



# Membrane modeling using CFD: Combined evaluation of mass transfer and geometrical influences in 1D and 3D



Bahram Haddadi\*, Christian Jordan, Martin Miltner, Michael Harasek

Institute of Chemical, Environmental & Bioscience Engineering, TU Wien, Getreidemarkt 9/166, 1060 Vienna, Austria

## ARTICLE INFO

### Keywords:

Gas permeation  
Multicomponent separation  
Computational Fluid Dynamics  
Process simulation  
Module performance

## ABSTRACT

In current literature two main approaches are used for the simulation of membrane contactors. One route considers membrane modules only in 1D for process simulation applications, the other route focuses on 3D simulation of modules using Computational Fluid Dynamics to provide very detailed information about membrane mass transfer or geometrical influences on the module performance.

A new CFD algorithm is introduced in the current work. It is capable of performing both 3D and 1D simulations using the same code – 1D to be used in fast process simulation applications whereas the 3D method can be applied for fully resolved CFD applications. Using experimental results from pure gas permeation of a hollow fiber module, it was demonstrated that 1D and 3D simulations compare with less than 2% deviation on a global scale. Based on the 3D simulations, it was found that the arrangement of the fibers can lead to high velocity zones close to the module walls. It was demonstrated that the 1D CFD method performs well even for almost pure gases like CH<sub>4</sub> at retentate side, by running simulations of a pilot scale biogas separation module in co- and counter-current configurations.

## 1. Introduction

Efficient design of membrane units helps to improve separation performance and to decrease the energy demand of the membrane processes. This is not possible without having a comprehensive insight into the underlying phenomena and it requires also performing sensitivity analysis of the membrane systems [1].

There are different approaches for studying membranes, e.g. experimental studies (lab or pilot scale) or simulation approaches. Experimental studies like permeation measurements on membrane module level do not provide data on the inner flow structures of a module or local effects as the concentration gradients along a membrane surface. Of course it is possible to conduct experiments with modules specifically constructed or equipped with sampling ports for e.g. local flow measurements using optical methods [2] or concentration or pressure measurements however, they are usually expensive, in some cases hard to perform and mostly provide point data which are not sufficient for full understanding of the system. Compared to experimental studies, simulations are usually much easier and cheaper to perform and also provide spatially resolved data for the whole geometry [3].

Membrane modeling using process simulation approaches (1D) can provide a reasonable insight into membrane processes for design and optimization of these systems on a plant scale and it has been used in various studies [4–10]. 2D simulations have been also performed for both simulation of membrane processes and also individual membrane units [11–13]. But 2D models are usually too time consuming for process simulation purposes and they also miss the detailed representation of the membranes compare to 3D simulations [14,15]. Full 3D discretization gives a very good insight into the simulated process (to observe e.g. boundary layer effects like concentration polarization or mixing promoters like spacers [16]). Using 3D simulation data mass transfer or pressure drop correlations or relations can be derived to be applied in the less detailed simulation methods e.g. black box and low resolution modeling [17]. Although in the recent years the increase in computers computational power has provided a very powerful infrastructure for numerical solutions [18,19], fully resolved simulations still need significant computational effort (simulation time demand and hardware availability) due to the geometric complexity and size of industrial membrane modules. Computational Fluid Dynamics (CFD) or numerical tackling of fluid flow, heat and mass transfer in the fluids is a 3D simulation approach, which can provide detailed temporal and

Abbreviations: CFD, Computational Fluid Dynamics; MPI, message passing interface; GPL, GNU public license

\* Corresponding author.

E-mail addresses: [bahram.haddadi.sisakht@tuwien.ac.at](mailto:bahram.haddadi.sisakht@tuwien.ac.at) (B. Haddadi), [christian.jordan@tuwien.ac.at](mailto:christian.jordan@tuwien.ac.at) (C. Jordan), [martin.miltner@tuwien.ac.at](mailto:martin.miltner@tuwien.ac.at) (M. Miltner), [michael.harasek@tuwien.ac.at](mailto:michael.harasek@tuwien.ac.at) (M. Harasek).

<https://doi.org/10.1016/j.memsci.2018.05.040>

Received 25 December 2017; Received in revised form 18 May 2018; Accepted 20 May 2018

Available online 24 May 2018

0376-7388/ © 2018 Elsevier B.V. All rights reserved.

Nomenclature			
$C_p$	heat capacity [J/(kg K)]	x	length [m]
$D_{AB}$	diffusion coefficient [ $m^2/s$ ]	Y	mass fraction [kg/kg]
E	activation energy [J/mol]	<i>Greek letters</i>	
$f_D$	friction factor [-]	$\rho$	density [ $kg/m^3$ ]
K	thermal conductivity [W/(m K)]	$\Pi$	permeance [ $m_{STP}^3/(m^2 s Pa)$ ]
p	pressure [Pa]	$\mu$	dynamic viscosity [Pa s]
R	universal gas constant [J/(mol K)]	$\nu$	kinematic viscosity [ $m^2/s$ ]
Re	Reynolds number [-]	<i>Subscripts</i>	
$S_e$	heat source term [J/( $m^3 s$ )]	i	species i
$S_m$	mass source term [ $kg/(m^3 s)$ ]	P	permeate
t	time [s]	R	retentate
T	temperature [K]	ref	reference value
$\mathbf{u}$	velocity vector [m/s]		
$\bar{u}$	average velocity [m/s]		

spatial data. As for other numerical solutions, in CFD problems the relevant physical domain should be discretized in time and space. Therefore, CFD simulations are performed on computational grids [20]. CFD relies on physical models to provide the desired insight to the phenomena therefore, it needs to be combined with experimental studies for support and validation of these models. Membrane simulations using CFD can be mainly categorized in hydrodynamic studies and mass transfer studies. On the one hand, in hydrodynamic studies, mostly turbulence promoters and geometry modifications for creation of secondary flows have been investigated. On the other hand mass transfer studies mostly focused on the region in the vicinity of the membrane surface and are mostly limited to laminar flow regime while excluding the design complexities of the modules [14,21–23].

