velocity of 11.37 m/s.

The Reynolds number is higher than the critical value of 2300. Though it is considered that the flow regime is laminar and no turbulence model is used.

All cases are set to run for 10 000 iterations. Only three cases are run for a shorter time as the convergence criteria of cumulative species residual of 1×10^{-6} is reached. The cases needed 1600, 1900 and 2800 iterations for the feed velocities of 0.5, 0.25 and 0.125 m/s.

Table 8 shows the changes to the standard values from table 7. The flow pattern, the velocity on the inlet, the feed pressure, the permeances and the courant number are varied.

Table 8: Testcases single fiber

Flow pattern	Velocity [m/s]	retentate pressure p [bar]	Permeances of CO ₂ , N ₂ , H ₂ $\left[\frac{m^3(STP)}{Pa \cdot s \cdot m^2}\right]$	Co nr.
Co-current	3.79	-	-	-
Co-current	5.68	-	-	-
Co-current	7.58	-	-	-
Co-current	9.47	-	-	-
Co-current	11.37	-	-	-
Counter-current	3.79	-	-	-
Counter-current	5.68	-	-	-
Counter-current	7.58	-	-	-
Counter-current	9.47	-	-	-
Counter-current	11.37	-	-	-
Counter-current	3.79	-	$2.396 \times 10^{-9}, 1.3 \times 10^{-10}, 5.864 \times 10^{-9}$	-
Counter-current	5.68	-	$2.396 \times 10^{-9}, 1.3 \times 10^{-10}, 5.864 \times 10^{-9}$	-
Counter-current	7.58	-	$2.396 \times 10^{-9}, 1.3 \times 10^{-10}, 5.864 \times 10^{-9}$	-
Counter-current	9.47	-	$2.396 \times 10^{-9}, 1.3 \times 10^{-10}, 5.864 \times 10^{-9}$	-
Counter-current	11.37	-	$2.396 \times 10^{-9}, 1.3 \times 10^{-10}, 5.864 \times 10^{-9}$	-
Counter-current	5.68	-	-	0.1
Counter-current	5.68	-	-	0.5
Counter-current	5.68	-	-	1
Counter-current	5.68	-	-	2
Counter-current	5.68	-	-	5
Co-current	5.68	2	-	-
Co-current	5.68	4	-	-
Co-current	5.68	6	-	-
Co-current	5.68	8	-	-
Co-current	5.68	16	-	-
Counter-current	5.68	2	-	-
Counter-current	5.68	4	-	-
Counter-current	5.68	6	-	-
Counter-current	5.68	8	-	-
Counter-current	5.68	16	-	-
Counter-current	3.79	-	$4.792 \times 10^{-10}, 2.6 \times 10^{-11}, 1.1728 \times 10^{-9}$	-
Counter-current	3.79	-	$9.584 \times 10^{-10}, 5.2 \times 10^{-11}, 2.3456 \times 10^{-9}$	-
Counter-current	3.79	-	$1.9168 \times 10^{-9}, 1.04 \times 10^{-10}, 4.6912 \times 10^{-9}$	-
Counter-current	3.79	-	$3.8336 \times 10^{-9}, 2.08 \times 10^{-10}, 9.3824 \times 10^{-9}$	-
Counter-current	3.79	-	$7.6672 \times 10^{-9}, 4.16 \times 10^{-10}, 1.87648 \times 10^{-8}$	-
Counter-current	3.79	-	$2.396 \times 10^{-10}, 1.3 \times 10^{-11}, 1.1728 \times 10^{-9}$	-
Counter-current	3.79	-	$2.396 \times 10^{-10}, 1.3 \times 10^{-11}, 2.3456 \times 10^{-9}$	-
Counter current	3.79	-	$2.396 \times 10^{-10}, 1.3 \times 10^{-11}, 4.6912 \times 10^{-9}$	-
Counter-current	3.79	-	$2.396 \times 10^{-10}, 1.3 \times 10^{-11}, 9.3824 \times 10^{-9}$	-
Counter-current	3.79	-	$2.396 \times 10^{-10}, 1.3 \times 10^{-11}, 1.87648 \times 10^{-8}$	-
Counter-current	1	-	-	-
Counter-current	0.5	-	-	-
Counter-current	0.25	-	-	-
Counter-current	0.125	-	-	-

4.4 Hollow fiber membrane with 30 fibers

In a first step, the 30 fiber membrane module was measured in order to get the physical dimensions for the computer model.

In a second step, the model geometry was created with CAD software and meshed.

In a third step, the case was set up with the software OpenFOAM with the available laboratory data as boundary values. The results were then compared to the lab experiments.

4.4.1 Membrane properties

The hollow fibers used for the experimental validation in the 30 fiber module are made of Polydimethylsiloxane (PDMS), an elastomere, according to the data sheet which can be found in the appendix. In [Mulder, 1996] the glass transition temperature of PDMS is -123°C . In general, elastomers have high permeabilities, in contrast to glassy polymeres which have low permeabilities whereas there are some exceptions. The following table shows a sample of permeabilities that hold for PDMS membranes measured at 25°C .

Table 9: Permeabilities of Polydimethylsiloxane [Robb, 1968]

Gas name	Formula	Permeability Coefficient [Barrer]	Permeability Coefficient [SI]
Nitrogen	N_2	280	2.1×10^{-15}
Carbon Monoxide	CO	340	2.55×10^{-15}
Oxygen	O_2	600	4.5×10^{-15}
Hydrogen	H_2	650	4.88×10^{-15}
Methane	CH_4	950	7.13×10^{-15}
Ethane	C_2H_6	2500	1.88×10^{-14}
Carbon Dioxide	CO_2	3250	2.44×10^{-14}
Water	H_2O	36000	2.7×10^{-13}

Units in Barrer are converted to SI units according to the definitions in 2.11.

As the PDMS membrane has a high permeability for carbon dioxide, the goal is to separate the carbon dioxide from the mixture as effectively as possible.

4.4.2 Dimensions of the test module

The membrane module has the following dimensions according to the data sheet.

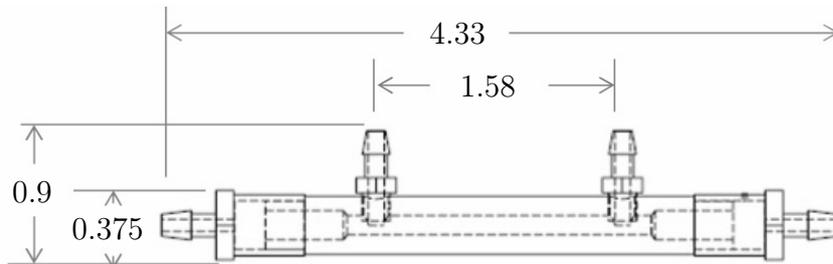


Figure 18: Membrane module as depicted in the data sheet

The following table 10 shows the dimensions of the membrane module which are measured

and the values which are taken from the data sheet in case the property could not be measured.

Table 10: Dimensions of the PDMS membrane module

Dimension	Measured value [mm]	Value from datasheet [mm]
Total length	110.5	109.982
Total height incl outlet	24	22.86
Outer diameter shell	9.5	9.525
Inner diameter shell	40	40.132
Fiber length between potting	45.5	-
Fiber inner diameter	-	0.190
Fiber outer diameter	-	0.300
Fiber wall thickness	-	0.055
Distance between the center of the outlets	40	40.132
Inner diameter inlet/outlet ports	2	-

In table 10, the measured values differ slightly from the values of the datasheet. In order to check the values from the datasheet, the module was measured with a caliper and with scale paper.

The values for the fiber diameter and thickness are taken from the datasheet and not verified with measurements.

The length of the fibers between the pottings was measured to be 45.5 mm. Together with the outer fiber diameter of $300 \mu\text{m}$ this gives a calculated overall surface area of 12.86 cm^2 . For comparison, the datasheet gives an effective membrane area of at least 10 cm^2 . The surface area, calculated with the inner fiber diameter is 8.1477 cm^2 . Thus, the ratio of outer to inner diameter is 1.5789. This has a considerable influence on the modeling approach of the module. The *LTSMembraneFoam* solver is not capable of calculating mass transport through porous layers and thus, the membrane wall is modeled with the inner diameter with zero wall thickness.

Figure 19 shows two photographs of the membrane module. It can be seen that the wall represents a considerable proportion of the module diameter of 9.5 mm. For the simulation, only the inner module diameter was modeled, which was measured to be 4 mm.



(a) Module side view



(b) Module top view

Figure 19: Actual membrane module

The following table 11 shows typical operating conditions of the membrane module according to the data sheet. The maximum module pressures for both domains, fibers and shell, and the maximum transmembrane pressure differences are given at a temperature of $25 \text{ }^\circ\text{C}$

Table 11: Operating conditions

Physical property	Unit	Range
Maximum continuous operating temperature	°C	60
Maximum shell side pressure	bar	3
Maximum lumen pressure	bar	5
Maximum Δp shell to lumen	bar	1
Maximum Δp lumen to shell	bar	3
Typical liquid flow rate	l/min	0.001-0.01
Typical gas flow rate	Nl/min	0.001-0.1

With the given maximum flow rate of 0.1 m/s and the inner diameter of 0.19 mm the flow regime can be analyzed according to equation 2.20. With a density of 7.2 kg/m³ and a dynamic viscosity η of ambient air at 4 bar absolute pressure, this results in a Reynolds number of around 7. Thus, the flow regime is considered to be laminar.

In figure 20 the technical drawing for meshing the module is depicted in side view and sectional view. For modeling, only the inner diameter of the inlets and outlets are considered. Figure 20 (a) shows that the potting is not evenly distributed on both sides. On the left side the potting reaches 17 mm into the shell, almost into the outlet. On the right side it reaches only 12 mm into the shell. The bolt circle has a radius of 0.4 mm for the inner fibers, 0.9 mm for the middle fibers and 1.3 mm for the outer fibers.

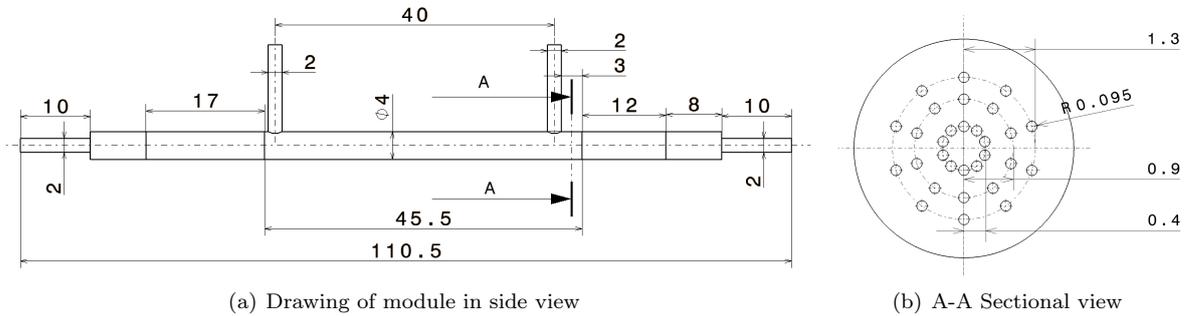


Figure 20: Technical drawing of the module (units in mm)

4.4.3 Meshing with snappyHexMesh

The mesh was created with the OpenFOAM tool *snappyHexMesh*. In this case, only step 1 and step 2, as elaborated in chapter 4.3.2 were executed as the layering occasionally resulted in a large increase of the cell count. For the geometry data, the CAD software CATIA was used, to create the solid geometry. From this solid geometry, the .stl files which describe the boundary of the regions *concentrateDomain* and *permeateDomain* were extracted. In this section, the labels *concentrateDomain*, *concentrate* or *concentrate zone* are equivalent to *permeateDomain*, *permeate* or *permeate zone*. Generally it is necessary that the .stl files for the *snappyHexMesh* tool encompass a closed geometry. This is especially important for geometries which consist of several different .stl files to create one closed geometry. To ensure that the mesh from the two domains, *concentrateDomain* and *permeateDomain* are compatible and connected, it is necessary that both domains are created from the same surfaces which are shared with both domains. In this case,

the .stl file *fiberwallmembrane.stl* which represents the surface area of the transmembrane flux, is part of both domains.

The .stl files were created for the *concentrateDomain* and for the *permeateDomain*. Those files were combined to one single .stl file for each domain and stored in the folder *triSurface*. In order to set the boundary types and the boundary conditions it was necessary, to store the .stl files for the inlets and outlets explicitly in the folder *trisurface* as well as implicitly in the domain files, as they are part of the combined .stl file.

Figure 21 shows a coarser blockMesh for the hollowfiber membrane module than actually used. For the actual case, a blockMesh with 600 cells in x-direction, 51 cells in y-direction and 120 cells in z-direction was used.

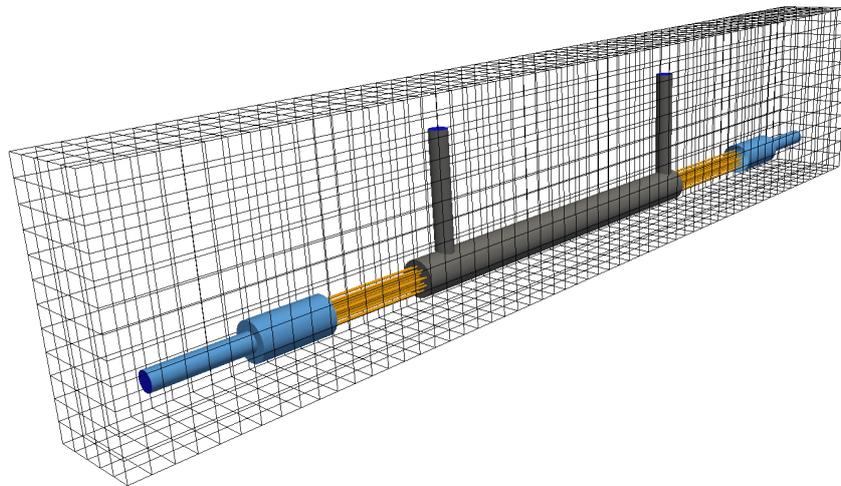
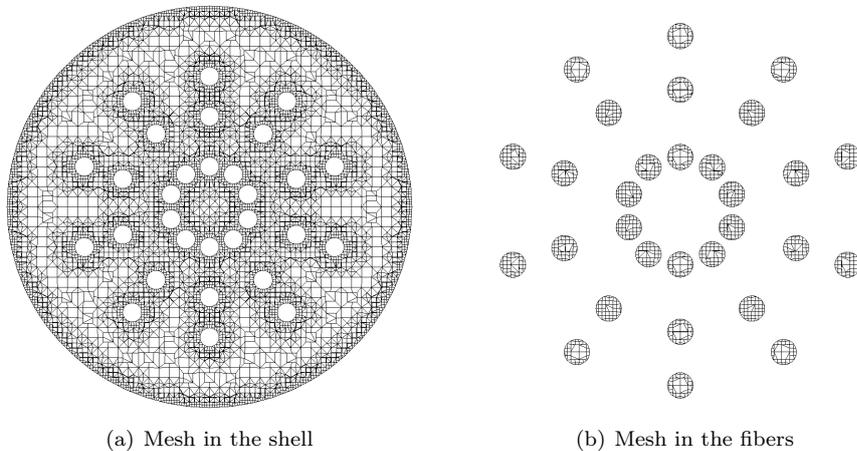


Figure 21: BlockMesh around hollow fiber membrane

The refinement level for both domains was set to 3 which has the consequence, that each cell which is in contact inside the surface is split three times. Its neighboring cell is split twice and the third adjacent cell row is split once. Though it is also possible to define several cells between each refinement level.

The resulting mesh of the *permeateDomain* is depicted in figure 22.



(a) Mesh in the shell

(b) Mesh in the fibers

Figure 22: Mesh in the concentrate and *permeateDomain*

The cell count on the concentrate and *permeateDomain* is around 4.579 and 6.595 million respectively. The utility *checkMesh* gives no errors for the concentrate domain except for the aspect ratio. The maximum aspect ratio is 7 for 164 skew faces on the concentrate Side. On the permeate side, the mesh check as well as the maximum skewness is OK.

4.4.4 Simulation settings in OpenFOAM

With the permeabilities from table 9 and a membrane thickness of $55\ \mu\text{m}$ according to the data sheet which can be found in the appendix, the permeances can be calculated according to equation 2.8.

The unit of the permeance is usually given in $\text{m}^3/(\text{m}^2\ \text{bar}\ \text{s})$ or in $\text{m}^3/(\text{m}^2\ \text{Pa}\ \text{s})$. Depending on where the selective layer is located, the permeance, which is related to the membrane surface area, has to be corrected. In this case, the selective layer is on the outside of the hollow fiber but for the CFD simulation the inner fiber diameter has been modeled as a pipe. Therefore the overall permeance is scaled with the factor 1.5789, which corresponds to the relation between the outer and inner fiber area between concentrate and permeate as it is explained in section 4.4.2.

Prior to mixed gas experiments and simulations, the permeabilities of the pure components hydrogen, (H_2), carbon dioxide (CO_2) and methane (CH_4) are determined. Therefore, the feed-, permeate- and retentate flow of each component is measured.

The input parameters feed velocity, retentate- and permeate outlet pressure and permeances are taken from the laboratory experiments.

The following table 12 shows the pressures on the retentate and permeate side, the feed velocities and the permeances.

Table 12: Input parameters for the simulations

Specie	Pressure retentate [Pa]	Pressure permeate [Pa]	Corrected feed velocity [m/s]	Permeance scaled [$\text{Nm}^3/(\text{m}^2\ \text{Pa}\ \text{s})$]
CO_2	400241	80392	0.58614	5.1406×10^{-10}
CH_4	400866	79493	0.60142	1.1749×10^{-10}
H_2	400349	79715	0.41268	1.1352×10^{-10}

The corrected feed velocity is a result from the correction of the deviation in the mass balance from the lab experiments.

The error in the mass balance from the lab experiments was 1 % for carbon dioxide, 10 % for methane and 15 % for hydrogen. In order to account for that error, it is assumed that 50 % of the mass flow was lost on the feed and 50 % on the retentate side. The mass balance error was corrected whereas the feed flow is reduced by half of the error and the retentate flow was increased by half of the error and thus also the velocity was adjusted accordingly. In chapter 8.3.1 the results from the experiments are explained in more detail.

5 Laboratory measurements

The measurements were carried out by Dr. Martin Miltner in the laboratory of the research division thermal process engineering and simulation from the institute of chemical engineering at TU Wien.

5.1 Feed and volume flow

The feed flow is measured with the digital mass flow controller mini CORI-FLOW[®] from the company Bronkhorst Cori-Tech B.V., Netherlands. The feed flow controller is designed to measure liquids and gases.

The measurement of the mass flow is based on the coriolis principle where a fluid exerts a coriolis force as it flows through a tube. The mass flow controller measures the variations of the twisting and bending of the tube, as a result of the coriolis force, which is proportional to the mass flow. The measurement principle is independent of density, temperature, viscosity, pressure and conductivity of the fluid.

The retentate and the permeate flow are measured with a Definer 220 flow meter from the company Mesa Labs, Inc. The measurement is based on the principle of positive displacement. The flow meter includes temperature sensors and pressure transducers in the flow stream to compensate for standard conditions.

5.2 Pressure

The pressure before and after the membrane module is measured with pressure sensors with the production series P3276 from the company tecsis GMBH, Germany. The feed and retentate pressure sensors have a measurement range of 0 to 100 bar and 0 to 25 bar respectively. Both sensors have an accuracy of 0.5 %.

The sensor for the feed pressure is placed right before the inlet into the membrane module. The retentate pressure sensor is located before the Definer flow meter. For measurement of the volume flow, no excess pressure is allowed in the Definer flow meter.

In order to compare the inlet and outlet flows, the feed flow is converted to a volume flow. The values for the density are taken from the NIST chemical webbook at standard conditions of 273.15 K and 1.013 25 bar [NIST, 2016].

6 Validation

The results of the simulations with the membrane module with five different outlet configurations are validated against a process simulation model with the software Aspen Custom Modeler[®] (ACM V7.3) distributed by Aspen Technology, Inc., Burlington, USA, 2011. The ACM uses an object-oriented modeling language which can be adapted to different models in order to implement them into a process flow sheet in Aspen Plus[®].

The software Aspen Plus[®] V7.3.2 solves the mass and energy balance equations. The physical properties of the flow conditions and of the species are obtained from the Aspen Plus component database. The ACM model is validated against an algorithm and against experiments.

6.1 Aspen Custom Modeler (ACM) model

The ACM gas permeation model which is used for the validation cases is briefly described in this section and further described in [Lassmann, 2015]. The ACM gas permeation model which was created by [Lassmann, 2015] was developed from previous models from [Rodrigues, 2009], [Rom et al., 2014] and [Makaruk and Harasek, 2009].

The model with the ACM is set up as a 1-dimensional case. The whole membrane model geometry is segmented into 100 parts or cells whereas each cell is of equal size. The model as it is implemented in the ACM is depicted in figure 23. It is designed to represent a hollow fiber membrane module with bore side feed flow. The entrance for the feed flow of the species CO_2 , N_2 and H_2 is on the top left side and the flow pattern as it is depicted is co-current. The lower left side, below the membrane can be used to insert sweep gas to the permeate side. In the case used for validation with the CFD membrane module, no sweep gas is used.

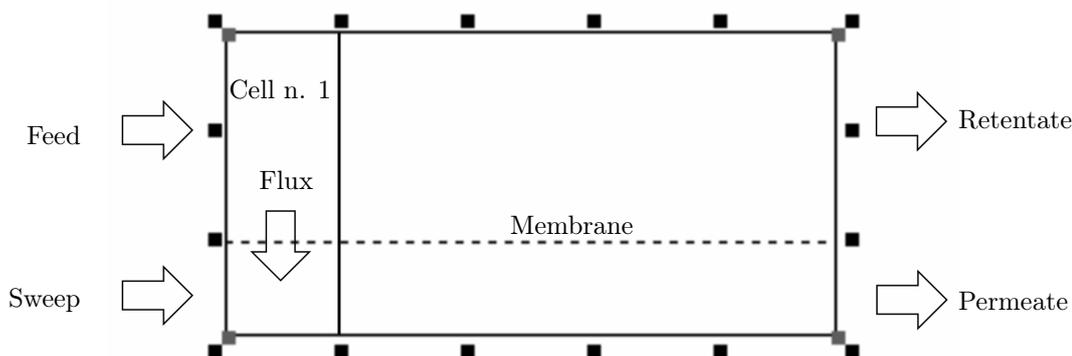


Figure 23: Membrane cell ACM, [Lassmann, 2015]

Figure 24 shows the connection between the cells and a counter-current flow pattern. For both, permeate and retentate, the flow is indicated with the letter n and the composition with y .