The above literature survey shows that much effort has been made on the modeling of membrane modules. 1D approaches are mostly appropriate for process simulation investigations and by using appropriate models (e.g. concentration polarization, pressure loss etc.) also for module level investigations [24,25] while 3D modeling approaches are better for detailed insight into the membrane module. Usually using 2D approaches for detailed investigations are not very promising because they cannot provide the same details (and sometimes wrong predictions – if the phenomena is 3D) as 3D simulations and in the case of process simulation they can provide almost the same details at the cost of slower simulations compared to 1D [15]. Although the hydrodynamics and mass transfer are inseparable and physically closely linked, in many of the 3D approaches just one of the effects is considered. In some studies just the geometric effects are considered without considering the change in the flow rate because of transmembrane flux. This might lead to wrong hydrodynamics predictions in case of high transmembrane fluxes. On the other hand considering just the transmembrane fluxes without hydrodynamics can also result in misleading conclusions, e.g. ignoring the concentration polarization close to membrane surface and the changes in the concentration layer because of the geometric effects caused by spacers or mixing promotion devices. Furthermore all approaches described so far can just operate in 1D or 3D as the chosen approaches cannot handle both, which might be interesting to give the user the flexibility to be able to choose between the details and speed using the same code or software package.

In this study, a new CFD algorithm for modeling membrane modules was developed which can also be operated in 1D mode for performing process simulation modeling. The new algorithm was applied to the simulation of a fully resolved module considering multi-component gas permeation through a hollow fiber membrane module. The same solver code was also applied to a more complex hollow fiber module by setting it to a one-dimensional mode to overcome the drawback of high computational effort. The results were compared to experimental data showing good agreement, which proved the capability of the suggested

algorithm.

## 2. Methodology

In this study, a new algorithm for CFD modeling of membrane separation is suggested. This new algorithm is based on a multi-region approach which makes it capable of detailed modeling of hydrodynamic behavior of both sides of the membrane (retentate and permeate). The algorithm also covers modeling of the trans-membrane flux between retentate and permeate for multi-component separation by providing a generic platform for implementation of different mass transfer models. It also includes generic per-region turbulence modeling and the capability to switch between 1D and full detail 3D membrane modeling. The suggested algorithm was implemented in the open-source platform OpenFOAM® [26] (version 4.1, 2016) with solution-diffusion model (mainly for gas permeation) as mass transfer mechanism. The developed solver is capable of handling different phases (liquid, gas) at both sides of the membrane, e.g. gas permeation [27] (gas – gas) or pervaporation [28] (gas – liquid).

Two different membrane modules were simulated using the new code: The first geometry is a small hollow fiber membrane module (with 30 fibers, Area  $\sim 10 cm^2$ ) for gas permeation. Pure gas measurements were performed in the lab on the module using three different gases. The module was simulated at the same operating conditions in 1D and 3D configurations and the results were compared.

The second membrane module was a pilot scale gas permeation membrane module with 800 fibers (Area  $\sim 0.38 m^2$ ) which was used for separation of multi-component biogas mixtures. Since the module was too big for detailed 3D simulation it was only simulated using the 1D approach. The results were compared to experimental data available in literature.

## 3. Model development

There are various CFD tools available commercially and for free. OpenFOAM® is a very promising free and open-source (released under GNU license – GPL, version 3, 2007) CFD package written in C++ which has been being used and improved during time by lots of users and core developers. Since it is open-source it gives the user the flexibility to implement new models and algorithms and also modify and optimize available models and algorithms for special purposes. The OpenFOAM® code consists of a collection of official and contributed/integrated solvers and libraries. An appropriate solver for a given simulation task needs to be selected based on the required physical models, and since the package is object oriented, additional libraries can be linked to it. OpenFOAM® also benefits from the MPI parallelization which can simulate complex geometries with highly resolved

details [29].

The new algorithm for modeling of membranes was implemented into OpenFOAM® community edition, version 4.1 (2016). The membrane algorithm relies on solving of multiple conservation equations including momentum, mass and energy for each membrane region (retentate and permeate). Coupling between these regions is achieved by integrated mass and energy source terms in the relevant equations. In the following, the details of model equations and the full algorithms for wrapping these equations are presented.

### 3.1. Momentum and continuity

Pressure and velocity for a compressible fluid are calculated using the non-linear implicitly coupled Navier-Stokes and continuity equations [30].

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (1)$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot \rho \mathbf{u} \mathbf{u} = -\nabla p + \nabla \cdot \mu (\nabla \mathbf{u} + \nabla^T \mathbf{u}) \quad (2)$$

where  $\rho$  [kg/m<sup>3</sup>] is the density,  $p$  [Pa] is the pressure  $\mu$  [Pa s] is the viscosity and  $\mathbf{u}$  [m/s] is the fluid velocity. Both pressure and velocity are unknown in solution of fluid flow and they need to be solved simultaneously. A well-known approach for solving Navier-Stokes and continuity equations is the pressure implicit with splitting of operator (PISO) algorithm which is suitable for non-iterative transient solution of compressible and incompressible flows [30,31] but it can also be applied to steady state flows [32].

In the PISO algorithm an intermediate velocity field is calculated by solving the momentum equations and based on the derived velocity field all cell face mass fluxes are updated before the first pressure correction equation is solved. Based on the pressure corrections, velocities and cell face mass fluxes are updated and used in the second pressure correction equation for calculation of new, improved pressure corrections [32]. After correcting pressure and velocity field, other transport equations (e.g. species, energy and turbulence equations) are solved and the time is increased for the next loop. If the calculations are performed in the transient mode time step size is controlled dynamically using a maximum Courant number to ensure a stable and convergent solution [33].

$$\text{Courant number } Co = \frac{u \Delta t}{\Delta x} \quad (3)$$

where  $u$  [m/s] is the velocity magnitude,  $\Delta t$  [s] is the time step size and  $\Delta x$  [m] is the length interval (mesh size). Diffusion terms may be corrected based on the predicted turbulent diffusion coefficient calculated using applied turbulence model. However, since the Reynolds numbers are low, in this study all simulations were performed in laminar conditions.

### 3.2. Energy equation

Heat transfer across system boundary and also within the system was described by the general heat transfer equation.

$$\frac{\partial \rho C_p T}{\partial t} + \nabla \cdot \rho C_p T \mathbf{u} = \nabla \cdot k (\nabla T) + S_e \quad (4)$$

$T$  [K] is the temperature,  $k$  [W/(mK)] is the thermal conductivity and  $C_p$  [J/(kg K)] is the medium heat capacity.  $S_e$  [J/(m<sup>3</sup> s)] is the heat source term and is calculated according to the phenomenon occurring in the fluid.  $S_e$  can be a volumetric source term e.g. the heat of reactions for membrane reactors or it can be a surface heat source, which has non-zero values just at the relevant boundaries, e.g. the latent heat of evaporation for pervaporation, or the Joule-Thomson effect for large pressure differences across the membrane for certain gases [34]. In this study both, the small and the pilot scale modules the gas permeation

processes were performed at rather low pressures, therefore  $S_e$  was considered to be zero. Density and compressibility effects are calculated based on the equation of state which will be covered in more detail in Section 3.4.

### 3.3. Species transport equation

Species transport was solved using the species conservation equation:

$$\frac{\partial \rho Y_i}{\partial t} + \nabla \cdot \rho Y_i \mathbf{u} = \nabla \cdot D_{AB} (\nabla Y_i) + S_{mi} \quad (5)$$

where  $Y_i$  [kg/kg] is the mass fraction of the species  $i$  and  $D_{AB}$  [m<sup>2</sup>/s] is the diffusion coefficient.  $S_{mi}$  [kg/(m<sup>3</sup> s)] is the mass transfer source term for species  $i$ , which is calculated based on the phenomena happening in the fluid. If just transmembrane flux is considered,  $S_{mi}$  is zero everywhere except on the membrane boundaries. The  $S_{mi}$  value on the membrane boundaries is calculated based on the mass transfer mechanism relevant for the membrane type and process considered in the application.

### 3.4. Physical and transport properties

Various physical and transport properties models in OpenFOAM® can be selected at runtime. Among available gas models, the “Ideal gas” approximation is suitable, since biogas processing is performed at rather low pressures [35] (< 10<sup>6</sup> Pa, which is far below the critical point for the selected gas components). Hence the ideal gas equation was used:

$$\rho = p/(RT) \quad (6)$$

where  $R$  [J/(mol K)] is the universal gas constant. The ideal gas viscosity was modeled using Sutherland's law [36]:

$$\mu = \mu_{ref} \left( \frac{T}{T_{ref}} \right)^{1.5} \times \frac{T_{ref} + S}{T + S} \quad (7)$$

This law gives the relation between dynamic viscosity ( $\mu$  [Pa s]) and the reference dynamic viscosity ( $\mu_{ref}$  [Pa s]) at a certain temperature ( $T$  [K]) where  $S$  [K] is a constant. Other thermodynamic properties such as heat capacities are calculated based on the Janaf polynomials [37].

### 3.5. Membrane model

Membranes are generally used in a wide variety of separation tasks ranging from light non-condensable gases over polar and nonpolar liquids to more complex long-chain molecules in solution. Based on the type of membrane a suitable mass transfer mechanism should be defined for calculation of mass transfer source term  $S_{mi}$  (Section 3.3). Gas permeation membranes are a very common type of membrane which are also used widely in biogas upgrading [38,39]. In this study for the first proof of concept, gas permeation membranes were selected and implemented into the suggested algorithm and relevant experiments and simulations were performed.