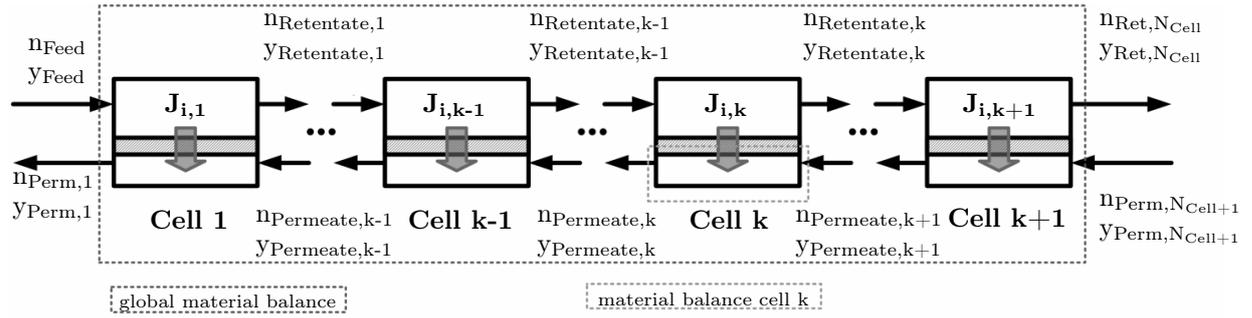


Figure 24: Membrane model, [Lassmann, 2015]

The following list shows the input parameters for the ACM model which was implemented by [Lassmann, 2015].

- Fiber diameter [m]
- Fiber length [m]
- Number of fibers [-]
- Number of cells [-]
- Temperature [°C]
- Feed flow [kmol/h]
- Feed composition [kmol/kmol]
- Feed pressure [bar]
- Permeance of each component [$m^3(STP)/(m^2 s bar)$]

The total membrane area is calculated with the diameter, length and number of the fibers. The temperature across the membrane module is kept at a constant level as it is assumed that the effects from temperature gradients can be ignored.

For every cell of the ACM membrane model, the flux across the membrane is calculated. The trans-membrane flux, J_i is defined as described by equation 2.35, where the driving force is the pressure difference Δp .

In the ACM model, the membrane thickness is modeled with infinite thickness. Therefore, the permeance is set as a given value.

On the permeate side, the pressure drop is assumed to be constant. In the ACM cases, the pressure drop is only calculated for the retentate side. With given values for the volume flux, the dimensions of the membrane and with the help of a database, the pressure difference on the retentate side can be calculated from Hagen Poiseuille's law, described in equation 2.25.

The permeance for each component is kept constant and doesn't change with the partial pressure on feed and permeate side.

6.2 Validation of the ACM model

The ACM model is validated against an algorithm developed by [Makaruk and Harasek, 2009] as well as against experiments. Both validations are carried out with a binary gas mixture of hydrogen and carbon dioxide. The pressure drop along the fibers on the retentate side is neglected for both validations.

6.2.1 Validation ACM model against an algorithm

The algorithm by [Makaruk and Harasek, 2009] is validated against experiments and literature results.

For validation of the ACM model with the algorithm, eleven test cases are set up with variations of the feed composition, the pressure ideal selectivity and the stage cut. The variations are listed in table 13 In all test cases, the number of cells is 100, the feed flow is $1 \text{ N m}^3/\text{s}$, the temperature is 25°C and the permeate pressure is 1 bar.

Table 13: Variations for the validation of ACM model with algorithm, [Lassmann, 2015]

Parameters	Unit	Variation
Feed volume fraction H_2	[vol/vol]	0.2
		0.5
		0.8
		3
Feed pressure	[bar]	4
		7
		10
		3
Stage cut	[vol/vol]	4
		7
		10
		3
Ideal selectivity	[-]	4
		7
		10
		10

The results of simulations with the ACM model with the deviations from the algorithm are listed in table 14. The deviation between both models is less than one percent for the most cases, except for the membrane area with 1.7%. The deviations are assumed to be a result of the conversion from molar flow to volume flow at standard conditions. As an input parameters to calculate the transmembrane flux, the algorithm uses molar fractions and the ACM model volume fractions.

Table 14: Comparison ACM model and algorithm, [Lassmann, 2015]

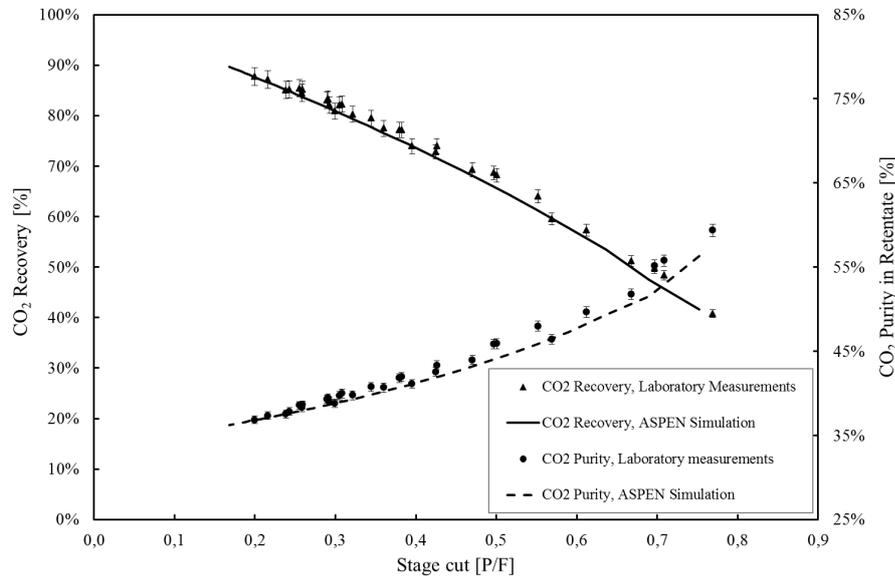
Validation parameters	Unit	Deviation lower bound [%]	Deviation upper bound [%]
Permeate volume flow	$[\text{N m}^3/\text{s}]$	-0.0024	0.0001
Permeate H_2 volume fraction	[vol/vol]	0.0001	0.009
Retentate volume flow	$[\text{N m}^3/\text{s}]$	-0.0001	0.0011
Retentate H_2 volume fraction	[vol/vol]	-0.0018	-0.0006
Membrane area	$[\text{m}^2]$	0.0100	0.017

6.2.2 Validation ACM model against lab experiments

The validation of the ACM model with the laboratory experiments, was carried out using an in-house built hollow fiber membrane module. The purity of hydrogen and carbon dioxide in the permeate stream was recorded for different pressure levels and stage cuts. For the validation a binary mixture was used with a volume fraction of 0.66 hydrogen and 0.34 carbon dioxide.

In the simulation for the validation with experiments, the temperature across the membrane was assumed to be constant with 303.14 K. The graphs with the validation data show hydrogen recovery and carbon dioxide recovery over a stage cut range from 0.18 to 0.76. The model predicts the experimental results within the error range of the experiments. It can be reasonably assumed that the elaborated ACM model can be used to simulate membrane gas separation processes.

The results for the validation of the ACM simulations and the experiments are shown in figures 25 and 26.



(a) ACM validation H₂

Figure 25: ACM validation carbon dioxide, [Lassmann, 2015]

It is assumed that the gas cools down on the permeate side as it is expanding as a result of lower pressure. In the lab experiment, the membrane is heated and therefore kept at a constant temperature of 303.14 K. Thus, repercussions of a lower permeability as a result of lower temperature on the permeate side of the membrane are omitted. In the ACM model which is used to model transmembrane fluxes, no energy balance was considered.

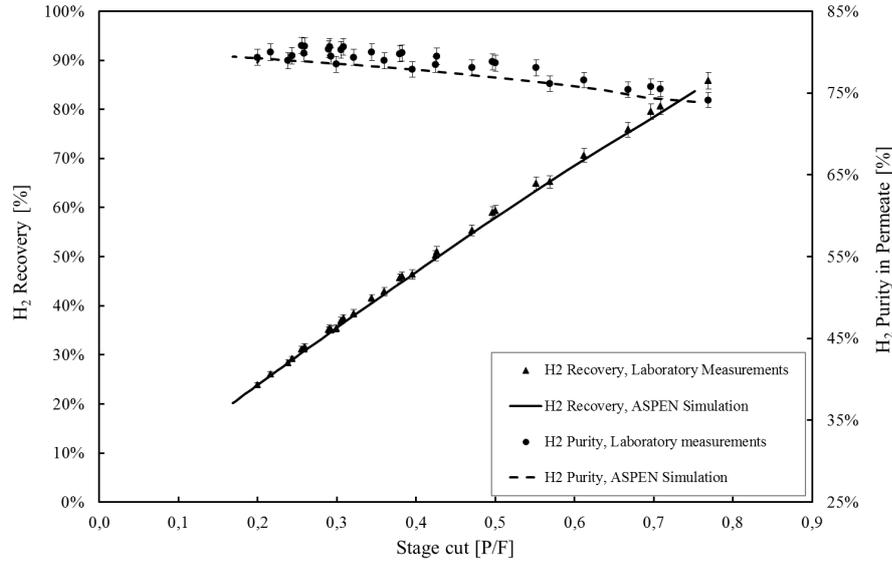
(a) ACM validation CO₂

Figure 26: ACM validation hydrogen, [Lassmann, 2015]

6.3 Validation CFD solver LTSMembraneFoam against ACM model

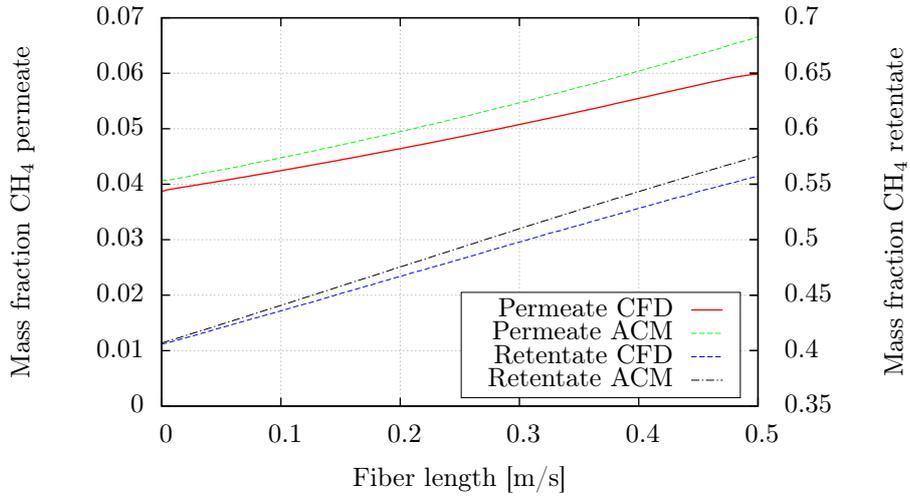
For validating the CFD solver, five cases are analyzed with different feed velocities and compared with the ACM model.

Figure 27 shows the mass fraction distribution for the fully counter current case for a feed velocity of 0.1 m/s which corresponds to a mass flow of 6.21×10^{-6} kg/s. The figure shows the mass fraction distribution for the species methane and carbon dioxide for both domains, retentate and permeate, along the module.

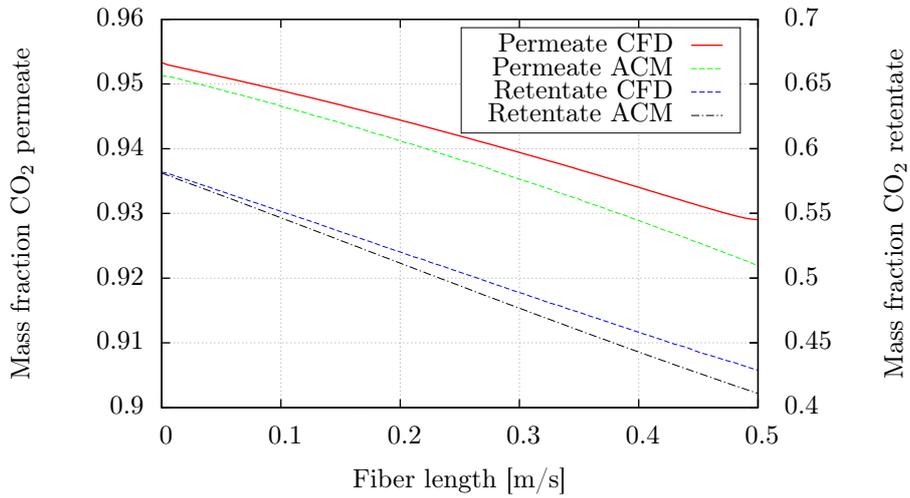
For the CFD model, the concentration of the higher permeating component, CO₂ has a higher mass fraction on the permeate side as well as on the retentate side compared to the ACM model. In contrast to CO₂, the CFD model shows a lower mass fraction in both domains for the lower permeating components CH₄ and O₂ compared to the ACM model. The ACM model predicts a lower mass fraction of the component with the higher permeance.

The pressure drop in the ACM case is only calculated for the retentate side. Figure 27(c) shows a pressure drop of 2 Pa along the shell without considering the pressure drop resulting from the small outlet tubes. The outlet tubes result in a pressure drop of around 4.3 Pa. Thus the overall shell side pressure drop, calculated from the CFD model, is 6.3 Pa.

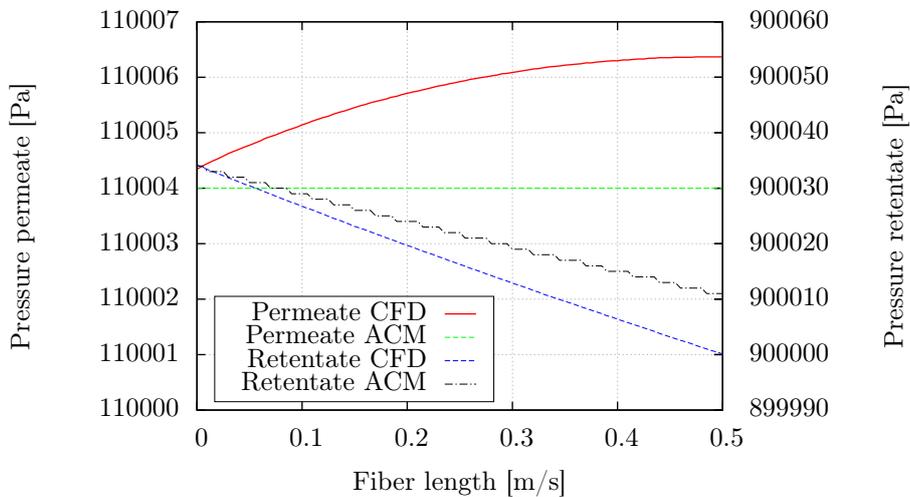
The CFD model shows an overall pressure drop of 34 Pa on the retentate side. This value is taken as an inlet boundary condition for the ACM model. In contrast to the ACM model the boundary condition for the CFD model is set at the outlet. The ACM model calculates a total pressure drop on the retentate side of 23 Pa which is 11 Pa lower than the result from the CFD model. The ACM pressure drop shows a stepwise trend on the retentate side. This can be attributed to the fact that for data extraction no decimal places are considered and the difference between neighboring cells is partially smaller than integer values.



(a) Mass fraction of CH₄



(b) Mass fraction of CO₂



(c) Pressure

Figure 27: Comparison CFD and ACM

The following figure 28 shows the comparison between the ACM simulations and the CFD simulations for different feed velocities and stage cuts.

Each stage cut belongs to a certain feed velocity, specified in table 5. Depending on the transmembrane flux, also the stage cut can differ in both models.

Stage cut

Unlike previously defined in equation 2.15, the stage cut in this case is given with the ratio of permeate mass flow to feed mass flow. For each feed velocity and for both, the permeate and the retentate domain, the CFD model predicts a lower stage cut than the ACM model for all species.

Permeate and retentate composition

For the CFD simulation, only the results for methane and carbon dioxide are interpreted. The results of oxygen are considered not to be reliable as the error in the mass balance of the specie oxygen is too high. For example for the velocity 0.01 m/s the error is in the range of 2 %, even after 50 000 iterations. This can be attributed to the low mass fraction of oxygen in the feed.

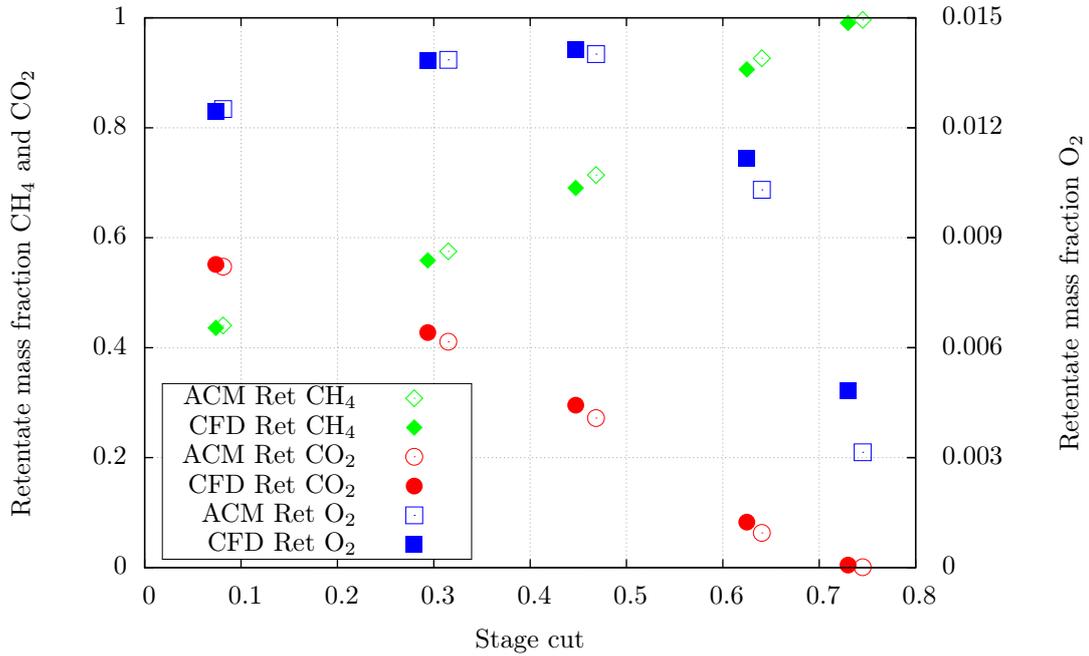
On the retentate domain in figure 28(a), the following can be observed for the specie composition.

- For carbon dioxide the CFD model shows a higher mass fraction than the ACM model.
- For methane the CFD model shows a lower mass fraction than the ACM model.

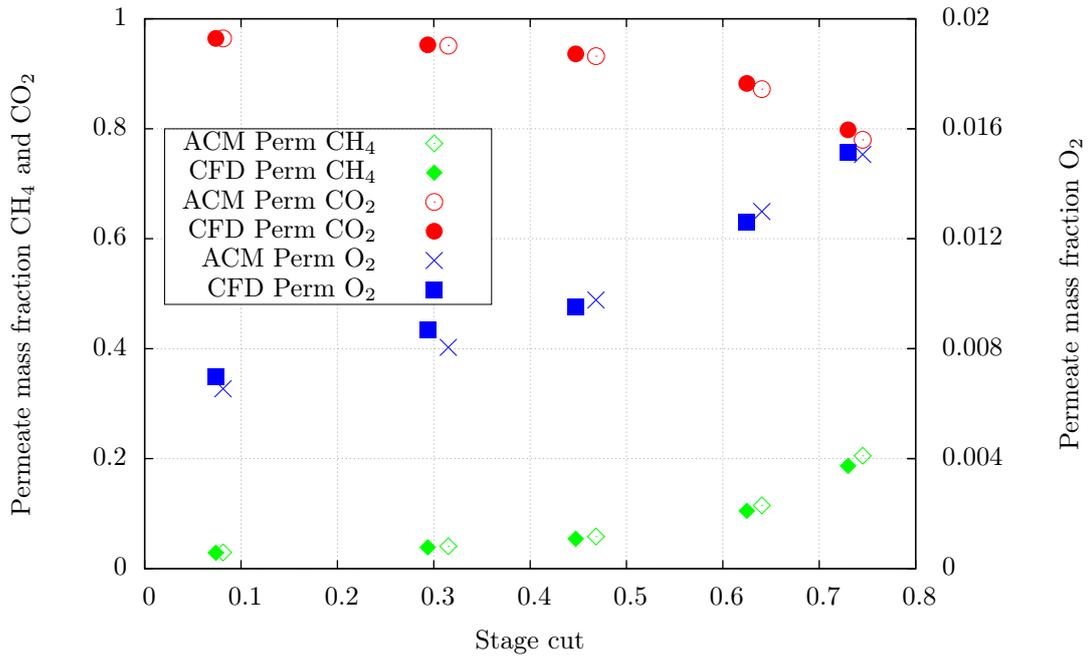
On the permeate domain in figure 28(b), the following can be observed for the specie composition.

- For carbon dioxide the CFD model shows a higher mass fraction than the ACM model.
- For methane the CFD model shows a lower mass fraction than the ACM model.

Generally, in both, the permeate and the retentate domain, the difference in the prediction of the specie mass fractions between ACM and CFD is larger for lower feed flows. Thus higher stage cuts result in a higher difference of the permeate and retentate specie composition between both models. The difference is considerably high on the retentate side for the highest stage cut. The difference falls below 1 % for the lowest stage cut. For the CFD simulations, for the smallest feed flow, the difference can be attributed to the mass balance error.



(a) Retentate composition



(b) Permeate composition

Figure 28: Comparison of composition at different stage cuts for CFD and ACM, [Haddadi et al., 2015a], [Haddadi et al., 2015b]

Retentate and permeate mass flow

Generally, the difference in the prediction of the retentate mass flow, between ACM and CFD is greater for lower feed flows which corresponds to a higher stage cut. A higher stage cut results in a larger difference of the retentate mass flow. A higher stage cut also implies an absolute lower permeate mass flow. Only the relative permeate mass flow,

which is related to the feed flow, is higher at lower stage cuts.

The difference in the prediction of the permeate mass flow, between ACM and CFD is larger for higher feed flows. A lower stage cut results in a higher difference of the permeate mass flow.

Table 15 shows the percentage mass flow error between the simulations of ACM and CFD model.

Table 15: Error for validation CFD and ACM of mass flow in [%]

Feed velocity	Mass stage cut	Feed mass flow			Permeate mass flow			Retentate mass flow		
		CH ₄	O ₂	CO ₂	CH ₄	O ₂	CO ₂	CH ₄	O ₂	CO ₂
0.01	0.73	0.18	0.18	0.17	11.4	1.6	0.3	5.7	48.0	150.1
0.02	0.62	0.18	0.18	0.17	11.5	5.4	1.2	2.0	32.9	34.2
0.05	0.45	0.18	0.18	0.17	11.6	7.1	4.1	0.6	4.7	12.1
0.1	0.29	0.17	0.18	0.17	11.8	0.6	6.9	0.04	2.9	6.9
0.5	0.07	0.16	0.16	0.19	11.0	3.1	9.7	0.1	0.3	1.6

Figures 29 and 30 show the mass flow at different stage cuts for the ACM and the CFD model. Two observations can be made.