The solution-diffusion mechanism is one well-established model for modeling nonporous membrane films and consequently membrane gas permeation. This mechanism is based on three main steps [40]:

- Sorption of the permeating components at the feed/retentate side of the membrane
- Diffusion across the membrane
- Desorption at the permeate side

For mathematical formulation a relation between the driving force and transmembrane mass flow for component  $i$  across the membrane with area  $A$  can be applied.

$$S_{mi} = \Pi_i A (p_{i,R} - p_{i,F}) \quad (8)$$

The basic driving force in dense membrane transport is always a difference in chemical potential in the two separate regions. This can be reduced to fugacities for pervaporation and gas permeation and further reduced to partial pressures for gas permeation under the ideal gas assumption. As it can be seen from Eq. (8), in case of gas permeation the driving force can be described by the difference of component partial pressures ( $p_i$ ) at both sides of the membrane. The proportionality between mass transfer rate and driving force can be established by defining a permeance  $\Pi_i$  [ $\text{m}^3_{STP}/(\text{m}^2 \text{ s Pa})$ ]. Permeance is a phenomenological parameter which for common gases is the product of diffusion coefficient and solution coefficient divided by the membrane thickness [41]. Diffusivity depends primarily on the molecule size and solubility describes the sorption coefficient to the membrane. It is also frequently linked to a property of the gas called condensability. Therefore for glassy polymers and non-condensing gases permeance for a specific specie does not depend on the gas composition, but is mainly a function of the membrane material and the temperature [42].

For modeling the gas permeation membranes the solution-diffusion model was implemented as runtime selectable library and linked into the new solver for modeling the mass transfer mechanism across membrane.

### 3.6. Membrane solver

Fig. 1 shows the suggested algorithm for a membrane model based on a multi-region approach for a transient implementation. In the case of steady state simulations, no physical time step is used and the solver iterates (loops over virtual time steps) until it converges. In this approach, membranes are treated as infinitely thin membranes. The cell zones on both of the sides of the membrane are treated as separate regions. The regions are coupled through the common membrane boundary condition between them. At the beginning of a time step or iteration the species transmembrane fluxes and heat fluxes through the membranes are calculated using the fluid properties at the two sides of the membrane (retentate and permeate) for that time step or iteration – e.g. based on the partial pressure difference of the permeating species in the retentate and permeate and the available membrane area for each cell adjacent to the membrane surface. Since PISO is a non-iterative algorithm, a sufficiently small time step should be used for yielding accurate results in transient simulations [30]. Utilizing small time steps in the case of transient simulations also ensures negligible changes in the material and flow properties in each time step and therefore the calculated fluxes and source terms can be considered to be constant during each time step. After calculating source terms (which are considered to be constant in each time step or iteration after this update) for species and energy for the membrane boundaries for all regions, a loop over different regions is executed. In this loop, the membrane boundary conditions are first updated for each region and then the PISO algorithm is used to calculate the pressure and velocity. The suggested algorithm proposes a segregated approach using the calculated pressure and velocity fields for solving the other conservation equations iteratively [43,44], e.g. species transport, energy equation and the required turbulence equations (based on the selected turbulence model - no equations will be solved in the case of laminar simulation). Finally, the fluid properties are updated using the new velocity, pressure, temperature fields for each region and the algorithm advances to the next region. After solving and updating the fields and properties for all regions in the case of transient simulation the time step is calculated using the maximum flow Courant number for all regions and the simulation marches to the next time step (for a steady state simulation, the next iteration).

The algorithm was implemented into OpenFOAM® as a new solver “membraneFoam”. The solver algorithm consists of two main loops, time loop and region loop (e.g. entire retentate or permeate), where the

calculations of each region are managed (e.g. retentate and permeate). Models like membrane models, turbulence models and permeance models can be advantageously added to the main solver as runtime selectable libraries. That is the new models can be added to the solver just by including them into the simulation settings at run time – without changing and compiling the main solver. This simplifies the addition and debugging of new models as they are independent from other parts of the code.

Fig. 2 shows the membraneFoam flowchart including its dependencies. In addition to the original libraries a few more libraries are added to the solver for covering membrane models. In the original OpenFOAM® code, diffusion coefficient is calculated based on the Schmidt analogy ( $Sc = 1$ ) which is mainly appropriate for gases at low pressure and very low mass/mole fraction of the diffusing component [45]. Therefore, a diffusion library has been added, which currently covers two diffusion models, the first one using a constant diffusion coefficient and the second model calculating the diffusion coefficients based on the Schmidt analogy.

$$Sc = \frac{\nu}{D} \quad (9)$$

where  $\nu$  [ $\text{m}^2/\text{s}$ ] is the kinematic viscosity and  $D$  [ $\text{m}^2/\text{s}$ ] is the mass diffusivity. New diffusion models can be added to this library and they will be available in the solver as runtime selectable models.