- On the permeate side the mass flow of carbon dioxide and methane is lower for the CFD model than for the ACM model.
- On the retentate side the mass flow of carbon dioxide and methane is higher for the CFD model than for the ACM model.

The interpretation of the oxygen mass flow is omitted because the mass flow is too low for the error range.

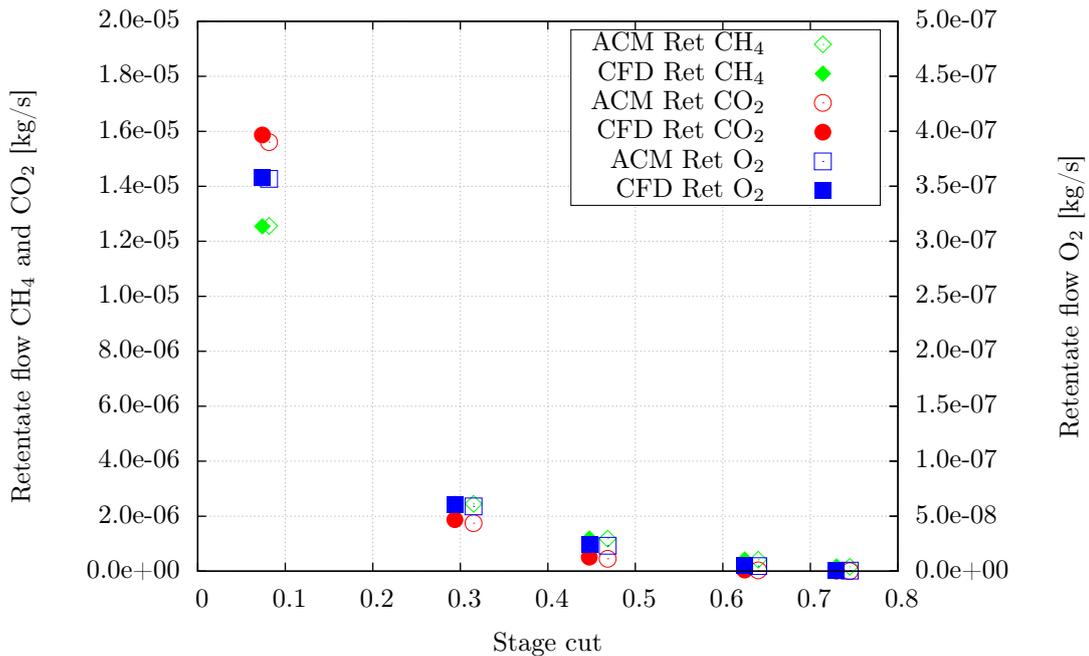


Figure 29: Comparison of retentate mass flow at different stage cuts for CFD and ACM

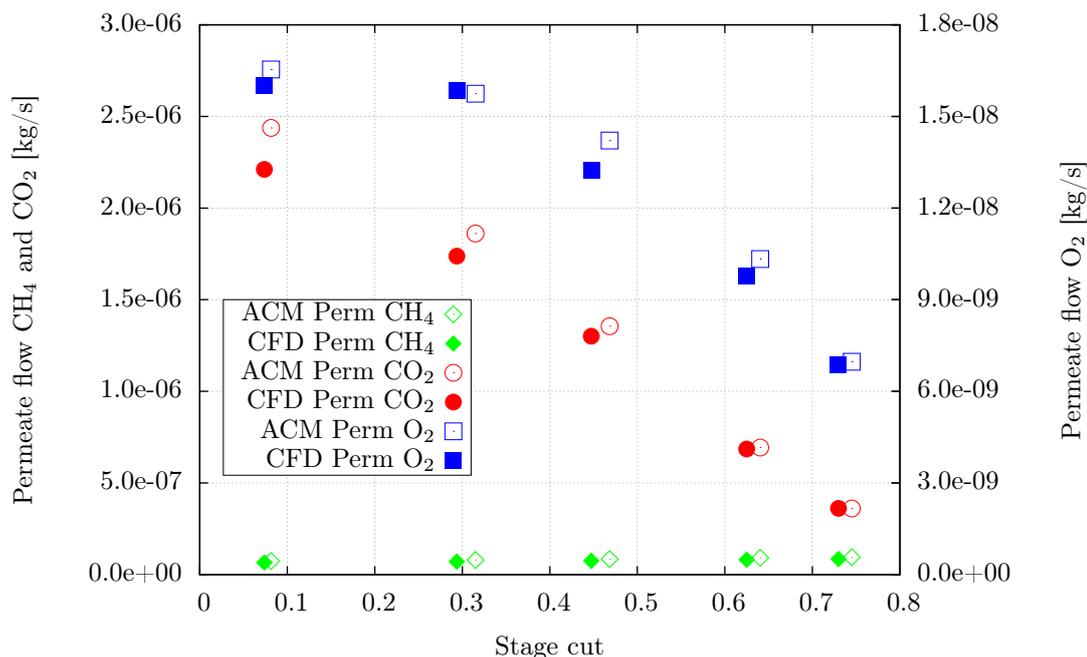


Figure 30: Comparison of permeate mass flow at different stage cuts for CFD and ACM

6.3.1 Deviation between ACM and OpenFOAM

The differences between the ACM and the CFD model can be summarized in the following five points. The first three points represent limited capabilities of the modeling approach of the ACM model.

1. Concentration polarization
2. Pressure drop on the permeate side
3. One dimensional modeling
4. Conversion mass fraction to mole fraction
5. Different reference conditions

Concentration polarization

In the CFD simulation it is possible to see concentration polarization. This can not be visualized with the ACM model which uses a 1-dimensional model and the results do not show any visual output.

Also concentration polarization cannot be calculated in the ACM model because the whole retentate and permeate domain normal to the membrane surface is modeled with one cell. The whole membrane domain in the ACM model is subdivided along the length of the module, parallel to its surface as depicted in figures 23 and 24. The model therefore can't calculate a concentration gradient of the species normal to the membrane surface.

The effect of concentration polarization is visualized in the following figure 31 which is taken from the central fiber from the middle of the seven fiber membrane module. The mesh is especially created in order to visualize possible occurrence of concentration polarization inside the fibers. Therefore, ten boundary layers are created with a thickness

of the smallest cell on the boundary of 10×10^{-5} m and a growth factor of 1.15 towards the center of the fiber. Each fiber has 91 438 cells in total and 262 cells in the cross section. The whole refined membrane module has 3 186 107 cells. Despite the higher cell count, the accuracy of the results which are explained above is not influenced significantly. Though it is expected that the results with the higher cell count and with the graded mesh are more accurate. The following table compares a simulation with fully co-current flow pattern and for a feed velocity of 0.1 m/s with the results from the coarser mesh.

Table 16: Comparison coarse mesh and graded mesh

	Mass flow CO ₂ retentate [kg/s]	Mass flow CO ₂ permeate [kg/s]	Mass flow CH ₄ retentate [kg/s]	Mass flow CH ₄ permeate [kg/s]
Coarse mesh	1.9113×10^{-6}	1.7045×10^{-6}	2.4505×10^{-6}	7.1465×10^{-8}
Graded mesh	1.9127×10^{-6}	1.7033×10^{-6}	2.4499×10^{-6}	7.1445×10^{-8}
Deviation [%]	6.4×10^{-2}	7.1×10^{-2}	1.8×10^{-2}	3.7×10^{-2}

All other results shown in this section 8.1.2 are based on calculations with the coarse mesh as described in section 4.1.2.

The center of the fiber shows a higher concentration of carbon dioxide (a) and a smaller concentration of methane (b). Towards the fiber wall, the mass fraction of carbon dioxide decreases whereas methane increases.

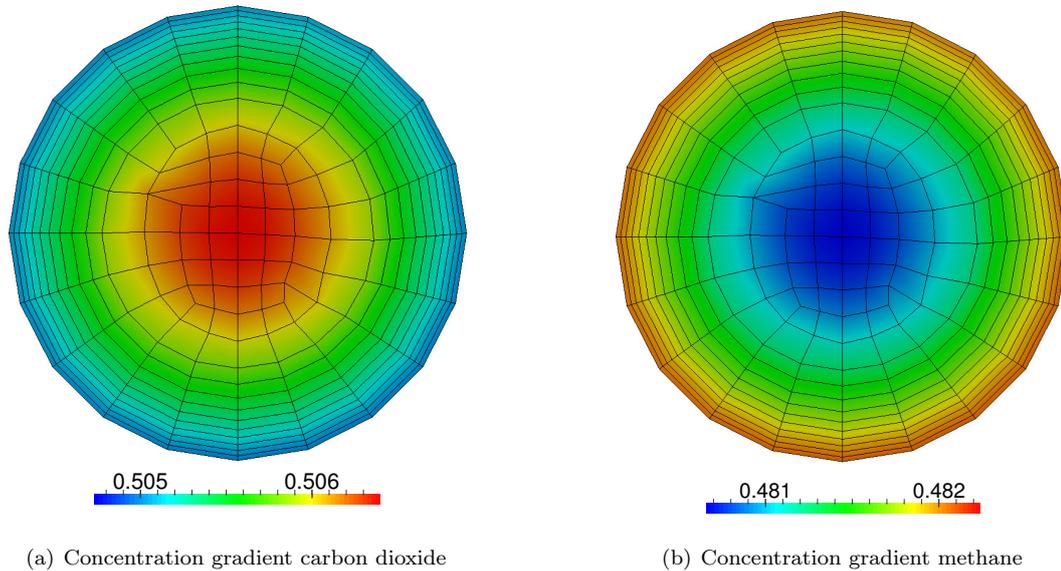


Figure 31: Concentration gradient in the central fiber

The following graph in figure 32 shows the course of the concentration in the center fiber along the diameter. According to equation 2.16 the concentration of a component with higher permeance increases exponentially towards the membrane surface near the membrane wall. In this case, in contrast, the mass fraction of methane increases in the whole cross section. From the center of the fiber to the membrane wall, the concentration only increases by 0.3 %. Considering a mass balance error of around 0.1 %, the influence of concentration polarization is very low.

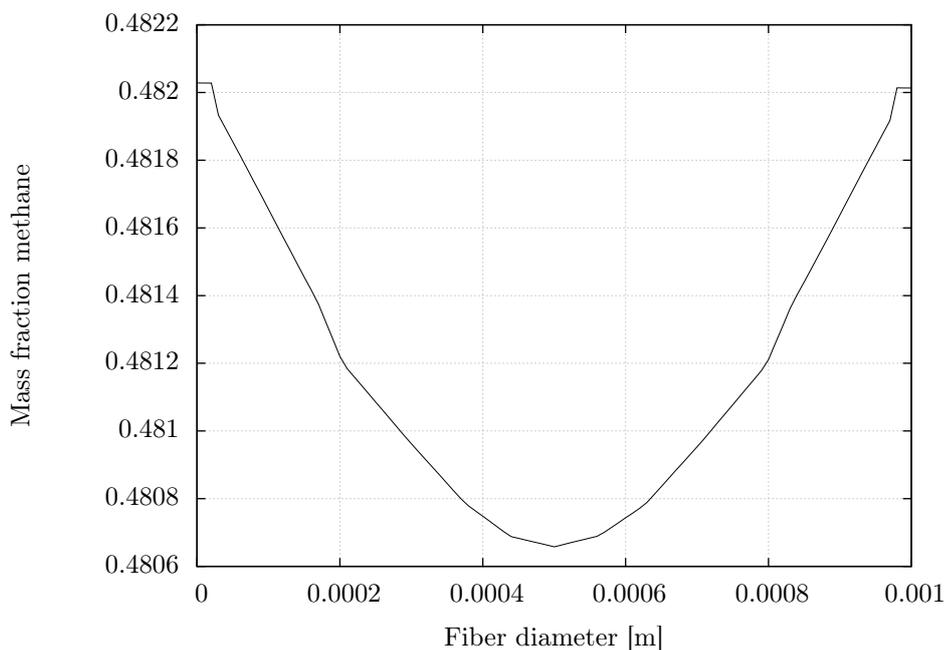


Figure 32: Concentration of methane in the cross section of the fiber

Pressure drop

The assumption of the ACM model of zero pressure drop on the permeate side is quite close to the results from the CFD simulation. The assumption is based on the velocity profile, which is close to zero, along the permeate zone. As it is difficult to predict the flux behavior on the permeate side of the membrane, modeling the pressure drop in the ACM model in the respective zone is omitted.

Modeling

The 1-d modeling approach makes the ACM model considerably simplified compared to the CFD 3-dimensional model. The ACM model models the concentration gradient only parallel to the membrane surface. The CFD model in contrast also considers a concentration gradient normal to the flow direction.

Conversion mass fraction

The input data for the feed flow for the ACM model is given in mole fractions. In the ACM model all fluxes, including the transmembrane flux, are calculated in the unit [kmol/hour]. For the CFD model the feed flow is given in mass fractions. Thus, the mass fractions from the CFD model are converted to mole fractions for the ACM model. The input parameters are set according to the values used in [Makaruk and Harasek, 2009], where mole fractions are used. The following table shows the input parameters for CFD and for ACM and the difference in the mass fractions.

Table 17: Input parameters ACM and CFD

Specie	Mass fraction CFD	Mass fraction ACM	Mole fraction ACM	Difference [%]
CH ₄	0.406	0.406809	0.651221	0.20
CO ₂	0.582	0.581167	0.339129	0.20
O ₂	0.012	0.012024	0.009650	0.14

Table 17 shows that even the feed composition has a deviation of 0.14 to 0.2 %. This results in a high deviation in permeate and retentate composition especially at low velocities. The deviation is very high for carbon dioxide as all of the species CO₂ is permeating through the membrane at low stage cuts and the remaining CO₂ in the retentate is 4 orders of magnitude lower than in the permeate. This is, especially for low velocities and for high transmembrane fluxes, in the order of the error range. On the other side, the deviation for the components in the permeate is small as the mass fractions are higher.

Different reference conditions

The difference in the reference conditions has implications on the permeances in both, the CFD and the ACM model. In the CFD model, the standard conditions are set at a pressure of 1×10^5 Pa and a temperature of 298.15 K. Thus the permeance is corrected as explained in section 3.6.2. In the ACM model, the standard conditions are set at 1.01325×10^5 Pa and a temperature of 273.15 K.

7 Convergence

In the following the convergence study for the physical and numerical errors is described exemplarily for case 5. Though, both error criteria are checked for each case.

7.1 Numerical errors

In order to check for the convergency of the solution, a threshold value for the specie residuals is set in the solver `membraneFoam`. The sum of the specie residuals, the cumulative specie residuals, is defined as follows

$$\sum_{i=1}^k \sum_{j=1}^n |Y_j^t - Y_j^{t-1}| \quad (7.1)$$

where n is the number of species, Y_j the value of a specific specie j and k stands for the total number of cells. The convergency criteria depends on the number of specie and on the total number of cells and thus the convergency criteria must be set individually for each case. Also, in general, the cumulative specie residual differ on the permeate and on the feed side. It is considered that a residual of 10^{-5} per species per cell is sufficiently low.

Figure 33 shows the residuals of CO_2 , CH_4 , enthalpy, pressure and density of case 5. The case is exemplary for the other cases which show almost identical residual plots. It can be seen that after around 3000 iterations the residuals of density, species and pressure reach a constant value. The enthalpy residuals reach a constant level of around 10^{-5} after around 5000 iterations. Each of the analyzed cases are run for at least 7000 iterations.

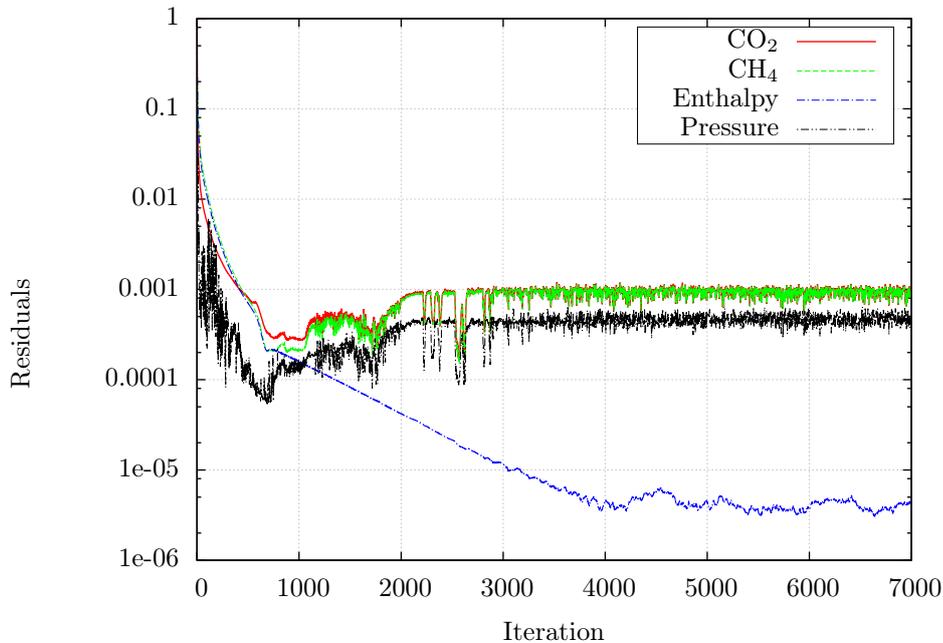
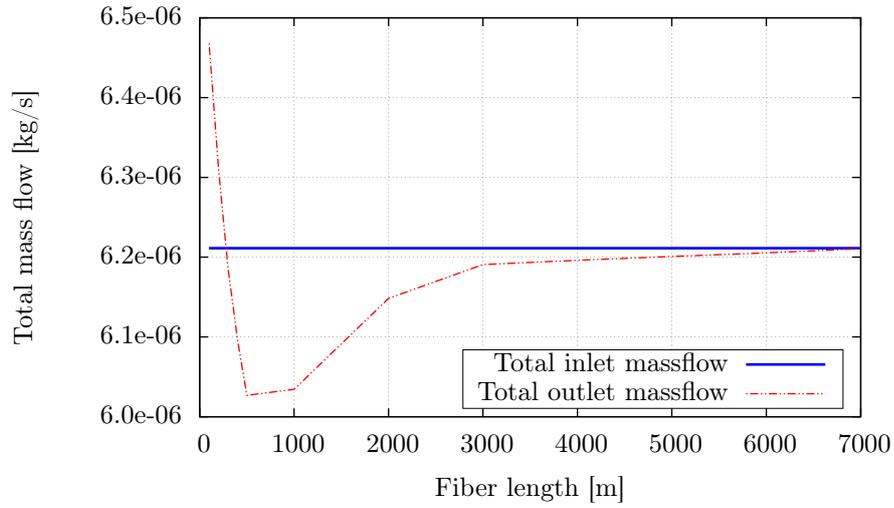


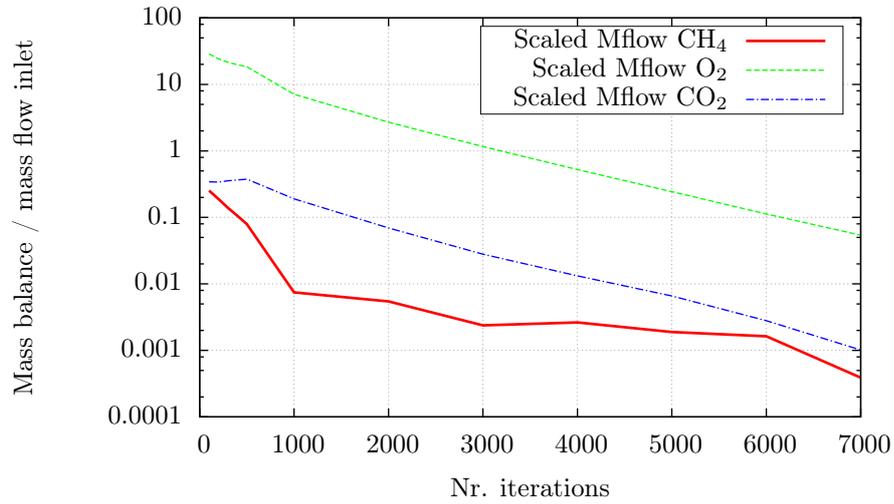
Figure 33: Course of residuals

7.2 Physical errors

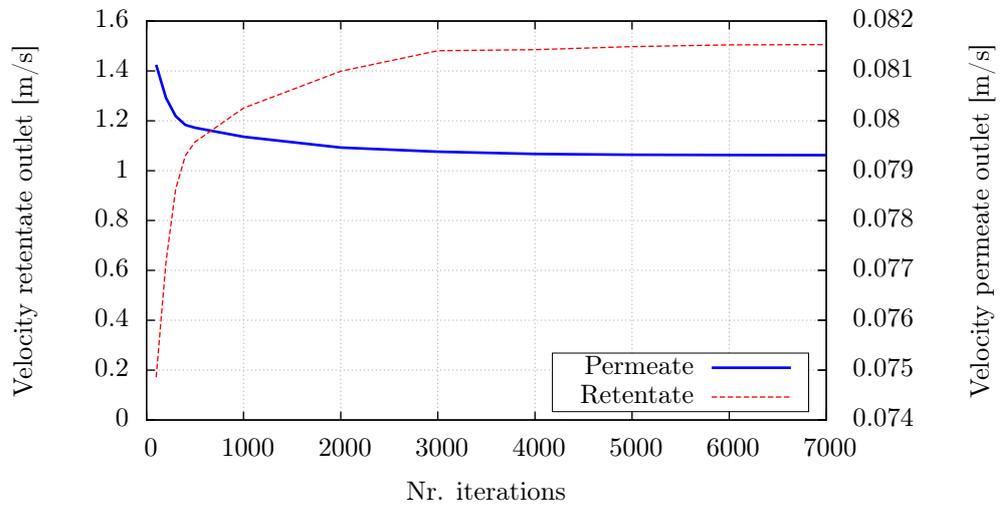
Figure 34 graphically illustrates the convergence of the solution over the iterations. Though, the convergence of the mass balance can also be checked with the mass balance error of the inlet and outlet streams as listed in table 20. Figure 34 (a) shows the total mass balance of all species. At 7000 iterations the mass balance between inlet and the outlets of permeate and retentate is closed. It is assumed that when the mass balance error reaches a level lower than 0.1 %, the solution is converged. Figure 34 (b) shows the scaled mass flow balance and (c) the velocities at the inlet and at the outlets of permeate and retentate. The scaled balance is defined as the absolute difference between the inlet and the outlet mass flow divided by the inlet mass flow. The balance errors are scaled with the mass flow in order to show them in the same error range. Oxygen shows the highest mass balance error as it has the lowest mass fraction in the feed. Although the scaled balances seem to continue decreasing, it is assumed that the overall error reached an acceptable range.



(a) Total mass balance



(b) Scaled absolute mass balance difference



(c) Velocities at inlet and outlet

Figure 34: Course of scaled mass balance difference and velocities, [Haddadi et al., 2015a], [Haddadi et al., 2015b]

7.3 Convergence of CFD simulations with different feed velocities

The CFD simulations with OpenFOAM show a faster convergence for the higher velocities 0.1 and 0.5 compared to the lower velocities 0.01, 0.02 and 0.05 m/s. A fewer number of iterations are necessary for the cases with higher velocities in order to reach a comparable convergence level. The convergence was checked with the same numerical and physical parameters as explained in chapter 7.

Whereas for the highest velocity, 0.5 m/s, only 6000 iterations are necessary in order to reach an overall mass balance error below 0.1 %, more than 50 000 iterations are necessary for the lowest velocity, 0.01 m/s to reach a comparable value. For example after 20 000 iterations, the overall mass balance error for the lowest velocity is still 2.2 % and the O₂ mass balance error is 137 %.

Table 18 shows the number of iterations necessary to reach a convergent solution. Also, the table shows the cumulative species residuals after 5000 iterations for each case. The cumulative species residuals number is implemented into the OpenFOAM code and calculates the sum of the residuals of each species over all cells. A lower feed velocity results in higher species residuals in both domains, in the hollow fibers and in the shell. Thus cases with lower feed velocity need more time to reach convergence.

Table 18: Convergence of validation simulations

Case number	Feed velocity [m/s]	Overall iterations	Overall mass balance error [%]	Cum Residuals after 5000 iterations (shell; fibers)
1	0.01	50000	0.1295	168.8; 37.1
2	0.02	40000	0.2099	158.2; 17.7
3	0.05	20000	0.0127	103.8; 10.9
4	0.1	10000	0.0949	25.3; 8.5
5	0.5	6000	0.0008	12.9; 4.7

7.4 Variation of Courant numbers

As described in chapter 3.6.2 the solver *LTSMembraneFoam* is a steady state solver and uses local time stepping. In order to examine if the solution with the *LTSMembraneFoam* solver depends on the Courant number, a simple case with different Courant numbers is compared to the results with the solver *membraneFoam*.

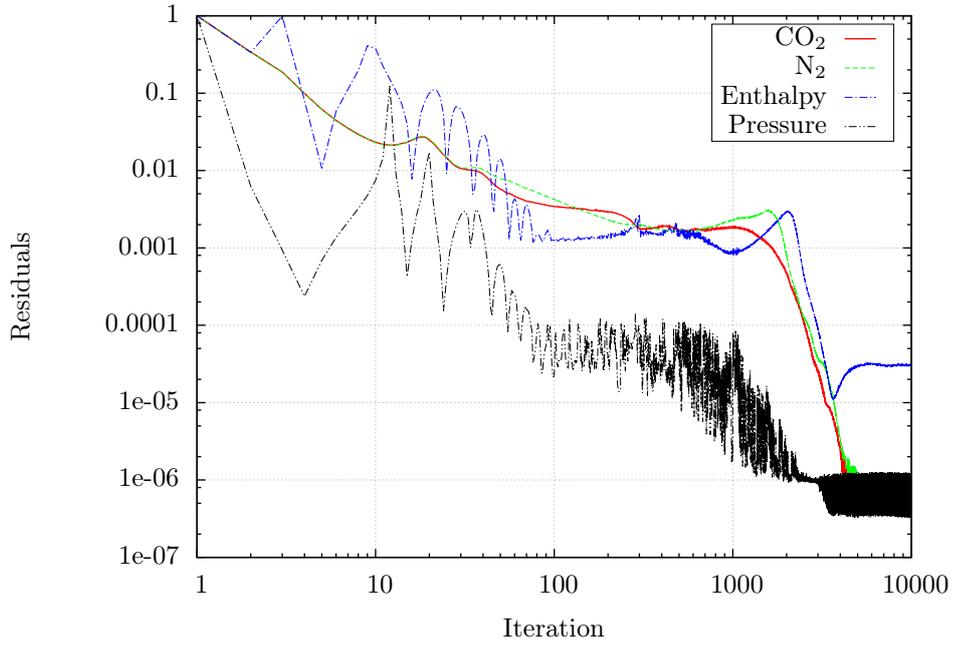
The following table shows the volume stage cuts and the deviations for the cases with the different Courant numbers. All deviations are compared to the transient solver *membraneFoam*. In all cases the feed velocity is 5.68 m/s, the transmembrane pressure is 9 bar and the flow pattern is counter-current.

Table 19: Comparison LTSMembraneFoam and membraneFoam

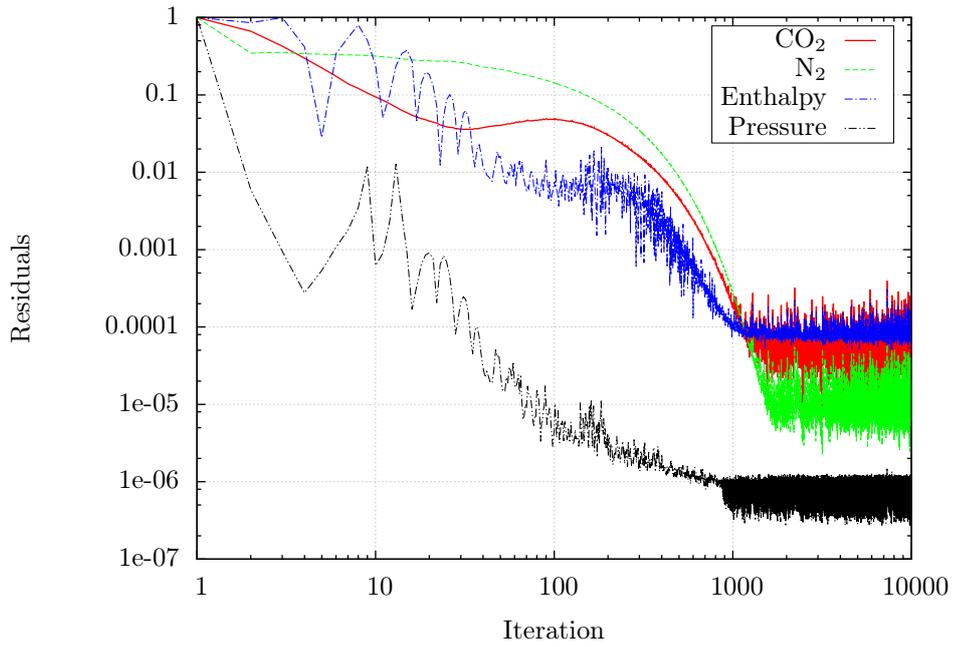
Courant number	Stage cut	Deviation Stage Cut [%]	Overall MB error [%]	Deviation permeate mass flow H ₂ [%]
0.1 (LTSMembraneFoam)	0.010274	-3.70×10^{-3}	0.04669	1.22×10^{-2}
0.5 (LTSMembraneFoam)	0.010274	-4.04×10^{-3}	0.04668	1.25×10^{-2}
1 (LTSMembraneFoam)	0.010274	-4.16×10^{-3}	0.04665	1.25×10^{-2}
2 (LTSMembraneFoam)	0.010274	-3.94×10^{-3}	0.04674	1.26×10^{-2}
5 (LTSMembraneFoam)	0.010274	-3.90×10^{-3}	0.04677	1.26×10^{-2}
1 (membraneFoam)	0.010274	0	0.04702	0

The stage cut and the permeate mass flow in all cases are very close to the results of the transient solver membraneFoam which is considered to be the most accurate. The biggest difference appears in the velocity and the density. In the cases with LTSMembraneFoam solver the velocity on the permeate outlet deviates by 16 % on average from the results with the membraneFoam solver. But as also the density on the permeate side deviates by 14 % on average, the mass flow of all species on the permeate side only deviates slightly. The permeate mass fraction difference between LTSMembraneFoam and membraneFoam is in the range of 10^{-3} % for hydrogen and carbon dioxide and 10^{-2} % for nitrogen.

The following figures show the residuals of the LTSMembraneFoam cases with Courant numbers 0.1 and 5. In case of a low Courant number, the residuals decrease slower compared to the higher Courant numbers. In the case with Courant number 0.1 the residuals reach a stable level after around 5000 iterations whereas the case with the higher Courant number reaches a constant level already after around 2000 iterations. Except for the pressure residual, all residuals of the Courant numbers 0.1, 0.5, 1 and 2 are in a close range as soon as they reach a constant level. The residuals of the case with Courant number 5 are oscillating within a high range, as depicted in figure 35(b). Though, the results of the case with Courant number 5 are as accurate as all other cases. Therefore it can be assumed that a higher Courant number leads to a faster convergence of the solution without changing the result. The simulation could therefore be started with a high Courant number and after around 2000 iterations, switched to a lower Courant number for stable residuals.



(a) Residuals Courant number 0.1



(b) Residuals Courant number 5

Figure 35: Residuals LTSMembraneFoam

8 Results and discussion

The CFD results of the test cases with seven fibers, one fiber and 30 fibers as previously described are discussed in this chapter.

8.1 Hollow fiber membrane with seven fibers

This section discusses the results of the cases which are simulated with the membrane module with seven fibers. Firstly five cases with different outlet positions are discussed. Secondly the convergency check of the simulations are described. Thirdly, five cases with different feed velocities are analyzed which are also part of the validation of the membrane solver.

8.1.1 Flow regime

Figure 36 shows the characteristic parabolic flow rate with the largest velocity in the middle of the pipe and zero velocity at the wall.

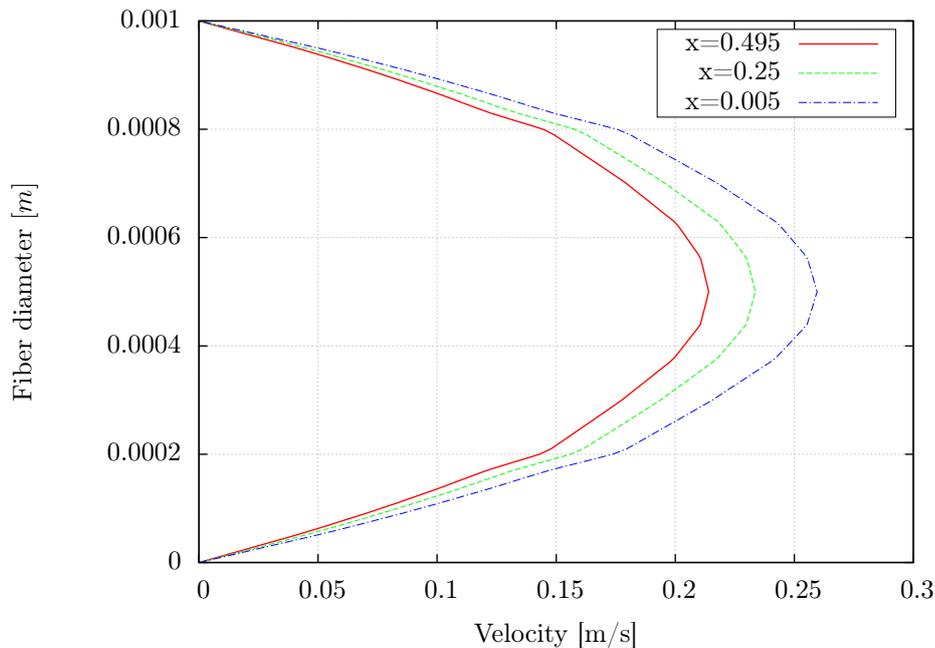


Figure 36: Velocity profile

The flow rates are depicted for three different cross-sections in the pipe. The solid curve displays the flow rate at 0.05 m after the inlet, the dashed curve from the middle and the dotdashed near the outlet. It can be observed, that the velocity is decreasing along the fiber length.

The flow profile is characteristic for a laminar, stationary current in a pipe. Though it has to be mentioned that the characteristic flow regime is a result of the settings in OpenFOAM. Even if the velocity and the diameter of the fiber was several orders of magnitude higher, no turbulent flow regime could have evolved because no turbulence model was used. Therefore, the conclusion of the flow regime can only be gathered from the Reynolds number but not from the results of the simulation.

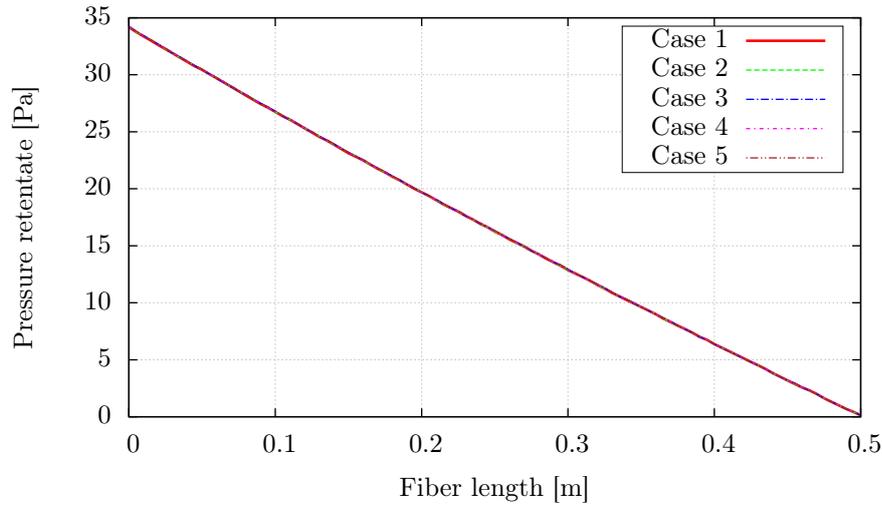
8.1.2 Flow pattern analysis

Pressure

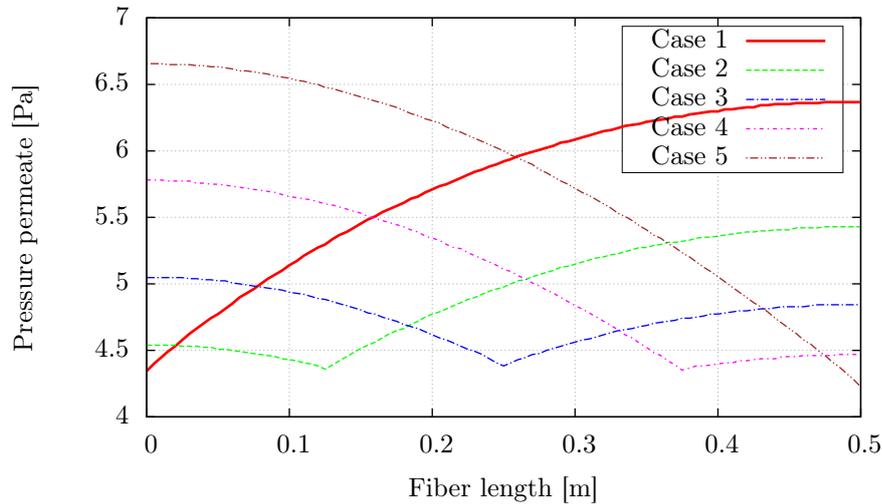
Figures 37(a) and (b) show the pressure drop on both regions, retentate and permeate, whereas the absolute pressure of 9 bar and 1 bar is subtracted. The case number stands for the exit number as labeled in figure 13. The outlet boundary condition is set to 1 bar which is also the area of the lowest pressure. The pressure rises with the distance from the exit. The relative pressure doesn't reach a value of zero at the outlet because the data is extracted from the center of the membrane shell as it is depicted in figure and doesn't go through the outlet area. The area for data extraction is shown in figure 14. The velocity in the shell is low compared to the outlets which are relatively thin. Therefore the major part of the pressure drop results from the exits.

It can be seen that the largest overall pressure drop emerges for the case with exit number five. The pressure drop for case number one is in the same range. The lowest pressure drop emerges for case number 3.

The pressure drop on the retentate side with around 35 Pa is considerably higher than on the permeate side with around 5 to 7 Pa. In each of the analyzed cases, the pressure drop is almost identical as it is depicted in figure 37 (a). The position of the permeate outlet has no influence on the pressure drop on the retentate side. It is governed by the higher velocities and by the small diameter of the fibers.



(a) Relative pressure retentate



(b) Relative pressure permeate

Figure 37: Pressure drop in the 7 fiber module, [Haddadi et al., 2015a], [Haddadi et al., 2015b]

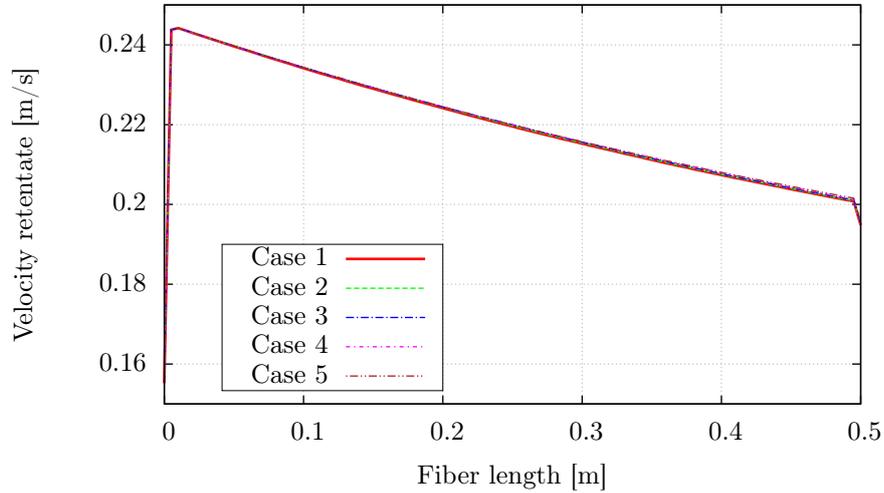
Velocity

The velocities with vector orientation along the fibers are depicted in figure 38. It can be seen that the highest velocities emerge for fully co-current and fully counter-current flow. In cases 1 and 5, the permeate has the greatest volume flow per cross section and thus the highest velocities. For each case the velocity is positive before the exit on the left side and negative after the exit on the right side. As the exit is on the far left side for case 1 the velocity is negative along the whole permeate side. Although the values for the velocity in figure 38 (b) are extracted from a line, the velocity field is representative for the whole permeate side.

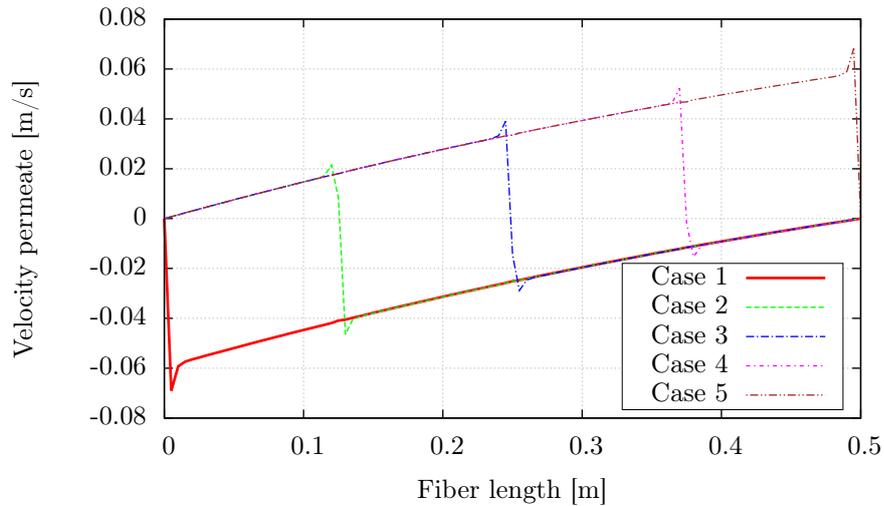
This is not the case for the retentate side as depicted in figure (a). The velocity is extracted from a line in the center of the fiber, where the velocity is considerably higher than the average velocity in the cross section. This results from the laminar velocity profile as depicted in figure 36.

Inside the hollow fiber membrane the velocities for the different outlet configurations are

almost not distinguishable. For all outlet configurations the velocity decreases in the same order of magnitude as a result of mass transfer through the membrane and decreasing mass flow in the fibers.



(a) Velocity retentate



(b) Velocity permeate

Figure 38: Velocity in the 7 fiber module, [Haddadi et al., 2015a], [Haddadi et al., 2015b]

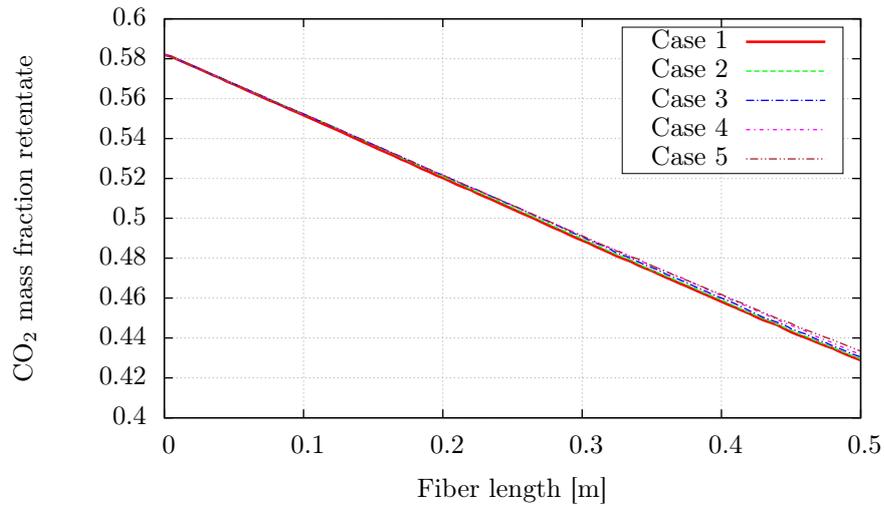
Mass fraction

As it is listed in table 3 carbon dioxide has the highest permeance followed by oxygen and methane. Thus, comparing figure 39 (b) with 40 (b) it can be seen that carbon dioxide has a higher mass fraction on the permeate side than methane. Oxygen has the lowest mass fraction in the permeate which is a result of the low mass fraction in the feed. The mass fraction for carbon dioxide in the shell side is increasing for counter-current flow towards the outlet which is from right to left in figure 39(b). Similarly the mass fraction is decreasing for co-current flow towards the outlet from left to right. In each of the cases the mass fraction of CO_2 is considerably lower for counter-current than for co-current flow. For a high transmembrane flux, the partial pressure difference between feed and permeate

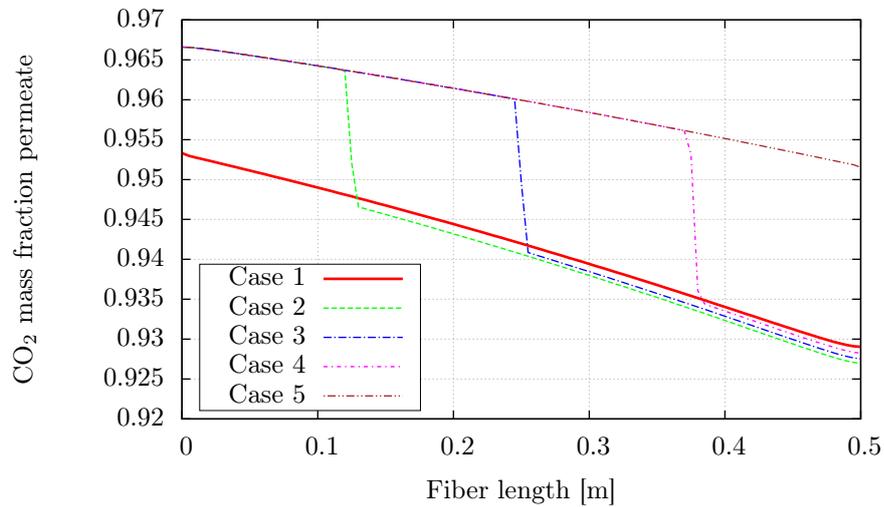
must be high as stated in equation 2.35. The partial pressure is directly proportional to the mass fraction which is lowest on the permeate side for case 1 and highest for case 5. This results in the highest driving force for case 1 leading to the highest mass flow of carbon dioxide of all cases which were investigated.