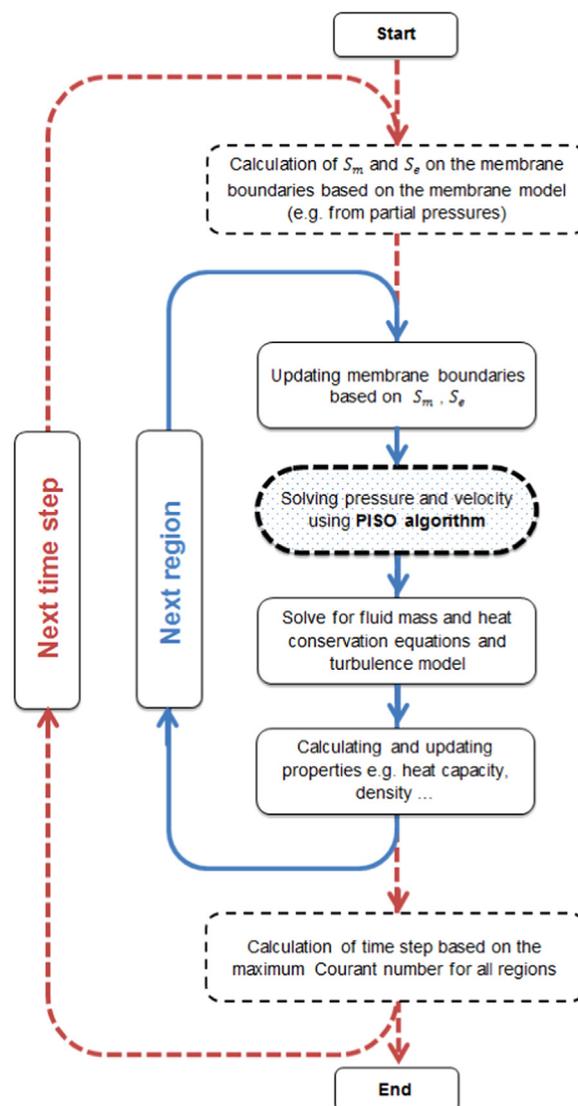


Fig. 1. Algorithm for modeling membranes (membraneFoam).

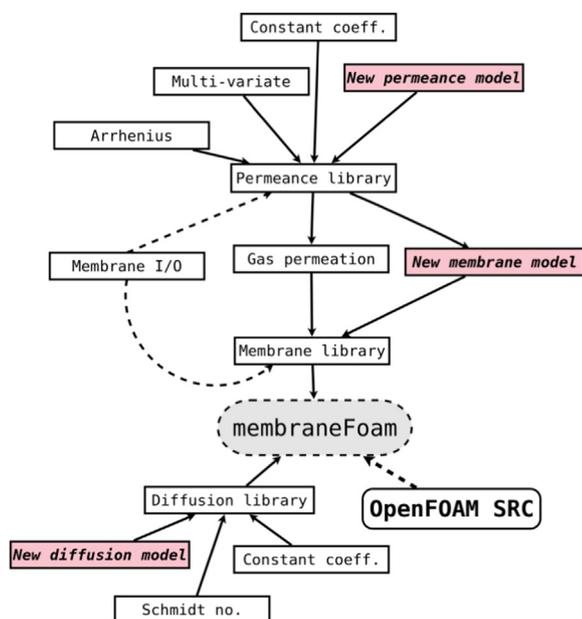


Fig. 2. membraneFoam flow chart with dependencies and linked libraries – the red boxes show the available possibilities for adding new runtime selectable models to the solver without modifying other parts of the code. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The membrane model library inherits from the membrane I/O and the permeance model libraries. The membrane I/O library is responsible for reading and writing the settings for membranes from dictionaries. New models can be added to the membrane model library in a similar way as to the diffusion model library and then being included into the solver as runtime selectable models. For this gas permeation study, the solution-diffusion mechanism using Eq. (8) was implemented to calculate the transmembrane fluxes based on permeances and the difference in the partial pressures on both sides of the membrane.

Different (run time selectable) permeance models can be added to the solver using the permeance library which currently features three types of permeance models: constant, Arrhenius and multi-variate. The Arrhenius model adds the temperature dependency to the constant permeance based on the Arrhenius equation [46]:

$$P = P_0 e^{-\frac{E}{RT}} \tag{10}$$

where  $P_0$  [ $m^3_{STP}/(m^2 s Pa)$ ] is the constant and  $E$  [J/mol] is activation energy. The multi-variate model can calculate the permeance as a function of different variables e.g. pressure, temperature and concentration.

#### 4. Experimental

The proposed model and implemented algorithm were tested against two different membrane systems:

- A *small module* with 30 hollow fibers – experiments performed in this study
- A *pilot scale module* with a few hundred hollow fibers [47] – literature values have been used

In the following, the characteristics and operating conditions of these two systems are described.

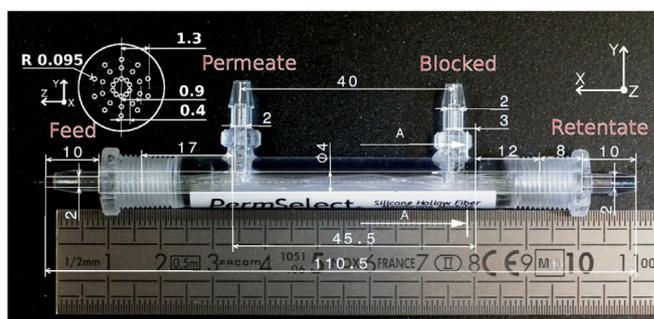


Fig. 3. Small membrane module and its dimensions (mm).

#### 4.1. Small module – pure gas operation

A hollow fiber membrane module (PermSelect, PDMSXA – 10 cm<sup>2</sup>) with 30 fibers made of polydimethylsiloxane (PDMS) was used for performing experiments [48]. The module and its dimensions can be seen in Fig. 3. Feed enters on the left hand side and retentate exits on the right hand side. The module was operated in counter-current configuration by closing the retentate side permeate outlet.

Fig. 4 shows the experimental setup used for the small membrane module measurements. The feed flow rate was controlled with the Coriolis principle based digital mass flow controller (CORI-FLOW®, Bronkhorst Cori-Tech B.V., Netherlands). The retentate and permeate mass flow rates were measured using a positive displacement flow meter (Type Definer 220, Mesa Labs Inc) with integrated pressure and temperature sensors to compensate for standard conditions with 1% standardized accuracy. Pressures before and after module were measured using P3276 relative pressure sensors (Tecsis GmbH, Germany). The Feed pressure sensor placed right before the inlet to the module had a measurement range of 0–100 bar and the sensors used for permeate and retentate (placed before flow meter to prevent excess pressures) had ranges of 0–25 bar with all sensors having an accuracy of 0.5% full scale.

Three pure gases (CH<sub>4</sub>, CO<sub>2</sub>, H<sub>2</sub>, all from AirLiquide or Messer, quality 5.0) were passed through the membrane module and the permeate flows were measured. Permeances were calculated based on the solution diffusion assumption and the membrane area reported in the membrane data sheet using the measured permeate flows. The permselectivity of CO<sub>2</sub> over CH<sub>4</sub> calculated from permeances was ~ 4. A list of gases, module properties and operating conditions are provided in Table 1.

#### 4.2. Pilot scale module – mixed gas operation

Membrane modules used in industry have commonly more than a

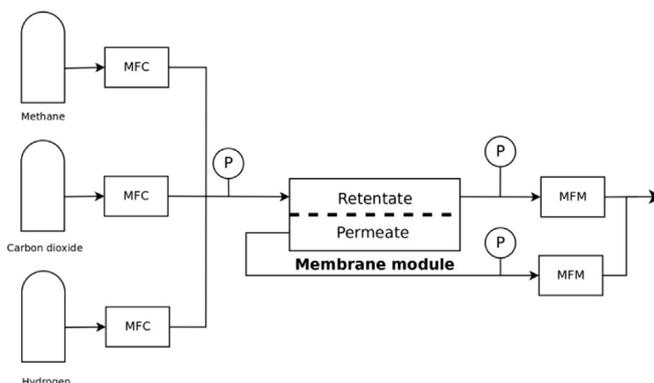


Fig. 4. Flowchart of experimental setup for small membrane module measurements.

**Table 1**  
Gases, module properties and operating conditions for pure gas operation [49].

Species	Feed flow [kg/s]	Permeance [ $\text{N m}^3/(\text{s m}^2 \text{ Pa})$ ]	Retentate absolute pressure [kPa]	Permeate absolute pressure [kPa]	Operating temperature [K]
CO <sub>2</sub>	$1.39 \times 10^{-5}$	$3.48 \times 10^{-10}$	400	80	298
CH <sub>4</sub>	$5.56 \times 10^{-6}$	$8.63 \times 10^{-11}$	401	80	298
H <sub>2</sub>	$5.57 \times 10^{-7}$	$8.34 \times 10^{-11}$	400	80	298

few hundred or thousand fibers. To check the validity of the proposed model and implemented methods and workflow for simulating the industrial scale membrane modules, the code outputs were compared to reported data in literature. The selected module is an aromatic polyimide (PI) hollow fiber membrane module with 800 fibers with an active fiber length of 0.38 m and a total of 0.38 m<sup>2</sup> membrane area [47]. Feed was entering the module at an absolute pressure of 9 bar and a temperature of 316.45 K. The permeate absolute pressure was kept constant at 1.1 bar.

Feed used for the experiments was a biogas-like mixture of methane, carbon dioxide and oxygen with the volume fractions given in the Table 2. The pure gas permeances are also listed in Table 2.

Results from different feed flow rates to achieve different stage cuts (permeate flow related to feed flow) ranging from 0.15 to 0.65 for both co-current and counter-current configurations are reported in [47].

## 5. Simulation boundary conditions and geometries

Two types of simulations were performed using the new solver:

- fully resolved 3D simulation of the *small module*
- 1D simulation of the *small module* and the *pilot scale module*.

Since the developed algorithm and solver are originally in 3D, a special workflow has been developed for preparing and performing the simulations in 1D. Since 1D simulations are just valid under certain conditions – such as low concentration polarization or good inlet flow distribution – a checklist for the most important assumptions has been included into the workflow. Fig. 5 shows the proposed 1D-geometry setup for a hollow fiber membrane module, including the regions (e.g. retentate and permeate) for performing 1D simulation. The cylindrical fiber wall is represented with a flat wall inside a rectangular box with just one row of cells in the directions perpendicular to the flow direction per region. The area between regions is equal to one single fiber surface area and the length in the direction of flow is equal to the average fiber length. The cross sections perpendicular to the flow direction are equal to the fiber cross section for the fiber side and to the shell cross section divided by number of fibers for shell side, respectively.

The workflow for utilizing the CFD code in 1D for different membrane types can be summarized as follows:

1. The 1D module length should be aligned with the main (e.g. hollow fiber axis) flow direction
2. The geometry should have the same length as the active fiber length (represented by a mesh with any number of cells which is suitable for resolution/accuracy) in axial direction
3. In the other directions the mesh should have just one cell per region (retentate/permeate)
4. The interface between two membrane regions should have the same width as membrane, e.g.:
  - a. Hollow fibers: the width should be the same as one fiber circumference
  - b. Flat sheets: the same width as sheet width
5. In case of a hollow fiber membrane the calculations just per one fiber are done and then they should be multiplied by the number of

fibers. Also the overall module flow rate should be divided by number of fibers and the velocity should be the superficial velocity for a single fiber

6. Assumption of perfect flow distribution - equal flow rate in all parallel fibers (perfect flow directions) and ideal concentration field (low/negligible concentration polarization relative to the flow path cross section)
7. Each region mesh should have the same equivalent cross section as the corresponding module part, e.g.: Hollow fibers: the fibers region should have the same cross section as one fiber and the shell region should have area equal to shell side cross section divided by number of fibers – for total mass flow the calculation result is multiplied by number of fibers
8. Boundary conditions should be treated carefully based on each case, e.g.:
  - a. Hollow fibers: the membranes surfaces as no-slip boundary and coupled heat transfer between two regions. The other boundaries (except for inlet and outlets) should be treated as slip (zero gradient for velocity) and adiabatic since those boundaries are introduced by representing a hollow fiber with a flat surface and should therefore not affect the flow results
  - b. Flat sheet: the boundaries including the membrane as no-slip boundary and also with heat transfer

### 5.1. Small module – pure gas operation

The hollow fiber membrane module with 30 fibers was simulated both in 3D and in 1D configuration to compare both approaches and check the validity of suggested methodology for the 1D simulation of membranes.