The sudden increases in the mass fractions in the area of the outlets can be attributed to the shift from co-current to counter-current flow.

On the retentate side a close look on CO_2 shows the lowest mass fraction and the largest permeate mass flow for case 1. Figure 39(a) shows that among all cases, the mass fraction increases along the fiber with increasing share of co-current flow in the shell.



(a) Carbon dioxide retentate

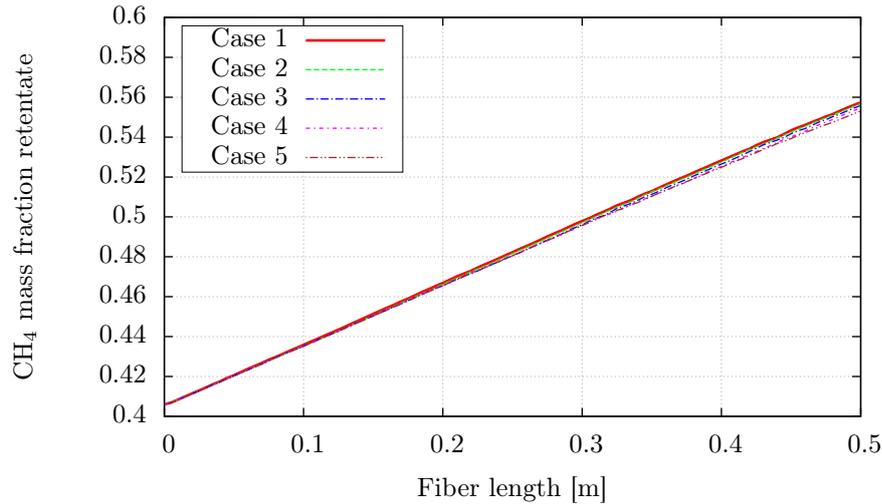


(b) Carbon dioxide permeate

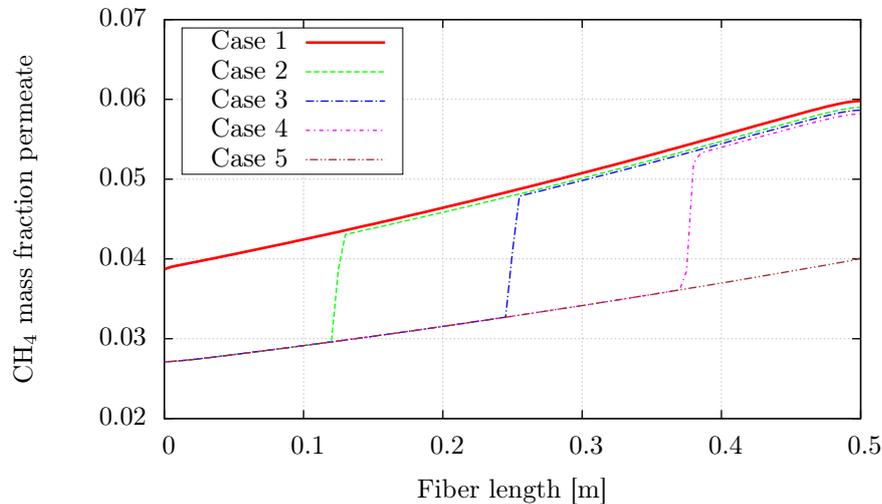
Figure 39: Carbon dioxide in the 7 fiber module, [Haddadi et al., 2015a], [Haddadi et al., 2015b]

Methane and oxygen show the opposite mass fraction distributions along the shell compared to carbon dioxide. Carbon dioxide has the highest permeance and the highest feed mass fraction and therefore governs the mass fraction distribution of the other species on the permeate and retentate side.

As depicted in figure 40, methane has the highest mass fraction along the shell in case 1. This results in the lowest driving force for the transmembrane flux among all cases. Also in the fibers, methane has the highest mass fraction distribution in case 1 which indicates the lowest transmembrane flux. The mass fraction in the fibers decreases from case 1 to 5 with increasing share of co-current flow in the module. Also for methane, the sudden increases in the mass fractions in the area of the outlets can be attributed to the shift from co-current to counter-current flow.



(a) Methane retentate



(b) Methane permeate

Figure 40: Methane in the 7 fiber module

The mass fraction of oxygen is not depicted. As the feed concentration of oxygen is very low also the changes of the concentrations in the fibers are very low. Though the mass fractions of oxygen on the retentate and on the permeate side are similar to methane. On the retentate side case 1 shows the largest mass fraction distribution and case 5 the lowest.

Mass flow

Analysing the mass balance, it can be observed, that around 48 % of the CO₂ from the feed permeates through the membrane. From the oxygen feed around 20 % permeates and from the methane feed flow only around 3 % permeates. These different transmembrane fluxes correspond to the permeances as depicted in table 3. The highest mass flow leaves the module from case 1 and decreases with an increasing share of co-current flow so that the lowest mass flow is in case 5. Similarly also the CO₂ mass fraction and the velocity have the highest values for case 1 and the lowest for case 5.

The maximum overall mass balance error over all species is around 0.1 %. The average mass balance error of CO₂ over all cases is 0.0432 %. The mass flows of oxygen and methane can't be listed as the error in the mass balance is too high to show accurate results. The average mass balance error is around 2.29 % for oxygen and 0.03 % for methane.

The permeate mass flow of CH₄ is expected to be lowest for counter-current flow and highest for co-current flow.

The permeate mass flow of O₂ is expected to be highest for counter-current flow and lowest for co-current flow as oxygen has a higher permeance than methane.

Table 20: Mass balance and stage cut

	Ex 1	Ex 2	Ex 3	Ex 4	Ex 5
Velocity perm [m/s]	1.0619	1.0602	1.0554	1.0501	1.0442
Velocity ret [m/s × 10 ⁻²]	8.1527	8.1680	8.1752	8.1832	8.1957
Mass flow CO ₂ perm [kg/s × 10 ⁻⁶]	1.7378	1.7335	1.7248	1.7152	1.7054
Mass flow CO ₂ ret [kg/s × 10 ⁻⁶]	1.8728	1.8804	1.8828	1.8985	1.9113
Volume stage cut [m ³ /s/m ³ /s]	0.03358	0.03340	0.03346	0.03354	0.03360
Mass frac CO ₂ perm [× 10 ⁻¹]	9.5252	9.5195	9.5190	9.5182	9.5179
Mass frac CO ₂ ret [× 10 ⁻¹]	4.2748	4.2817	4.2919	4.3061	4.3226
Overall MB error [%]	0.0949	0.0239	0.0003	0.0078	0.0214

8.2 Hollow fiber membrane with one fiber

In this section the results of the testcases as shown in table 8 are described. Hydrogen has the highest permeance for the membrane which is used in the simulation and therefore the permeate mass fraction is of specific interest.

8.2.1 Variation of velocities with co-current flow

In table 21 the volumetric stage cuts are depicted for each species. The highest stage cut corresponds to the lowest velocity and vice versa.

The permeate mass fractions of both, nitrogen and carbon dioxide increase with an increasing stage cut. The same phenomenon is also observed in figure 30. With an increase of the stage cut, the purity of the desired component with the highest permeance decreases.

Pressure drop

The pressure drop on the permeate side is 148.6 Pa for the lowest feed velocity, 3.79 m/s and 150 Pa for the highest feed velocity, 11.37 m/s. In contrast, the pressure drop on the retentate side is 12 610 Pa for the highest feed velocity and 3967 Pa for the lowest feed velocity.

Table 21: Permeate composition for different feed velocities and co-current flow

Velocity [m/s]	Stage cut	M-frac H ₂	M-frac CO ₂	M-frac N ₂	MB error [%]	Perm Δp [Pa]	Ret Δp [Pa]
3.79	0.0154	0.65804	0.31660	0.02537	0.05	148.6	3967
5.68	0.0103	0.65892	0.31584	0.02523	0.05	149.1	6038
7.58	0.0077	0.65938	0.31545	0.02517	0.05	149.5	8175
9.47	0.0062	0.65967	0.31521	0.02512	0.05	149.9	10354
11.37	0.0051	0.65987	0.31504	0.02509	0.05	150.2	12460

8.2.2 Variation of velocities with counter-current flow

In the cases with counter-current flow, the same observations can be made as for the co-current cases above. A higher feed velocity results in a higher share of the desired component in the permeance flow. On the other side, the stage cut decreases and the mass fraction of the desired component, hydrogen increases in the retentate.

Table 22: Permeate composition for different feed velocities and counter-current flow

Velocity [m/s]	Stage cut	M-frac H ₂	M-frac CO ₂	M-frac N ₂	MB error [%]	Perm Δp [Pa]	Ret Δp [Pa]
0.125	0.3631	0.58756	0.37217	0.04027	0.04	103.0	103
0.25	0.2085	0.62521	0.34332	0.03147	0.05	126.0	229
0.5	0.1110	0.64336	0.32867	0.02797	0.05	137.6	487
1	0.0572	0.65209	0.32148	0.02642	0.05	143.3	1006
3	0.0154	0.65840	0.31625	0.02535	0.07	147.6	3968
3.79	0.0154	0.65804	0.31660	0.02537	0.05	147.6	3967
5.68	0.0103	0.65892	0.31584	0.02523	0.05	148.1	6038
7.58	0.0077	0.65938	0.31545	0.02517	0.05	148.5	8175
9.47	0.0062	0.65967	0.31521	0.02512	0.05	148.8	10358
11.37	0.0051	0.65987	0.31504	0.02509	0.05	146.3	12610

Pressure drop

The pressure drop on the permeate side is in the range of 148 Pa for all velocities. On the retentate side, the pressure drop is considerably higher with 12 610 Pa for the highest and 3967 Pa for the lowest velocity.

Comparison between co-current and counter-current flow

In this section, only the velocities 3.79 to 11.37 m/s are compared for co- and counter current flow. With the same feed velocities, the counter-current flow pattern shows a higher permeate mass fraction than the co-current flow pattern. This can be also observed in the seven fiber membrane module as depicted in table 20. In figure 41 both flow patterns are compared. For every feed velocity variant the counter-current set-up shows a higher mass fraction of hydrogen and less carbon dioxide in the permeate.

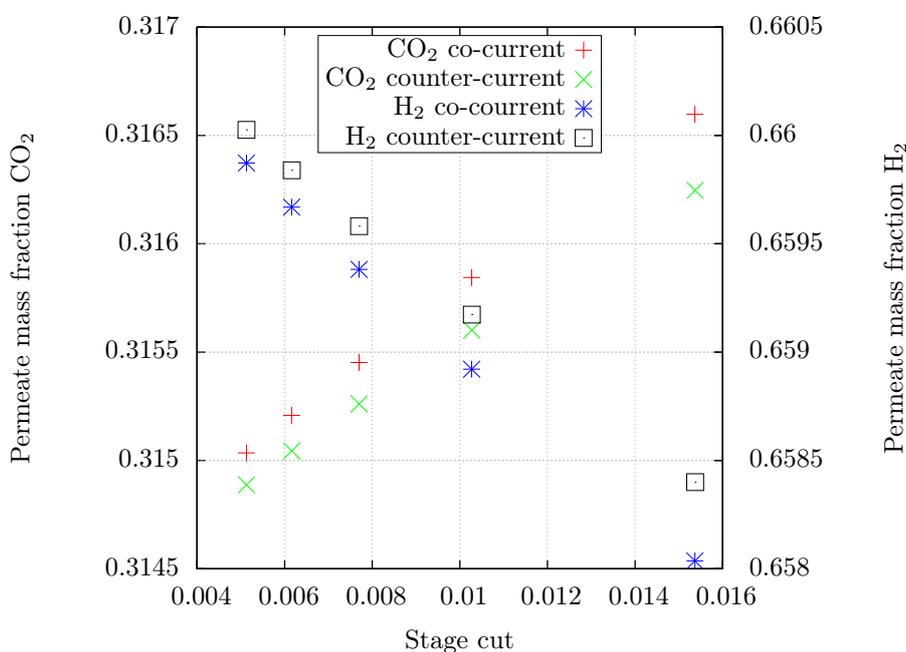


Figure 41: Comparison of different velocities for co- and counter-current flow

The permeate mass fraction of hydrogen for a velocity of 3.79 m/s for the counter-current flow pattern is 0.05 % higher compared to the co-current flow pattern. The difference becomes smaller with increasing velocity. For the highest feed velocity, the difference in the permeate mass fraction of hydrogen is 0.02 %. The same can be observed for the permeate mass fraction of carbon dioxide which shows higher differences for low feed velocities.

Comparing each feed velocity, the counter-current flow pattern shows a higher stage cut in each case. The stage cut for co- and counter-current is the same in both cases with a difference below the magnitude of the mass balance error. Although the results for the permeate mass fractions seem reasonable, they are not significant as the difference for both flow patterns is lower than the mass balance error.

Pressure drop

Comparing both flow patterns, the difference of the pressure drop between co and counter-

current is, lower than 0.04 % on the retentate side and on average 1 % on the permeate side.

For a higher mass flow the pressure difference is higher and therefore the transmembrane flux according to equation 2.34 increases. Though, the increase of the permeate mass flow is much lower than the increase of the feed flow and therefore the stage cut decreases with higher feed velocity.

8.2.3 Variation of transmembrane pressures with co-current flow

Chapter 2.8.4 describes that the driving force for the transmembrane flux in dense membranes depends on the transmembrane pressure which is directly proportional to the molar fraction X_i of a component i in the gas phase. For a constant velocity of 5.68 m/s, table 23 shows that a higher transmembrane pressure (TMP) results in a higher stage cut and in a higher purity of hydrogen on the permeate side. With an increasing TMP, the permeate mass fraction of hydrogen increases and the mass fraction of carbon dioxide decreases.

Table 23: Permeate composition for different TMP and co-current flow

TMP difference [bar]	Stage cut	M-frac H ₂	M-frac CO ₂	M-frac N ₂	MB error [%]	Perm Δp [Pa]	Ret Δp [Pa]
1	0.002820	0.504057	0.41432	0.08162	0.06	9	5972
3	0.007092	0.605591	0.35868	0.03573	0.05	42	5957
5	0.008811	0.636052	0.33479	0.02915	0.05	77	5974
7	0.009716	0.650520	0.32289	0.02659	0.05	113	6004
15	0.011119	0.671056	0.30552	0.02342	0.04	257	6149

Pressure drop

On the retentate side, the pressure drop ranges from 5972 Pa for 1 bar TMP to 6149 Pa for 15 bar TMP. On the permeate side, the relative difference of the pressure drop is higher. For the lowest examined TMP the pressure drop is 8.5 Pa whereas for the highest TMP the pressure drop is 257 Pa.

8.2.4 Variation of transmembrane pressures with counter-current flow

Increasing the TMP, for a constant velocity of 5.68 m/s in the membrane module with counter-current flow shows the same trends as for the co-current flow. The stage cut as well as the permeate mass fraction of hydrogen increase with the applied TMP.

Table 24: Permeate composition for different TMP and counter-current flow

TMP difference [bar]	Stage cut	M-frac H ₂	M-frac CO ₂	M-frac N ₂	MB error [%]	Perm Δp [Pa]	Ret Δp [Pa]
1	0.002845	0.50767	0.41170	0.08064	0.06	8	5972
3	0.007099	0.60636	0.35795	0.03569	0.05	41	5955
5	0.008816	0.63649	0.334374	0.02914	0.05	77	5975
7	0.009719	0.65084	0.32258	0.02658	0.05	112	6005
15	0.011121	0.67122	0.30537	0.02342	0.04	256	6149

Pressure drop

On the retentate side, the pressure drop for counter-current flow is in the same range

as for the co-current cases. On the permeate side, the relative difference of the pressure drop is higher. For the lowest examined TMP the pressure drop is 8.3 Pa whereas for the highest TMP the pressure drop is 255 Pa.

Comparison between co-current and counter-current flow

Figure 42 compares co and counter-current flow at different TMP. In all cases the counter-current flow shows a higher stage cut. The highest difference in the stage cut of 0.9 % appears for the lowest TMP of 1 bar. For the highest TMP, 15 bar the difference is 0.02 %.

Also for the permeate mass fraction of hydrogen the same observations can be made that in all cases the counter-current flow shows a better performance. For the lowest TMP the difference is 0.7 % and for the highest TMP the difference is 0.02 %. The differences of the permeate mass fraction of carbon dioxide and nitrogen follow the same trend.

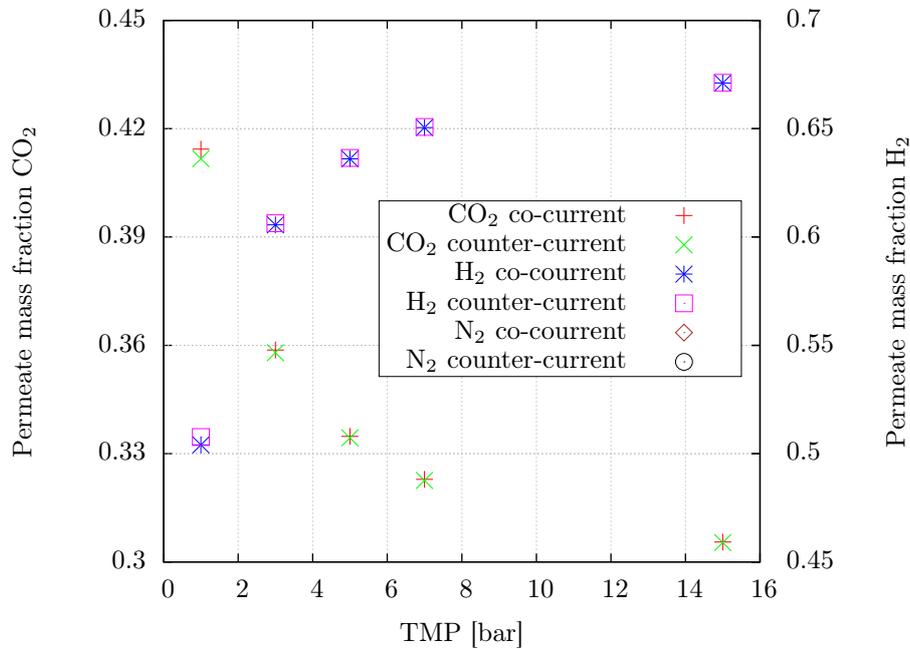


Figure 42: Comparison of different transmembrane pressures for co and counter-current flow

8.2.5 Variation of velocities with higher permeances

In this section, five different velocities in counter-current mode are analyzed, similar as in section 8.2.2, whereas the permeances are increased tenfold for all species compared to the standard values as depicted in table 7.

Table 25: Permeate composition for different feed velocities and higher permeances

Velocity [m/s]	Stage cut	M-frac H ₂	M-frac CO ₂	M-frac N ₂	MB error [%]	Perm Δp [Pa]	Ret Δp [Pa]
3.79	0.1421	0.6353	0.3352	0.0295	0.02	1321	3664
5.68	0.0972	0.6430	0.3290	0.0280	0.02	1372	5704
7.58	0.0738	0.6468	0.3258	0.0273	0.03	1392	7811
9.47	0.0594	0.6491	0.3239	0.0269	0.03	1413	9965
11.37	0.0497	0.6507	0.3227	0.0267	0.04	1420	12188

Comparison between low and high permeances

Analogous to the previous study with different velocities and counter-current flow, the permeate mass fraction of hydrogen increases with the feed velocity. Comparison of table 22 and 25 shows that the stage cut increases by a factor of 9.2 for the lowest and by 9.7 for the highest feed velocity as a result of the increase of the permeance.

On the other side, the permeate mass fraction and the purity of hydrogen in the permeate decreases. For the lowest velocity, the permeate mass fraction of hydrogen decreases by 3.6 % whereas the mass fraction of carbon dioxide and nitrogen increase by 5.8 % and 15 % respectively. For higher velocity, the difference in the permeate mass fraction between high and low permeances decreases to 1.4 % for hydrogen.

Thus, in case the permeances increase by a factor of ten for all species, the mass flow on the permeate side doesn't increase by the same amount.

Pressure drop

The pressure drop on the retentate side exceeds the pressure drop from the cases with the lower permeances which are described in section 8.2.2. For the lowest velocity the difference is around 8 % whereas for the highest velocity the difference is 3.4 %. On the permeate side, the pressure drop increases 9 times for the lowest velocity and 9.7 times for the highest velocity.

8.2.6 Variation of the permeance of hydrogen

In this section, five different permeances in counter-current mode are analyzed and compared to the base value as described in table 7. In each case the base value of the permeance of hydrogen is multiplied by a factor and the permeances of carbon dioxide and nitrogen are kept constant. It becomes obvious that with a higher permeance of hydrogen, its mass fraction in the permeate increases and the mass fraction of carbon dioxide and nitrogen decreases. This leads to an increase of the stage cut as the feed velocity and the feed flow is constant in all cases. In case the permeance of hydrogen is doubled, the stage cut rises by a factor of 1.6. In the case with the highest transmembrane flux, the permeance of hydrogen is 32 times higher than the base case the stage cut rises by a factor of 11.

Table 26 shows that a doubling of the permeance of hydrogen leads to an increase of its mass fraction by 17 % compared to the base value. Increasing the permeance of hydrogen to $1.8764 \times 10^{-8} \frac{m^3(STP)}{Pa \cdot s \cdot m^2}$ which is 32 times the base value, increases its mass fraction in the permeate by 46 % compared to the base value. Whereas a doubling in a low permeance value leads to a high increase in the permeate mass fraction and the stage cut, a doubling of an already high permeance value only marginally increases its mass fraction in the permeate. This can be ascribed to the driving force which, according to equations

2.34 and 2.36 decreases as the retentate mole fraction decreases and the permeate mole fraction increases.

Table 26: Permeate composition for different permeances of hydrogen

Permeance [m ³ /(m ² Pa s)]	Stage cut	M-fraction H ₂	M-fraction CO ₂	M-fraction N ₂	MB error [%]	Perm Δp [Pa]	Ret Δp [Pa]
x1	0.0154	0.65804	0.31660	0.02537	0.05	148	3966
x2	0.0244	0.77554	0.20835	0.01611	0.03	233	3946
x4	0.0408	0.86104	0.12922	0.00974	0.01	384	3908
x8	0.0696	0.91582	0.07837	0.00581	0.03	639	3839
x16	0.1151	0.94704	0.04933	0.00363	0.09	1012	3723
x32	0.1715	0.96262	0.03482	0.00256	0.11	1377	3555

As the molecular weight of hydrogen is relatively low, the density in the permeate decreases with an increase of the permeance. Further, the velocity and thus also the pressure drop increase on the permeate side with a higher mass fraction of hydrogen. On the retentate side, the opposite phenomena can be observed. The density increases while the velocity and the pressure drop decrease with higher permeances of hydrogen. These observations and the permeance mass fraction of hydrogen and carbon dioxide are also depicted in figure 43. The scale for the permeate mass fraction is on the left and the pressure drop scale is on the right side of the graph.

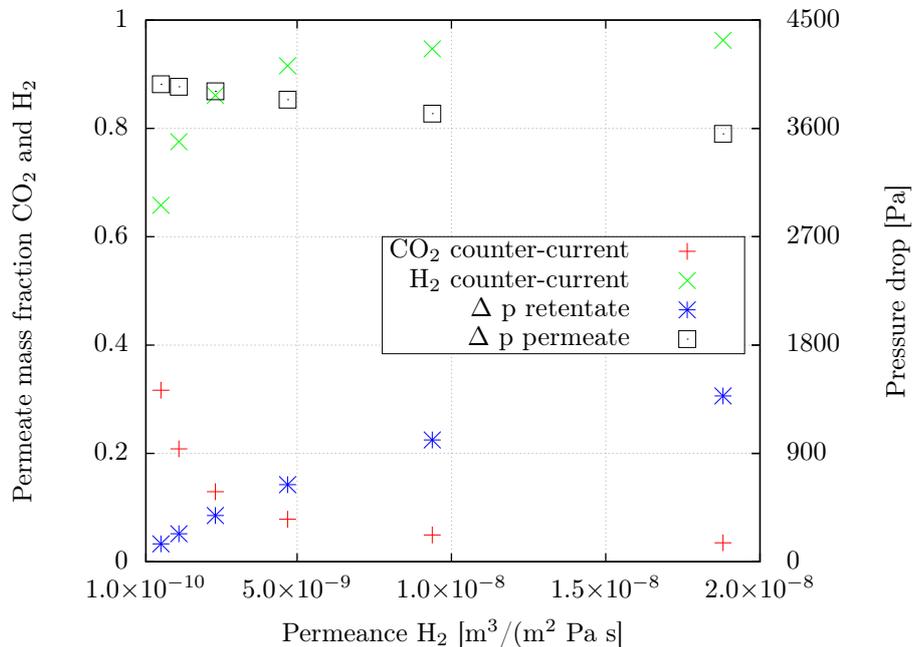


Figure 43: Permeate composition and pressure drop at different permeances of hydrogen

Pressure drop

On the permeate side, a doubling of the permeance compared to the base value leads to a pressure drop of 233 Pa whereas for the highest permeance, the pressure drop is 1377 Pa. On the retentate side, the highest permeance leads to a pressure drop of 3555 Pa.

8.2.7 Variation of permeances of all species

In the following table 27 the results of varying permeances of all species with a constant velocity of 3.79 m/s are shown. The factors show how many times the permeance of each substance is increased compared to the values as listed in table 7. The increase of all permeances leads to an increase of mass flow, to a higher velocity and thus to a higher pressure drop on the permeate side.

Also the stage cut increases with the increasing permeances and if the feed flow is kept constant. In case all permeances are increased by a factor of two, the stage cut increases by a factor of 1.98. From the cases which are investigated, in case the permeances are increased by the maximum factor of 32, the stage cut only rises 24 fold.

The permeate mass fraction decreases with a higher transmembrane flux of all species. Whereas for the lowest permeance, the permeate mass fraction of hydrogen is 65.8 %, the value decreases to 57.9 % for the highest permeance. For the other species, carbon dioxide and nitrogen, the opposite can be observed. Here the permeate mass fraction increases with the transmembrane flux.

Table 27: Permeate composition for different permeances of all species

Permeance [m ³ /(m ² Pa s)]	Stage cut	M-frac H ₂	M-frac CO ₂	M-frac N ₂	MB error [%]	Perm Δp [Pa]	Ret Δp [Pa]
x1	0.0154	0.65804	0.31660	0.02537	0.05	148	3966
x2	0.0305	0.65587	0.31836	0.02578	0.04	291	3931
x4	0.0560	0.65076	0.32259	0.02665	0.02	567	3862
x8	0.1157	0.64046	0.33104	0.02850	0.01	1082	3728
x16	0.2150	0.61965	0.34769	0.03266	0.03	1965	3484
x32	0.3679	0.57906	0.37807	0.04287	0.01	3141	3078

In figure 44 the pressure drop and the permeate mass fraction is plotted over the factors of increasing permeances of all species. The base values of the permeances are given in table 7. As all permeances are increased by the same factor, the pressure drop and also the permeate mass fractions show a direct proportional relationship to the change in the permeances.

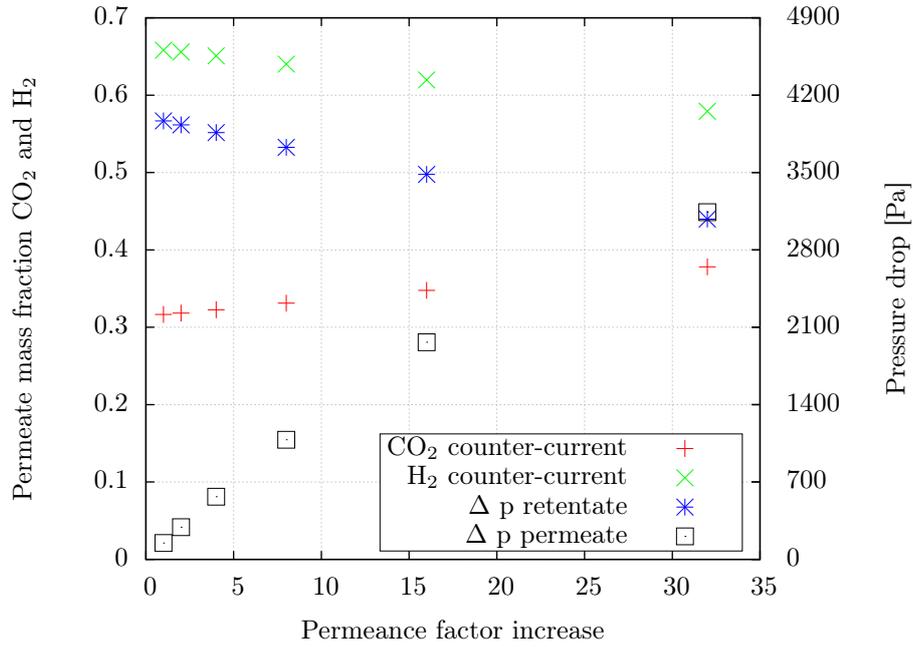


Figure 44: Permeate composition and pressure drop at different permeances of all species

Pressure drop

On the retentate side, the highest pressure drop of 3966 Pa appears for the lowest permeance and the lowest pressure drop is 3078 Pa for the highest permeance. The mass flow, the velocity and the pressure drop decrease if the permeances of all species are increased. On the permeate side, the lowest pressure drop is 148 Pa for the lowest permeance and the highest pressure drop is 3141 Pa for the highest permeance.

8.3 Hollow fiber membrane module with 30 fibers

First, the results from the lab experiments are described. Second, the results of the CFD simulation are described and compared to the lab experiments.

8.3.1 Laboratory experiments

Table 28 shows the results from three pure gas laboratory measurements with the pure components carbon dioxide (CO₂), methane (CH₄) and hydrogen (H₂). The table shows the inlet and outlet flows, the transmembrane pressure (TMP) and the deviation of the mass balance. In all three cases the goal is to reach a feed pressure of 4 bar and a permeate pressure of 1 bar which should result in a TMP of 3 bar. In all experiments, the permeate side shows a slight under pressure of around -0.2 bar. The transmembrane pressure is calculated by the difference between the average of the feed and the retentate pressure and the pressure on the permeate side which are all listed in table 29.

Table 28: Mass flows from lab experiments

Specie	TMP [bar]	Feed flow [kg/s]	Feed flow [Nm ³ /s]	Perm flow [Nm ³ /s]	Ret flow [Nm ³ /s]	MB Deviation [%]
CO ₂	3.23	1.3866×10^{-5}	7.0141×10^{-6}	9.7750×10^{-8}	6.9128×10^{-6}	1.6
CH ₄	3.12	5.5548×10^{-6}	7.7423×10^{-6}	2.2388×10^{-8}	6.9369×10^{-6}	10.1
H ₂	2.99	5.5670×10^{-7}	6.1935×10^{-6}	2.1593×10^{-8}	5.2446×10^{-6}	14.9

The following table shows the pressures resulting from the laboratory experiments

Table 29: Pressures from lab experiments

Specie	Abs. Pressure feed [Pa]	Abs. Pressure retentate [Pa]	Abs. Pressure permeate [Pa]	Abs. Pressure drop [Pa]
CO ₂	3.07	400241	80392	7211
CH ₄	3.24	400865	79493	22655
H ₂	2.97	400348	79715	NV

With the given membrane area from the datasheet of 10 cm² and a pressure across the membrane of around 3 bar the permeances are calculated as defined in equation 2.9. For example the permeance of the specie methane is calculated by

$$Q_{\text{CH}_4} = \frac{2.2388 \times 10^{-8}}{10^{-4} \cdot 3.33} = 6.73 \times 10^{-11} \left[\frac{\text{m}^3 (\text{STP})}{\text{Pa s m}^2} \right] \quad (8.1)$$

The permeance of the specie carbon dioxide is calculated by the average of five experiments, whereas in each case the transmembrane pressure is changed. The TMP for the five cases are 3.04, 2.95, 2.84, 2.77, 2.71 and 2.62 bar and the resulting permeances are 3.02×10^{-10} , 3.05×10^{-10} , 2.96×10^{-10} , 2.83×10^{-10} , 2.76×10^{-10} and 2.81×10^{-10} Nm³/(m² Pa s). In this section, only the results of the first experiment with CO₂ are listed. For both specie, methane and hydrogen, only one experiment is carried out. The following table 30 shows the resulting permeances of all species from the lab experiments, the permeability from the data sheet and the selective layer which is calculated with equation 2.8.

Table 30: Permeances of species from lab experiments

Specie	Permeance lab [Nm ³ /(m ³ Pa.s)]	Permeability data sheet [Nm ³ m/(s m ² Pa)]	Selective layer thickness [μm]
CO ₂	2.90×10^{-10}	2.44×10^{-15}	8.1
CH ₄	6.73×10^{-11}	7.13×10^{-16}	10.6
H ₂	6.77×10^{-11}	4.88×10^{-16}	7.2

8.3.2 CFD simulations

In order to obtain the inlet velocities for the CFD simulation, the mass flow from the lab experiments must be converted to a volume flow. Therefore the mass flow is scaled with the density at the operating conditions of 4.23, 4.12 and 2.99 bar and 295 K at the inlet. The velocities are calculated from the volume flow and the inlet surface area.

To compare the feed and the retentate volume flow, the values from table 28 are corrected for the mass balance error and compared to the values from the simulation from table 31. The transmembrane pressure (TMP) is set according to the lab experiments.

The permeances are taken from the lab experiments as listed in table 30 and scaled with the factor 1.5789, which corresponds to the ratio of inner and outer fiber surface area.

Table 31 shows the feed flows which are used as input parameters for the simulation. The mass balance in table 28 shows a higher deviation for hydrogen than for carbon dioxide. It is assumed that especially the lighter species diffuse through the plastic tubes before and after the membrane. Therefore, in order to correct the mass balance, 50 % of the mass balance error is corrected on the feed and 50 % on the retentate side. Therefore the amount of the input parameter feed flow is reduced by half of the error and the retentate mass flow is increased by half of the error so that the mass balance is correct. The velocity is calculated from the corrected feed flow with the inlet surface area.

With velocities of about 0.5 m/s and a fiber diameter of 190 μm the Reynolds number of the pure gases is in the range of 160. Thus the flow profile is considered to be laminar.

Mass flows and pressure drop

The following table shows the corrected mass and volume feed flows as well as the resulting permeate and retentate volume flows and the pressure drop for for each specie.

Table 31: Mass flows from the CFD simulations

Specie	Feed flow [kg/s]	Feed flow [Nm ³ /s]	Perm flow [Nm ³ /s]	Ret flow [Nm ³ /s]	MB Deviation [%]	Δp [Pa]
CO ₂	1.3246×10^{-6}	6.9448×10^{-6}	8.3786×10^{-8}	6.9188×10^{-6}	0.27	2341
CH ₄	4.9706×10^{-6}	7.3368×10^{-6}	1.8814×10^{-8}	7.3498×10^{-6}	0.16	2372
H ₂	4.2756×10^{-7}	5.7189×10^{-6}	1.8891×10^{-8}	5.7111×10^{-6}	0.15	1590

Comparing table 28 and 31 the permeate flows from the experiments and the simulations show a deviation of 15 % for carbon dioxide, 13 % for hydrogen and of 17 % for methane.

The comparison of the corrected values of the feed volume flow shows a deviation of 0.19 % in all three cases. The retentate flow deviates by 0.8 % for carbon dioxide, by 0.05 % for hydrogen and by 0.3 % for methane.

8.4 Deviation between simulation and experiments

Modeling of the transmembrane flow

The influence of the resistances to the transmembrane flow, such as concentration polarization or the pressure drop in the porous layer, as mentioned in section 2.5, is not accurately considered as the membrane module is modeled with an infinitely thin membrane. Though, as a result of the high diffusion, concentration polarization only has a minor effect on the resistance to the transmembrane flow.

Apart from the concentration profile, also the temperature can have an effect on the performance of the membrane. As it is shown in equation 2.12, a higher temperature increases the permeability of the membrane. In the current set up, the dependence of the permeability on the temperature is not considered and the experiments and the simulations are carried out at constant temperatures.

Measurement errors

The accuracy of the permeate and retentate flow meters, according to the data sheet, is 1 % standardized and 0.75 % volumetric.

The mass flow controller has an accuracy of 0.2 % for liquids and 0.5 % for gases.

As it is depicted in table 28, the mass balance of the lab balance deviates considerably for hydrogen with 15 %. For methane the mass balance deviates by 10 % and for carbon dioxide by 1.6 %.

It is described in chapter 5 that the feed and retentate pressure sensors have a measurement range of 0 to 100 bar and 0 to 25 bar respectively. Both sensors have an accuracy of 0.5 %. The high measurement range can explain the low pressure on the permeate side. It is expected that the actual pressure on the permeate side is higher than the atmosphere pressure.

Modeling and simulation inaccuracies

It needs to be stated that results from CFD simulations deliver approximate results as the equations are solved numerically. Firstly, the differential equations describing fluid phenomena may contain approximations. Secondly, approximations are made in the discretization of the conservation equations with the algebraic system of equations. Thirdly, the iterative process to solve the discretized equations is aborted after a certain time resulting in a difference between the iterative solution and the exact solution of the algebraic equations system. [Ferziger and Peric, 2002]

A minor source of error is the difference between the actual area of the membrane inlet and the area which results from the mesh.

The measured inlet diameter of the membrane module is 2 mm and thus the resulting inlet area is 3.1416 mm². The surface from the mesh on the inlet is 3.136 246 mm², which gives a deviation of 0.17 %. This can explain the deviation of the feed mass flow of 0.19 % in each case.

The input parameters for the simulation are the velocities, the mass fractions and the outlet pressures. The density in OpenFOAM is thus calculated with the perfect gas equation $\rho = p/(R * T)$ which gives the volume flow at operating conditions. On the contrary, in the experiments, the mass flow is converted to volume flow with the densities from the NIST chemical webbook.

Permeance

Another source of inaccuracy is the membrane surface area of 10 cm^2 which is taken from the data sheet to calculate the permeances. It is shown in section 4.4.2 that the actual outer membrane surface area, based on rough measurements, is 12.86 cm^2 . Though, taking a larger surface area for calculating the permeance, would reduce the permeances for all species and therefore increase the deviations.

The permeate mass flow in the simulations is underestimated. This indicates, that the actual outer surface area of the membrane could be higher. This can be ascribed to a higher outer surface area as the fiber bundle is twisted in the module and therefore the membrane surface is increased.

9 Conclusion and Outlook

Prior to this thesis, a new transient solver, *membraneFoam* has been developed, based on the Open Source Field Operation and Manipulation library, OpenFOAM[®], version 2.3.1, which can solve different problems in fluid mechanics. The new solver can simulate mass and energy transfer separately for several regions. By adding source and sink terms on the boundaries, heat and mass transfer can be solved for each region.

In membrane processes, often steady state conditions need to be analyzed. Therefore the local time stepping (LTS) method was added to the *membraneFoam* solver. With the LTS method, the time step is set as high as possible for every cell which drastically reduces the simulation effort. The new steady state solver is called *LTSMembraneFoam*.

For all simulations, dense polymer membranes were assumed with mass transfer based on the solution-diffusion model.

In order to test the membrane solver code, a 3-D model of a module with seven fibers and five different permeate outlet positions is created and meshed. With this module, the flow normal to the membrane and flow patterns like co-current, counter-current and mixed-co- and counter-current, are analyzed. The assumption of a higher separation efficiency for fully developed counter-current flow could be confirmed.

Another part of this work was to create test cases with different inlet mass flows with the same seven-fiber module. The results were used, to validate the *LTSMembraneFoam* solver against a model which was created with the commercial process simulation software Aspen Custom Modeler[®], ACM. The results from the ACM model were already validated against experiments and thus, the ACM model was proved to work properly and useful for the simulation of gas permeation separation processes. The validation of the newly developed CFD code showed comparable results to the ACM model. Variations can be reasonably explained with the setup and the model parameters of both models.

Further, a single fiber 3-D membrane module is created in order to examine the influence of different Courant numbers. The results of cases with different Courant numbers with the solver *LTSMembraneFoam* are compared with the results of the transient solver *membraneFoam*. It could be shown, that a higher Courant number leads to faster convergence but has no influence on the final solution.

The single fiber module was also used to investigate the influence of different flow patterns, transmembrane pressures, permeances and inlet mass flows on the separation performance. The examination of co- and counter-current flow with different transmembrane pressures resulted in very low stage cuts. To further investigate the influence of flow patterns, lower feed velocities or higher permeances should be used.

All simulations showed that the solver generates reasonable and reliable, reproducible data for stage cuts, permeate- and retentate compositions and the pressure drops.

In a first step, an attempt was made to mesh the single fiber membrane module with the OpenFOAM tool *snappyHexMesh*. This approach was aborted because no mesh could be generated with an efficient amount of cells. One possible solution is the dictionary *extrudeMeshDict* which allows the extrusion and grading to a predefined 2-d mesh. This approach looks promising for straight pipes but difficult for bent ones.

One major task for the thesis was the modeling of the PDMSXA-10 membrane module from the company PermSelect, Inc, with 30 hollow fibers and 10 cm² surface area on the outer membrane wall. The special feature of this membrane is its high permeability for carbon dioxide compared to other common gases such as methane, hydrogen, oxygen or nitrogen.

The membrane module model was created with a commercial CAD software. The drawing, which had several separate regions, was imported into *snappyHexMesh*, a meshing tool which is supplied with OpenFOAM, and meshed.

Results from simulations with too high permeances showed an influx to the membrane module also from the retentate outlet. This behavior is not expected in real membrane modules and this setting needs to be fixed in a future version of the *membraneFoam* solver.

The simulation results of the 30-fiber module are compared to data from lab measurements. The retentate flow shows a deviation between 0.05 % and 0.8 %. The permeate flow shows a deviation in the range of 15 % which can be attributed to the simplification which had to be made for the membrane. The actual module has a membrane wall thickness of $55\ \mu\text{m}$, whereas the membrane fibers in the simulation needed to be created with infinitely thin walls. The actual membrane surface could not be measured but had to be calculated. As the membrane area for the simulated membrane module is too low, in a next step, swirled, half twisted fibers could be created.

With the current setup of the membrane module, in a further calibration step, the permeances could be adapted and calibrated so that the same permeate mass flow as in the experiments is reached. Further, with the adapted permeances, mixed gas simulations can be done.

Implementing a porous layer to consider membrane wall thickness is an important next step for the development of the solver code. The 30 fiber module has two outlets. It could also be interesting to use one outlet as an inlet for sweep gas and investigate the influence on concentration polarization and separation efficiency.

Recently, an open source solver code was published, which handles mass flow through asymmetric membrane channels to simulate forward osmosis. [Koch, 2015] The solver models the effects of concentration polarization on the rejection and on the draw sides of the membrane on the osmotic pressure loss. Hence, the implementation of a new solver code to consider other fluids than gases in order to investigate membrane separation under a different driving force, e.g. osmotic pressure, can be subject to further investigation.

At the moment, the solver is only capable to model dense membranes. The implementation of molecular sieving and Knudsen diffusion, which is currently only handled by commercial software can be considered for the long term development of the base solver. Further work which can be done without modification of the current solver is the analysis of the effects of mixing promoters, baffles and spacers on the turbulence and on the flow profile, the concentration gradient of the species and the separation efficiency. Therefore, not only hollow fiber membranes but also tubular or flat membrane modules could be modeled.

In general, the permeances change with the temperature. Another task could be the determination of certain permeances at different temperatures with lab experiments and compare the results with the simulations. In all simulations, the module geometry was kept constant and only parameters like pressure, feed flow or permeances were changed. In order to further optimize hollow fiber membrane modules with given feed flow, the parameters fiber outer diameter and length of the fiber can be modified. For a smaller fiber diameter, the packing density increases whereas on the other side, the pressure drop increases. The pressure drop also increases if the length of the module increases. The literature suggests that the permeance is constant for all transmembrane pressures. In the laboratory experiments, described in section 8.3.1 this was not the case. Another addition to the solver could be the implementation of pressure-dependent permeances.