#### 5.1.1. 3D – fully resolved simulation

The small module shown in Fig. 3 was drawn with the CAD program Catia® (V5, 2016). Mesh creation was done using the automatic mesher snappyHexMesh (included with OpenFOAM® – version 4.1, 2016) resulting in approximately 10 million cells with more than 99% hexahedral and polyhedral cells. Fig. 6 shows the geometry of the module and the mesh structure at two cross sections with an average mesh size of around 10–20 μm near permeable walls. In order to have a sufficient mesh resolution in the small gaps between fibers, a higher mesh refinement level was applied in this zones.

The types of boundary conditions for this simulation case are listed in Table 3. The relevant boundary values (flow rates, temperatures, pressures) for feed entering the module, retentate and permeate streams are summarized in Table 1.

The high flow rates inside the fibers and low fiber cross section compared to shell side makes the pressure drop on the fiber side more dominant. As the current model neglects the wall thickness of the membrane only the inner fiber diameter was implemented in the

**Table 2**  
Pure gas permeances and feed volume fractions [47].

Species	Permeance $\text{N m}^3_{\text{TP}}/(\text{s m}^2 \text{ bar})$	Volume fraction
CO <sub>2</sub>	$5.91 \times 10^{-5} \pm 2\%$	0.345
CH <sub>4</sub>	$1.59 \times 10^{-6} \pm 2\%$	0.645
O <sub>2</sub>	$1.36 \times 10^{-5} \pm 2\%$	0.01

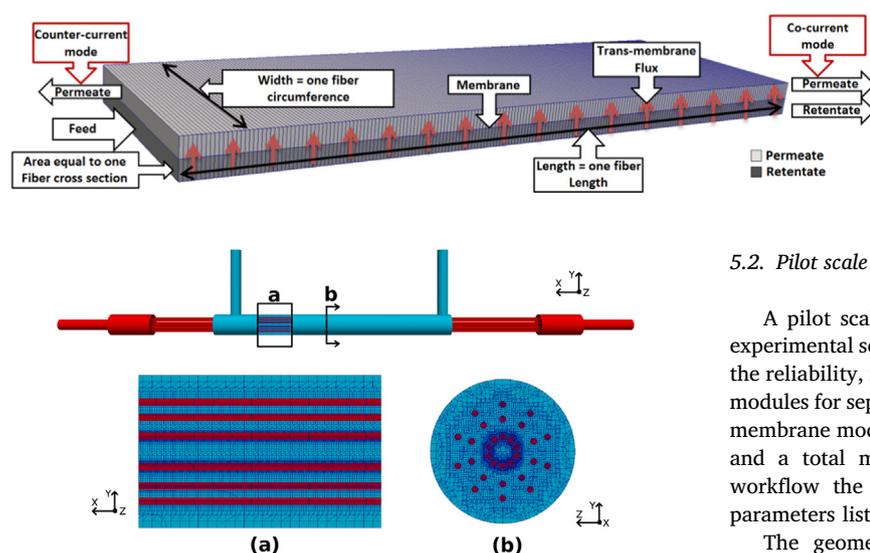


Fig. 5. One dimensional module sample geometry and mesh.

Fig. 6. Small module geometry and mesh at two cross sections (geometry: Catia<sup>®</sup>, mesh: snappyHexMesh, Visualization using Paraview).

Table 3  
Simulation boundary conditions.

	Pressure	Velocity	Temperature	Species
<b>Feed</b>	Zero gradient	Mass flow rate	Fixed value	Fixed value
<b>Retentate</b>	Fixed value	Zero gradient	Zero gradient	Zero gradient
<b>Permeate</b>	Fixed value	Zero gradient	Zero gradient	Zero gradient
<b>Membrane</b>	Zero gradient	No slip	Coupled wall	Permeable membrane
<b>Walls</b>	Zero gradient	No slip	Zero gradient	Zero gradient

geometry for representation of the fibers to enable better capturing of the flow, pressure drop and the flow structures on both sides of the membrane. Furthermore the larger cross section in the shell side keeps the relative error due to membrane wall thickness smaller in the shell side. Permeances were adapted based on this assumption [49]. The hollow fibers were considered as tubular structures with smooth surfaces (selective membrane layer), with parallel orientation and with no contact points between them. The simulations were performed in steady state mode with second order linear discretization [50] applied to all numerical schemes (e.g. gradient, divergence, laplacian etc.). With a flow velocity inside of the fibers of around 2 m/s and a fiber diameter of 190  $\mu\text{m}$  the Reynolds number is around 50; the flow can be considered as laminar flow.