References

- [Andalib et al., 2015] Andalib, E., Ghasemzadeh, K., and Basile, A. (2015). Modelling study of palladium membrane reactor performance during methane steam reforming using CFD method. *The 12th international conference on Membrane Science and Technology (MST 2015)*.
- [Babu et al., 2014] Babu, V., Thapliyal, A., and Patel, G. K. (2014). *Bifuels Production*. Scrivener Publishing.
- [Baker, 2004] Baker, R. W. (2004). *Membrane Technology and Applications*. John Wiley & Sons, Inc.
- [Banitaba et al., 2015] Banitaba, F. S., Mansurpur, Z., and Fatemi, S. (2015). Cfd simulation of co2-ch4 mixture separation by sapo-34 tubular membrane. *The 12th international conference on Membrane Science and Technology (MST 2015)*.
- [Bird et al., 2002] Bird, R., Stewart, W., and Lightfoot, E. N. (2002). *Transport Phenomena*. John Wiley & Sons, Inc.
- [CFDDirect, 2016] CFDDirect (2016). *website cfddirect*. Website: <http://cfd.direct/openfoam/user-guide/standard-utilities>, accessed 01/13/2016.
- [Costa et al., 2005] Costa, A. P., Jordan, C., Friedl, A., and Harasek, M. (2005). Modeling of hollow fiber transverse flow modules using CFD. *Projektbericht ZID 05-166-6, Institute of Chemical Engineering, Vienna University of Technology*.
- [Courant et al., 1928] Courant, R., Friedrichs, K., and Lewy, H. (1928). Über die partiellen Differenzgleichungen der mathematischen Physik. *Mathematische Annalen Bd. 100, S. 32-74*.
- [Ferziger and Peric, 2002] Ferziger, J. H. and Peric, M. (2002). *Computational Method for Fluid Dynamics*. Springer-Verlag.
- [Fletcher, 1988] Fletcher, C. A. J. (1988). *Computational Techniques for Fluid Dynamics 2 Specific Techniques for Different Flow Categories*. Springer-Verlag.
- [Ghidossi et al., 2006] Ghidossi, R., Veyret, D., and Moulin, P. (2006). Computational fluid dynamics applied to membranes: State of the art and opportunities. *Chemical Engineering and Processing, 437-454*.
- [Gruber et al., 2012] Gruber, M. F., Johnson, C. J., Tang, C., Jensen, M. H., Yde, L., and Helix-Nielsen, C. (2012). Validation and analysis fo forward osmosis cfd model in complex 3d geometries. *Membranes 2012, 2(4), 764-782; doi:10.3390/membranes2040764*.
- [Gruber et al., 2011] Gruber, M. F., Johnson, C. J., Tang, C. Y., Jensen, M., Yde, L., and Helix-Nielsen, C. (2011). Computational fluid dynamics simulation of flow and concentration polarization in forward osmosis membrane systems. *Journal of Membrane Science 379, 488-495*.
- [Haddadi et al., 2015a] Haddadi, B., Jordan, C., Schretter, P., Lassmann, T., and Harasek, M. (2015a). Designing better membrane modules using CFD. *The 12th international conference on Membrane Science and Technology (MST 2015)*.

- [Haddadi et al., 2015b] Haddadi, B., Jordan, C., Schretter, P., Lassmann, T., and Harasek, M. (2015b). Designing better membrane modules using CFD. *Chemical Product and Process Modeling, ISSN: 1934-2659*.
- [Izadiar et al., 2015] Izadiar, A., Aroujalian, A., Moraveji, M. K., and Asl, M. S. (2015). A model for gas permeation using mixed matrix membranes in temperature and pressure different. *The 12th international conference on Membrane Science and Technology (MST 2015)*.
- [Koch, 2015] Koch, J. (2015). Genetic algorithm based optimization of baffle positions in a forward osmosis draw channel. *Engineering and Applied Science Thesis & Dissertations, Paper 83*.
- [Kuhlmann, 2007] Kuhlmann, H. C. (2007). *Strömungsmechanik*. Pearson Studium.
- [Lassmann, 2015] Lassmann, T. (2015). *The purification of fermentatively produced hydrogen using gas permeation: A practical and simulative approach*. PhD thesis, TU Wien.
- [Makaruk and Harasek, 2009] Makaruk, A. and Harasek, M. (2009). Numerical algorithm for modelling multicomponent multipermeator systems. *Journal of Membrane Science, 344, 258-265*.
- [Makaruk et al., 2010] Makaruk, A., Miltner, M., and Harasek, M. (2010). Membrane biogas upgrading processes for the production of natural gas substitute. *Separation and Purification Technology, 74, 83-92*.
- [Mansourpour et al., 2015] Mansourpour, Z., Halimejani, H. Z., Sadeghnejad, S. M., and Boskabadi, M. (2015). Cfd simulation of hydrogen separation in pd hollow fiber membrane. *The 12th international conference on Membrane Science and Technology (MST 2015)*.
- [Marriott and Sorensen, 2003] Marriott, J. and Sorensen, E. (2003). A general approach to modelling membrane modules. *Chemical Engineering Science, 4975-4990*.
- [Mulder, 1996] Mulder, M. (1996). *Basic Principles of Membrane Technology*. Kluwer Academic Publisher.
- [Nanduri, 2011] Nanduri, S. (2011). Cfd investigation of mass transfer to crimped hollow fiber membranes. Master's thesis, The University of Toledo.
- [NIST, 2016] NIST (2016). *NIST Chemistry WebBook, National Institute of Standards and Technology*. Website: <http://webbook.nist.gov/chemistry/fluid/>, accessed 01/13/2016.
- [Ohlrogge and Ebert, 2006] Ohlrogge, K. and Ebert, K. (2006). *Membrane Grundlagen, Verfahren und industrielle Anwendungen*. Wiley-VCH Verlag GmbH & Co.KGaA.
- [OpenFOAM, 2014] OpenFOAM (2014). Openfoam, the open source CFD toolbox, user guide, version 2.3.1.
- [openfoamwiki.net, 2016a] openfoamwiki.net (2016a). Website: https://openfoamwiki.net/index.php/OpenFOAM_guide/The_PIMPLE_algorithm_in_OpenFOAM, accessed 01/13/2016.

- [openfoamwiki.net, 2016b] openfoamwiki.net (2016b). Website: https://openfoamwiki.net/index.php/Getting_started_with chtMultiRegionSimpleFoam_-_planeWall2D, accessed 01/13/2016.
- [Pabby et al., 2009] Pabby, A. D., Rizvi, S. S. H., and Sastre, A. M. (2009). *Handbook of Membrane Separations Chemical, Pharmaceutical, Food, and Biotechnological Applications*. CRC Press.
- [Pantakar, 1980] Pantakar, S. V. (1980). *Numerical Heat Transfer and Fluid Flow*. McGraw-Hill Book Company.
- [ParaView, 2015] ParaView (2015). Paraview manual, a parallel visualization application, version 4.0.
- [Rautenbach and Melin, 2004] Rautenbach, R. and Melin, T. (2004). *Membranverfahren Grundlagen der Modul- und Anlagenauslegung*. Springer.
- [Robb, 1968] Robb, W. (1968). Thin silicone membranes - their permeation properties and some applications. *Annals of the New York Academy of Sciences, vol. 146, issue 1 Materials in* , pp. 119-137.
- [Rodrigues, 2009] Rodrigues, D. (2009). Model development of a membrane gas permeation unit for the separation of hydrogen and carbon dioxide. Master's thesis, TU Wien.
- [Rom et al., 2014] Rom, A., Wukovits, W., and Friedl, A. (2014). Development of a vacuum membrane distillation unit operation: From experimental data to a simulation model. *Chemical Engineering and Processing, 86, 90-95*.
- [Santafe-Moros and Gozalvez-Zafrilla, 2013] Santafe-Moros, A. and Gozalvez-Zafrilla, J. M. (2013). Numerical simulation of concentration polarization to estimate gypsum and calcium carbonate scaling on membrane surfaces. *Excerpt from the Proceedings of the COMSOL Conference in Rotterdam*.
- [Schlichting and Gersten, 2005] Schlichting, H. and Gersten, K. (2005). *Grenzschicht-Theorie*. Springer-Verlag Berlin Heidelberg.
- [Scott, 1996] Scott, K. (1996). *Handbook of Industrial Membranes*. Elsevier Science.
- [Sharafpoor et al., 2015] Sharafpoor, A., Mansourpour, Z., and Ghaee, A. (2015). Numerical study on concentration polarization for H₂-N₂ separation through a thin pd membrane by using computational fluid dynamics. *The 12th international conference on Membrane Science and Technology (MST 2015)*.
- [Staudacher, 2000] Staudacher, M. (2000). Fluidynamische Simulation des Stofftransportes in einem Dialysem modul. Master's thesis, TU Wien.
- [Staudacher et al., 2002] Staudacher, M., Harasek, M., Brinkmann, T., and Hilgendorff, W. (2002). Numerische Simulation der Strömung in einem Membrantestmodul. *Projektbericht ZID 02-166-2, Institut für Verfahrenstechnik, Umwelttechnik und Technische Biowissenschaften, TU Wien*.
- [Strathmann, 2011] Strathmann, H. (2011). *Introduction to Membrane Science and Technology*. Wiley-VCH, 97 98 edition.

- [Tahvildari et al., 2015] Tahvildari, K., Razavi, S. M. R., Tavakoli, H., Mashayekhi, A., and Golmohammadzadeh, R. (2015). Modeling and simulation of membrane separation process using computational fluid dynamics. *Arabian Journal of Chemistry*.
- [Turner, 2004] Turner, J. A. (2004). Sustainable hydrogen production. *Science* 305, 972-974.
- [Versteeg and Malalasekera, 1995] Versteeg, H. K. and Malalasekera, W. (1995). *An Introduction to Computational Fluid Dynamics. The Finite Volume Method*. Longman Scientific & Technical.
- [Wiley and Fletcher, 2002] Wiley, D. E. and Fletcher, D. F. (2002). Computational fluid dynamics modelling of flow and permeation for pressure-driven membrane processes. *Desalination* 145, 183-186.
- [Yampolskii and I. Pinnau, 2007] Yampolskii, Y. and I. Pinnau, B. F. (2007). *Materials Science of Membranes for Gas and Vapor Separation*. John Wiley & Sons, Ltd.
- [Zeynali et al., 2015] Zeynali, R., Ahmadnejgad, F., Ghasemazdeh, K., and Babalou, A. A. (2015). Investigation of palladium membrane reactor performance during ethanol steam reforming using CFD method. *The 12th international conference on Membrane Science and Technology (MST 2015)*.

List of symbols

Symbol	Unit	Name
a	mol m^{-3}	activity
A_m	m^2	membrane area
c	$\text{J mol}^{-1} \text{K}^{-1}$	heat capacity
C	mol m^{-3}	concentration
Δ	—	difference
d	m	characteristic length
d_{hyd}	m	hydraulic diameter
D	$\text{m}^2 \text{s}^{-1}$	diffusion coefficient
Da	u	Dalton
E_a	J mol^{-1}	activation energy
e	J	internal energy
f	N	force
φ	—	fugacity coefficient
Φ	—	dissipation function
γ	$\text{m}^3 \text{mol}^{-1}$	activity coefficient
μ	J mol^{-1}	chemical potential
J_ν	$\text{mol m}^{-2} \text{s}^{-1}$	volume flux
k	$\text{J mol}^{-1} \text{K}^{-1}$	Boltzmann constant
k	—	sorption coefficient
κ	$\Omega^{-1} \text{m}^{-1}$	electrical conductivity
K_i^H	$\text{mol m}^{-3} \text{Pa}$	Henry coefficient
L	m	characteristic length
L_i	$\text{m}^3 \text{m m}^{-2} \text{Pa}^{-1} \text{s}^{-1}$	transport coefficient
λ	$\text{W m}^{-1} \text{K}^{-1}$	heat conductivity
λ	m	distance
M	g mol^{-1}	molecular weight
μ	$\text{kg m}^{-1} \text{s}^{-1}$	dynamic viscosity
n	—	number
ν	$\text{m}^2 \text{s}^{-1}$	kinematic viscosity
p	Pa	pressure
P	$\text{m}^3 \text{m m}^{-2} \text{s}^{-1} \text{J}^{-1}$	permeability coefficient
Q	$\text{m}^3 \text{m}^{-2} \text{Pa}^{-1} \text{s}^{-1}$	permeance
r	m	radius
ρ	kg m^{-3}	density
R	$\text{J mol}^{-1} \text{K}^{-1}$	gas constant
\dot{q}	W m^{-2}	heat transfer
r_p	m	pore radius

Symbol	Unit	Name
T	K	temperature
τ	—	tortuosity factor
$\underline{\tau}$	N m^{-2}	viscous stress tensor
Θ	—	stage cut
u_{th}	m s^{-1}	thermal velocity
v	m s^{-1}	velocity
V_i	$\text{m}^3 \text{mol}^{-1}$	molar volume
\dot{V}_f	$\text{m}^3 \text{s}^{-1}$	volume flow component i
$w_{i,F}$	—	mass fraction component i in the feed
$w_{i,P}$	—	mass fraction component i in the permeate
x	—	directional coordinate
x_i, X_i	—	mole fraction component i
X	N m mol^{-1}	pressure
y_i	—	mole fraction component i in the product
z	m	directional coordinate

List of Figures

1	Membrane separation principle, adapted from [Rautenbach and Melin, 2004]	6
2	Membrane classification, [Mulder, 1996]	7
3	Symmetric (left) and asymmetric membrane (right) [Rautenbach and Melin, 2004]	7
4	Hollow fiber membrane [Scott, 1996]	9
5	Flow patterns (left), cross flow (top), co-flow (middle), counter-flow (bottom), concentration gradient (right) [Baker, 2004]	13
6	Course of the concentration [Rautenbach and Melin, 2004]	14
7	Transport of gases through membranes [Baker, 2004]	16
8	Relation Knudsen to Poiseuille flow [Baker, 2004]	18
9	molecular sieving (top), surface diffusion (middle), capillary condensation (bottom) [Rautenbach and Melin, 2004]	19
10	Solution diffusion model, adapted from [Strathmann, 2011]	20
11	Discretization of the computational domain with grids, [Ferziger and Peric, 2002]	25
12	Finite volume, [Bird et al., 2002]	29
13	Membrane module with 5 exits, scaled 1:10	36
14	Membrane module with 7 fibers, [Haddadi et al., 2015a], [Haddadi et al., 2015b]	37
15	Single fiber membrane module	40
16	Refinement level 0, level 1, level 2, level 3	41
17	Mesh of the single fiber membrane	42
18	Membrane module as depicted in the data sheet	45
19	Actual membrane module	46
20	Technical drawing of the module (units in mm)	47
21	BlockMesh around hollow fiber membrane	48
22	Mesh in the concentrate and <i>permeateDomain</i>	48
23	Membrane cell ACM, [Lassmann, 2015]	51
24	Membrane model, [Lassmann, 2015]	52
25	ACM validation carbon dioxide, [Lassmann, 2015]	54
26	ACM validation hydrogen, [Lassmann, 2015]	55
27	Comparison CFD and ACM	56
28	Comparison of composition at different stage cuts for CFD and ACM, [Haddadi et al., 2015a], [Haddadi et al., 2015b]	58
29	Comparison of retentate mass flow at different stage cuts for CFD and ACM	59
30	Comparison of permeate mass flow at different stage cuts for CFD and ACM	60
31	Concentration gradient in the central fiber	61
32	Concentration of methane in the cross section of the fiber	62
33	Course of residuals	64
34	Course of scaled mass balance difference and velocities, [Haddadi et al., 2015a], [Haddadi et al., 2015b]	66
35	Residuals LTSMembraneFoam	69
36	Velocity profile	70
37	Pressure drop in the 7 fiber module, [Haddadi et al., 2015a], [Haddadi et al., 2015b]	72
38	Velocity in the 7 fiber module, [Haddadi et al., 2015a], [Haddadi et al., 2015b]	73
39	Carbon dioxide in the 7 fiber module, [Haddadi et al., 2015a], [Haddadi et al., 2015b]	74
40	Methane in the 7 fiber module	75
41	Comparison of different velocities for co- and counter-current flow	78
42	Comparison of different transmembrane pressures for co and counter-current flow	80
43	Permeate composition and pressure drop at different permeances of hydrogen	82
44	Permeate composition and pressure drop at different permeances of all species	84

45	Concentration of carbon dioxide in the shell of the seven fiber module, scaled 1:10	xvi
46	Concentration of carbon dioxide in the fibers of the seven fiber module, scaled 1:10	xvii
47	Vector plot of velocity in the shell of the 30 fiber module with central exit, scaled 1:10	xviii

List of Tables

1	Membrane separation processes, structures, materials and driving forces [Strathmann, 2011]	22
2	Server hardware	35
3	Feed composition	37
4	Input parameters for test cases	38
5	Validation cases CFD and ACM	39
6	Standard values for single fiber membrane	40
7	Standard values for single fiber membrane	42
8	Testcases single fiber	44
9	Permeabilities of Polydimethylsiloxane [Robb, 1968]	45
10	Dimensions of the PDMS membrane module	46
11	Operating conditions	47
12	Input parameters for the simulations	49
13	Variations for the validation of ACM model with algorithm, [Lassmann, 2015]	53
14	Comparison ACM model and algorithm, [Lassmann, 2015]	53
15	Error for validation CFD and ACM of mass flow in [%]	59
16	Comparison coarse mesh and graded mesh	61
17	Input parameters ACM and CFD	63
18	Convergence of validation simulations	67
19	Comparison LTSMembraneFoam and membraneFoam	68
20	Mass balance and stage cut	76
21	Permeate composition for different feed velocities and co-current flow	77
22	Permeate composition for different feed velocities and counter-current flow	77
23	Permeate composition for different TMP and co-current flow	79
24	Permeate composition for different TMP and counter-current flow	79
25	Permeate composition for different feed velocities and higher permeances	81
26	Permeate composition for different permeances of hydrogen	82
27	Permeate composition for different permeances of all species	83
28	Mass flows from lab experiments	85
29	Pressures from lab experiments	85
30	Permeances of species from lab experiments	86
31	Mass flows from the CFD simulations	86

Acknowledgements

Mein aufrichtiger Dank gilt allen, die zum Zustandekommen dieser Diplomarbeit beigetragen haben.

Ich danke meinen Eltern Doris Schretter und Reinhard Schretter für die finanzielle Unterstützung.

Von der Forschungsgruppe Thermische Verfahrenstechnik und Fluidodynamische Simulation am Institut für Verfahrenstechnik der TU Wien danke ich meinen Betreuern Michael Harasek, Christian Jordan und Bahram Haddadi Sisakht für die wertvolle intellektuelle Unterstützung.

Ich danke Martin Miltner für die Messungen des 30-Faser-Membranmoduls im Labor.

Ich danke meiner Kommilitonin Liisa-Maria Marin für die spezielle Unterstützung.

Ich danke meinen Freunden und Freundinnen für die sexuelle Unterstützung.

Ich danke Gott und Jesus für die spirituelle Unterstützung.



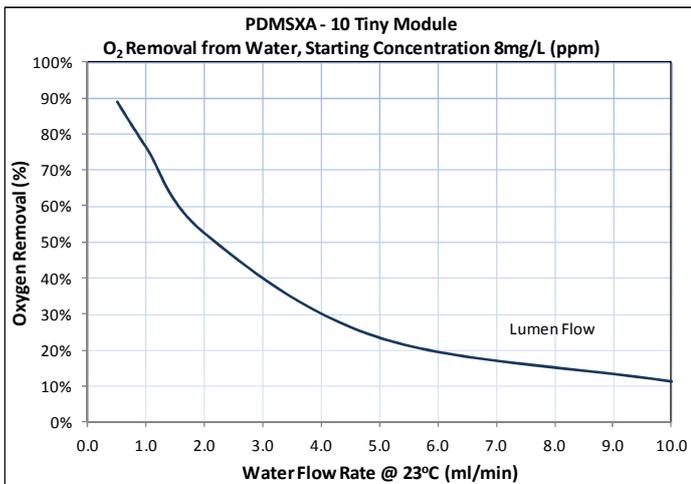
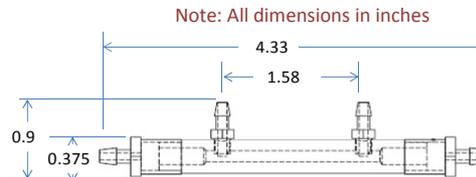
Module Characteristics

Membrane Material	PDMS (Silicone)	
Membrane Type	Dense Hollow Fiber	
Fiber ID	μm (in)	190 (0.00748)
Fiber OD	μm (in)	300 (0.0118)
Fiber Wall Thickness	μm (in)	55 (0.0022)
Fiber Count	#	30
Membrane Area¹	cm ² (ft ²)	10 (0.011)
Module Length	cm (in)	10.9 (4.3)
Module Diameter	cm (in)	0.95 (0.375)
Fittings / Connection Size	in	Barbed 1/8
Shell / End Caps	Polycarbonate	
Fittings Material	Polycarbonate	
Potting Material	Polyurethane	
Other Materials of Construction²	Acrylic	

Operating Conditions

Max Continuous Operating Temperature	$^{\circ}\text{C}$ ($^{\circ}\text{F}$)	60 (140)
Max Shell Side Pressure	bar (psi)	3 (45) @ 77 $^{\circ}\text{F}$
Max Lumen Pressure	bar (psi)	5 (75) @ 77 $^{\circ}\text{F}$
Max TMP³ Shell to Lumen	bar (psi)	1 (15) @ 77 $^{\circ}\text{F}$
Max TMP³ Lumen to Shell	bar (psi)	3 (45) @ 77 $^{\circ}\text{F}$
Typical Liquid Flow Rate	l/min (gpm)	0.001 - 0.01 (0.00026 - 0.0026)
Typical Gas Flow Rate	scfm (slpm)	0.00004 - 0.004 (0.001 - 0.1)

- 1- Effective Membrane Area
- 2- Traces of PVOH (polyvinyl alcohol) possible in lumens
- 3- TMP (Transmembrane Pressure)

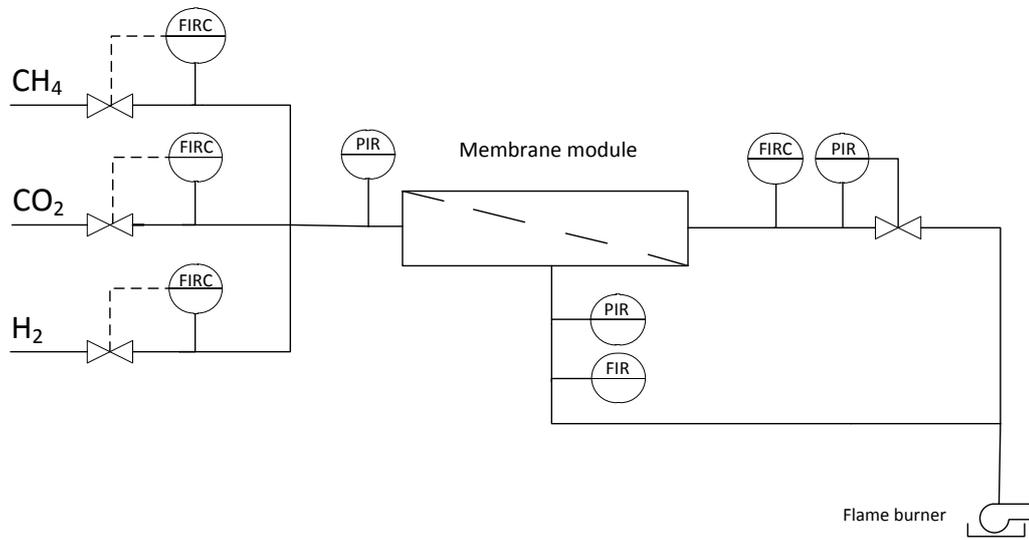


This product is to be used only by persons familiar with its use. It is the responsibility of the user to determine the suitability of PermSelect® membrane modules in its specific application. Membrane modules can fail, permitting fluid discharge into the environment and mixing of shell and tube side fluids. **Users must take all precautions to ensure safety to people and property in case of module failure.** Purchaser assumes all responsibility for the suitability and fitness for use as well as for the protection of the environment and for health and safety involving this product. Seller reserves the right to modify this document without prior notice. Check our web site to verify the latest update. To the best of our knowledge the information contained herein is accurate. However, neither MedArray, Inc. nor any of its affiliates assumes any liability whatsoever for the accuracy or completeness of the information contained herein. Final determination of the suitability of any material and whether there is any infringement of patents, trademarks, or copyrights is the sole responsibility of the user. Users of any substance with PermSelect® membrane modules should satisfy themselves by independent investigation that the material can be used safely. We may have described certain hazards, but we cannot guarantee that these are the only hazards that exist.

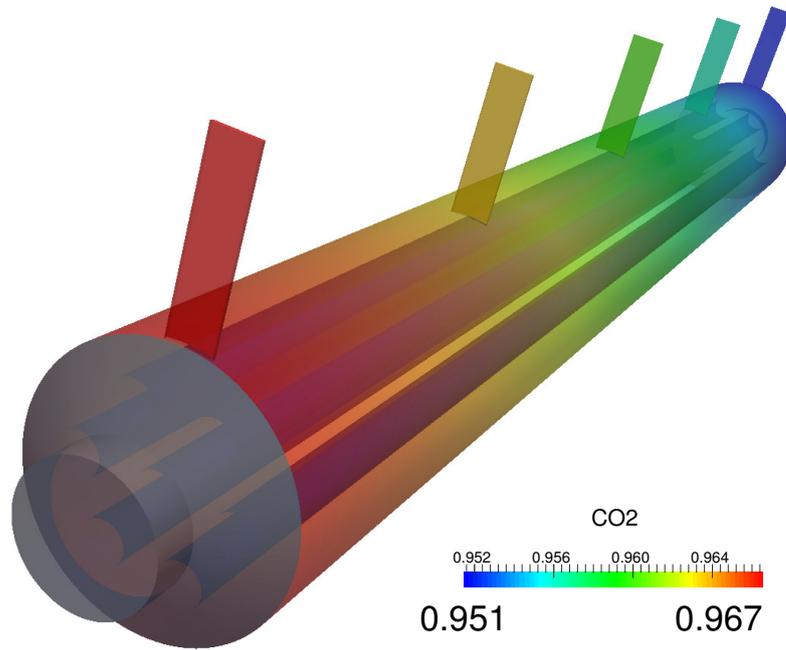
PermSelect® is a registered trademark of MedArray, Inc., 3915 Research Park Dr., Ste. A-4, Ann Arbor, MI 48108; +1 (734) 769-1066; www.permselect.com. PermSelect® membranes and membrane modules are protected by US and international patents.

PDMSXA-10-13-2

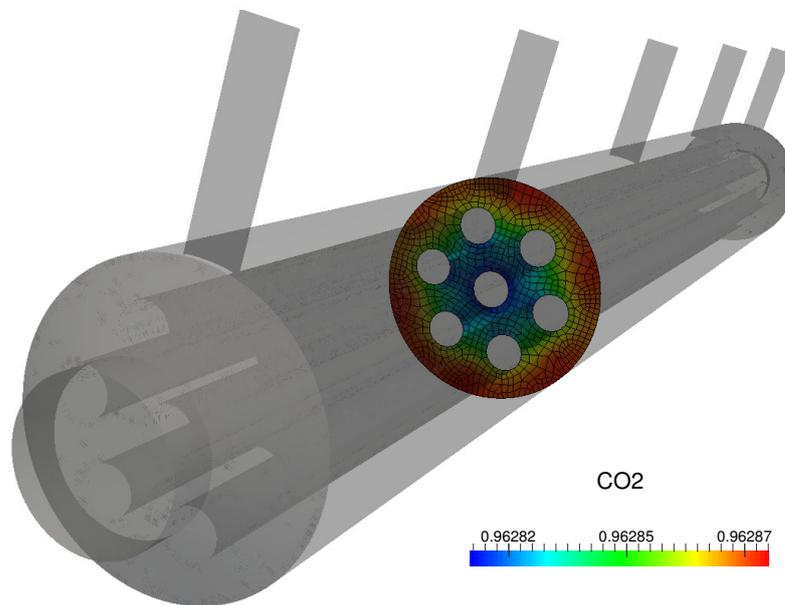
Flowsheet for laboratory measurement setup



Plots from CFD simulation

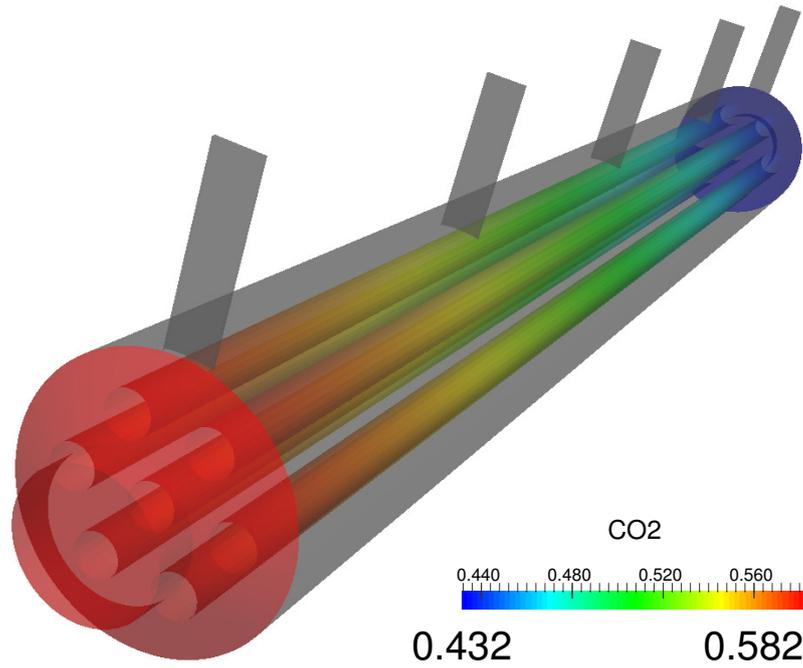


(a) Concentration along the shell

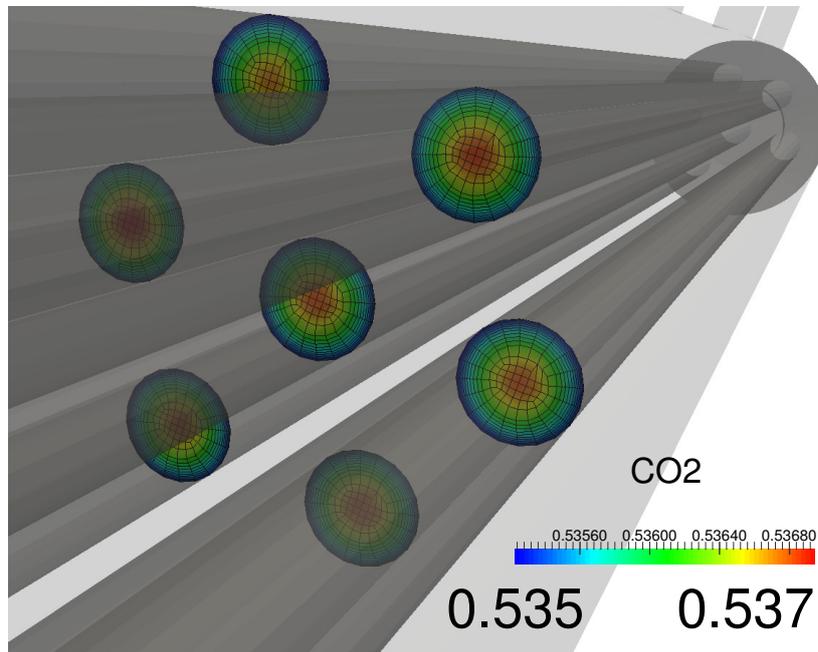


(b) Concentration in the cross section of the shell

Figure 45: Concentration of carbon dioxide in the shell of the seven fiber module, scaled 1:10

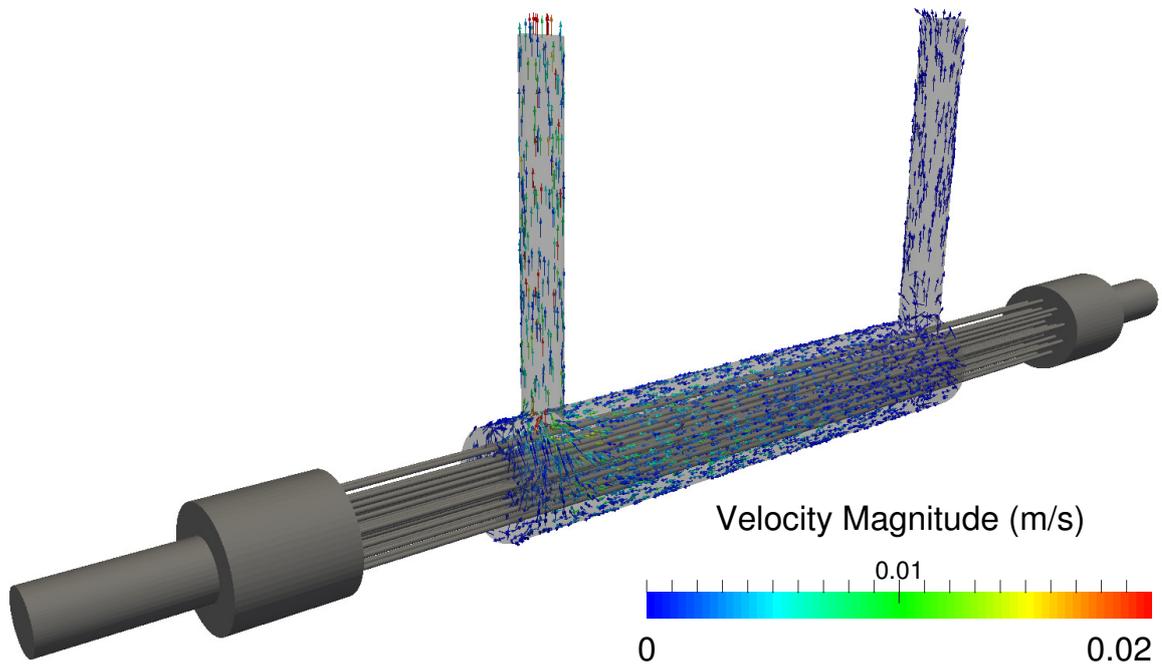


(a) Concentration along the fibers

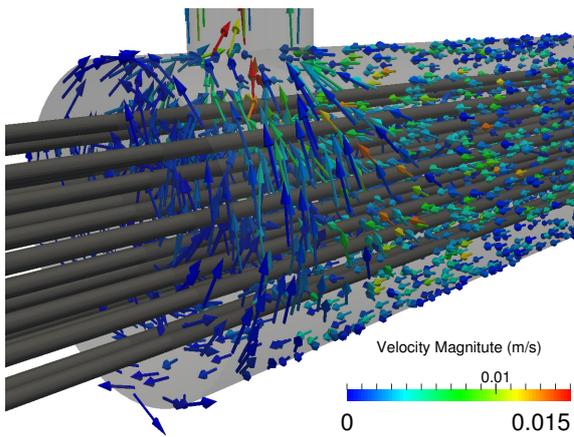


(b) Concentration in the cross section of the fibers

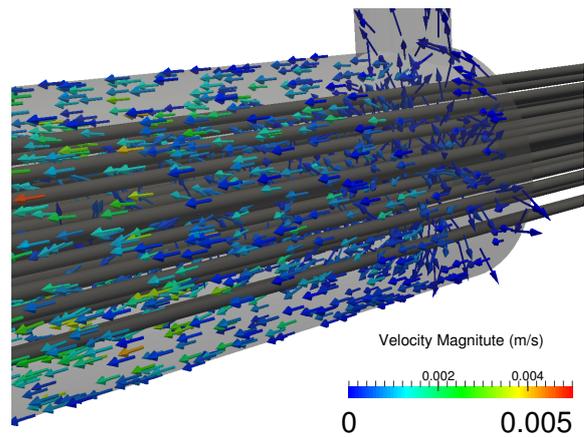
Figure 46: Concentration of carbon dioxide in the fibers of the seven fiber module, scaled 1:10



(a) Velocity in the shell



(b) Velocity in the outlet area



(c) Velocity on the right end

Figure 47: Vector plot of velocity in the shell of the 30 fiber module with central exit, scaled 1:10

Case setup

Case setup of the 30 fiber membrane module

- 0

- **concentrateZone**
 - * CO2
 - * H2
 - * CH4
 - * p
 - * T
 - * U
- **permeateZone**
 - * CO2
 - * H2
 - * CH4
 - * p
 - * T
 - * U

- constant

- **concentrateZone**
 - * **polyMesh**
 - boundary
 - boundaryReionAddressing
 - cellRegionAddressing
 - faceRegionAdressing
 - faces
 - faceZones
 - neighbour
 - owner
 - pointRegionAddressing
 - points
 - pointZones
 - * chemistryProperties
 - * combustionProperties
 - * g
 - * radiationProperties
 - * RASProperties
 - * reactions
 - * thermo.compressibleGas
 - * thermophysicalProperties
 - * turbulenceProperties
- **permeateZone**
 - * **polyMesh**
 - boundary
 - boundaryReionAddressing
 - cellRegionAddressing
 - faceRegionAdressing
 - faces
 - faceZones
 - neighbour
 - owner
 - pointRegionAddressing
 - points
 - pointZones
 - * chemistryProperties
 - * combustionProperties
 - * g
 - * radiationProperties
 - * RASProperties
 - * reactions
 - * thermo.compressibleGas
 - * thermophysicalProperties
 - * turbulenceProperties
- **triSurface**
 - * concentrateDomain.eMesh
 - * concentrateDomain.stl
 - * feedinlet.stl
 - * outletone.stl
 - * outlettwo.stl
 - * permeateDomain.eMesh
 - * permeateDomain.stl
 - * pottingleftfiberwall.stl
 - * pottingrightfiberwall.stl
 - * retentateoutlet.stl
- cellToRegion
- membraneProperties
- regionProperties

- system

- **concentrateZone**
 - * decomposeParDict
 - * fvSchemes
 - * fvSolution
- **permeateZone**
 - * decomposeParDict
 - * fvSchemes item fvSolution
- controlDict
- decomposeParDict
- fvSchemes
- fvSolution
- meshQualityDict
- snappyHexMeshDict
- surfaceFeatureExtractDict

system/controlDict

```
// ***** //  
application    LTSMembraneFoam;  
  
startFrom     latestTime;  
  
startTime     0;  
  
stopAt        endTime;  
  
endTime       10000;  
  
deltaT        1;  
  
writeControl   runTime;  
  
writeInterval  100;  
  
purgeWrite     0;  
  
writeFormat    ascii;  
  
writePrecision 8;  
  
writeCompression off;  
  
timeFormat     general;  
  
timePrecision  6;  
  
runTimeModifiable yes;  
  
maxDeltaT      1e-2;  
  
maxCo          1;  
  
alphaTemp      0.05;  
  
rDeltaTSmoothingCoeff 1;  
  
rDeltaTDampingCoeff 1;  
  
convergencyCheck yes;  
  
tolerance      1e-6;  
  
libs  
(  
"libmembraneBoundaryConditions.so"  
);  
  
// ***** //
```

system/fvSchemes

```
// ***** //  
ddtSchemes  
{  
}  
  
gradSchemes  
{  
}  
  
divSchemes  
{  
}  
  
laplacianSchemes  
{  
}  
  
interpolationSchemes  
{  
}  
  
snGradSchemes  
{  
}  
  
fluxRequired  
{  
}  
  
// ***** //
```

system/fvSolution

```
// ***** //  
  
PIMPLE  
{  
  nOuterCorrectors 1;  
}  
  
// ***** //
```

system/concentrateZone/fvSchemes and system/permeateZone/fvSchemes

```
// ***** //  
ddtSchemes  
{  
  default    localEuler rDeltaT;  
}  
  
gradSchemes  
{  
  default    Gauss linear;  
}
```

```

}

divSchemes
{
default    none;

div(phi,U)    Gauss limitedLinearV 1;
div(phi,Yj_h) Gauss limitedLinear 1;
div(phi,K)    Gauss limitedLinear 1;
div(phi,h)    Gauss limitedLinear 1;
div(phi,k)    Gauss limitedLinear 1;
div(phi,epsilon) Gauss limitedLinear 1;
div(phi,R)    Gauss limitedLinear 1;
div(R)        Gauss linear;
div((muEff*dev2(T(grad(U)))))) Gauss linear;
}

```

```

laplacianSchemes
{
default    Gauss linear orthogonal;
}

```

```

interpolationSchemes
{
default    linear;
}

```

```

snGradSchemes
{
default    orthogonal;
}

```

```

fluxRequired
{
default    no;
p_rgh;
}

```

```

// ***** //

```

system/concentrateZone/fvSolution and system/permeateZone/fvSolution

```

// ***** //

```

```

solvers
{
"rho.*"
{
solver    diagonal;
// solver    PCG;
// preconditioner    DIC;
// tolerance    1e-5;
// relTol    0.1;
}
}

```

```

p_rgh
{
solver    GAMG;
tolerance 1e-7;
relTol   0.01;
smoother  GaussSeidel;
cacheAgglomeration true;
nCellsInCoarsestLevel 10;
agglomerator faceAreaPair;
mergeLevels 1;
maxIter   10000;
}

p_rghFinal
{
$p_rgh;
tolerance 1e-7;
relTol   0;
}

"(U|h|k|epsilon|R)"
{
solver    PBiCG;
preconditioner DILU;
tolerance 1e-6;
relTol   0.1;
}

"(U|h|k|epsilon|R)Final"
{
$U;
relTol   0.1;
}

Yj
{
$U;
relTol   0.1;
}

PIMPLE
{
momentumPredictor no;
nCorrectors 1;
nNonOrthogonalCorrectors 0;
}

relaxationFactors
{
fields
{
}
equations
{
"h.*" 1;
"U.*" 1;
}
}

```

```
// ***** //
```

system/meshQualityDict

```
// ***** //
```

```
// Include defaults parameters from master dictionary  
#include "$WM_PROJECT_DIR/etc/caseDicts/meshQualityDict"
```

```
//- minFaceWeight (0 -> 0.5)  
minFaceWeight 0.02;
```

```
// ***** //
```

system/snappyHexMeshDict

```
// ***** //
```

```
castellatedMesh true;  
snap true;  
addLayers false;
```

geometry

```
{  
concentrateDomain.stl  
{  
type triSurfaceMesh;  
name concentrate;  
}
```

```
permeateDomain.stl  
{  
type triSurfaceMesh;  
name permeate;  
}
```

```
feedinlet.stl  
{  
type triSurfaceMesh;  
name feedinlet;  
}
```

```
retentateoutlet.stl  
{  
type triSurfaceMesh;  
name retentateoutlet;  
}
```

```
outletone.stl  
{  
type triSurfaceMesh;  
name outletone;  
}
```

```
outlettwo.stl  
{  
type triSurfaceMesh;  
name outlettwo;  
}
```

```
};
```

```

castellatedMeshControls
{
maxLocalCells 2000000;
maxGlobalCells 90000000;
minRefinementCells 0;
maxLoadUnbalance 0.1;
nCellsBetweenLevels 1;

features
(
{
file "concentrateDomain.eMesh";
level 3;
}
{
file "permeateDomain.eMesh";
level 3;
}
);

refinementSurfaces
{
concentrate
{
level (3 3);
faceZone concentrateFaces;
cellZone concentrateZone;
cellZoneInside insidePoint;
insidePoint (-0.0161 0 0);
}
permeate
{
level (3 3);
faceZone permeateFaces;
cellZone permeateZone;
cellZoneInside insidePoint;
insidePoint (0.0211 0 0.00181);
}
feedinlet
{
level (0 0);
patchInfo
{
type patch;
}
}
retentateoutlet
{
level (0 0);
patchInfo
{
type patch;
}
}
outletone
{
level (0 0);
patchInfo
{
type patch;
}
}

```

```

}
outlettwo
{
level (0 0);
patchInfo
{
type patch;
}
}
}
resolveFeatureAngle 30;
refinementRegions
{
}
locationInMesh (0.0251 0 0);
allowFreeStandingZoneFaces false;
}

snapControls
{

nSmoothPatch 20;
tolerance 1.3;
nSolveIter 600;
nRelaxIter 20;
nFeatureSnapIter 5;
implicitFeatureSnap false;
explicitFeatureSnap true;
multiRegionFeatureSnap true;
}
addLayersControls
{
relativeSizes false;
layers
{
concentrateZone_to_pottingLeftZone
{
nSurfaceLayers 2;
}
concentrateZone_to_permeateZone
{
nSurfaceLayers 2;
}
concentrateZone_to_pottingRightZone
{
nSurfaceLayers 2;
}
}
}
expansionRatio 1.3;
finalLayerThickness 0.00002;
minThickness 0.00001;
nGrow 0;
featureAngle 85;
slipFeatureAngle 25;
nRelaxIter 5;
nSmoothSurfaceNormals 4;
nSmoothNormals 3;
nSmoothThickness 10;
maxFaceThicknessRatio 0.5;
maxThicknessToMedialRatio 0.2;
minMedianAxisAngle 90;

```

```

nBufferCellsNoExtrude 0;
nLayerIter 10 ;
}
meshQualityControls
{
#include "meshQualityDict"
nSmoothScale 4;
errorReduction 0.75;
}
writeFlags
(
scalarLevels
layerSets
layerFields
);
mergeTolerance 1e-6;

// ***** //

```

system/surfaceFeatureExtractDict

```

// ***** //
permeateDomain.stl
{
extractionMethod extractFromSurface;
extractFromSurfaceCoeffs
{
includedAngle 150;
}
subsetFeatures
{
nonManifoldEdges no;
openEdges yes;
}
writeObj no;
}

concentrateDomain.stl
{
extractionMethod extractFromSurface;
extractFromSurfaceCoeffs
{
includedAngle 150;
}
subsetFeatures
{
nonManifoldEdges no;
openEdges yes;
}
writeObj no;
}

// ***** //

```