### 5.1.2. 1D – global scale simulation

In the next stage the small membrane module was simulated as a 1D module using the proposed procedure for using membraneFoam as a 1D solver. The 1D geometry consisted of a simple cubic structure with the properties as reported in Table 4.

The geometry and mesh were created using the OpenFOAM<sup>®</sup> blockMesh utility with 200 cells in the main flow direction (along fibers) and one cell per other directions per region. Since the module operates in counter-current configuration, feed inlet and permeate outlet are located at one side and retentate flow exits at the opposite side. All the boundary conditions are set based on the guide lines given in Section 5. The same boundary conditions and boundary values as reported in Tables 1 and 3 was also applied to the 1D case. The same models, discretization schemes and solution algorithm which were applied to 3D case were also used for 1D simulation.

### 5.2. Pilot scale module – mixed gas operation

A pilot scale hollow fiber membrane module as explained in the experimental section was modeled using the solver in 1D mode to check the reliability, robustness and stability of the solver in simulation of big modules for separation of multi-component gas mixtures. The simulated membrane module had 800 fibers with an active fiber length of 0.38 m and a total membrane area of 0.38  $\text{m}^2$  [47]. Using the proposed workflow the geometry was represented using a cuboid with the parameters listed in Table 5.

The geometry and mesh were created using the OpenFOAM<sup>®</sup> blockMesh utility with 200 cells in the main flow direction (along fibers) and one cell per other directions per region. The feed flow entered the module at left side (Fig. 5). In co-current mode the permeate outlet was placed at the right end (close to retentate outlet), in counter-current mode it was placed at the left end. The boundary conditions were set as explained for the small module in the previous section. The flow was considered laminar (Reynolds number below 50 for all inlet velocities). Second order linear discretization schemes were used and simulations were performed in steady state mode.

The same feed flow rates (at different flow rates to achieve various stage cuts) as for the experiments were used for the simulation applying the pure gas permeances and the feed volume fractions as listed in Table 2. Because of the low  $\text{CO}_2$  partial pressure plasticization was neglected for the membrane. Feed was entering the module at an absolute pressure of 9 bar and a temperature of 316.45 K. The permeate absolute pressure was kept constant at 1.1 bar.

## 6. Results and discussions

A grid convergence study was performed on both, the *small module* (1D and 3D) and the *pilot scale module* (1D) to make sure about the independency of the results from applied spatial resolution. The permeate flow rates and also the mass fractions of the permeate flow were compared for simulations with different mesh density. Since all the simulations were performed in steady state mode the convergence was checked by both physical and numerical monitors. For physical monitor the species stage cut (permeate flow rate related to feed flow rate) and for numerical monitor species mass fraction residuals were monitored. Like original OpenFOAM<sup>®</sup> solvers, this solver can make use of all available library based extensions present in this CFD code (e.g. numerical schemes or solution algorithms, transient or steady state solution algorithms).

The 3D simulations (*small module*) were performed in parallel on 40 CPU cores (AMD FX-8320E @ 23 Gflops Processors). Each of the simulations took around 24 h to converge. For the 1D cases (*small module* and *pilot scale module*) the time needed for running each of the simulations on a single core was about 3–5 s on an AMD Phenom™ II X6 1045T (15 Gflops) Processor. The run times for these simulations are

Table 4  
Small module 1D geometrical parameters.

	Representative quantity	Value
<b>Length</b>	Fiber length	0.0455 m
<b>Cross section</b>	Fiber cross section	$2.8 \times 10^{-8} \text{ m}^2$
<b>Width</b>	Fiber circumference	$6.0 \times 10^{-4} \text{ m}$

**Table 5**  
Pilot scale module 1D geometrical parameters.

	Representative quantity	Value
Length	Fiber length	0.38 m
Cross section	Fiber cross section	$2.5 \times 10^{-7} \text{ m}^2$
Width	Fiber circumference	$1.25 \times 10^{-3} \text{ m}$

very good in comparison with the time needed to perform the same simulation on a single core with the same number of discretization points with other process simulation codes, e.g. Makaruk et al. [47].

### 6.1. Small module – pure gas operation

The results of 1D and 3D simulations of the small module were compared to experimental data. In the following, more details of the results are discussed.

#### 6.1.1. Velocity

The 3D simulations can provide spatial information; therefore using 3D simulations the velocities and velocity distributions for hydrogen will be discussed. The other two gases showed comparable behavior in the simulations. Fig. 7 shows the velocity distribution in the retentate side of the membrane on the symmetry plane for hydrogen. The plot is scaled 1:10 in the direction of the fibers. As it can be seen in Fig. 8 peak velocity magnitudes and velocity profiles for the fibers in the symmetry plane of the module are the identical (fibers with numbers 1–6 in Fig. 9). The peak velocity for all fibers is around 3.9 m/s. The parabolic velocity profile in the fibers was expected as there is laminar flow on the fiber side.

In Fig. 9, the contour plot of the hydrogen gas velocity on the symmetry plane in the permeate side is shown. The highest velocity is 0.018 m/s and as expected, it can be found in the permeate outlet, (for enhanced visibility of the velocity gradient on the shell side, the velocity range was scaled to 0.004 m/s). Since the membrane was operated in counter-current configuration the flow in the permeate (shell) side is flowing from right to left. By moving from right to left the total shell side flow increases as it equals the integral of the transmembrane flux over the fiber length. Fig. 10 shows the velocity profiles extracted from permeate side at the positions marked in Fig. 9 on the horizontal lines passing the center of the shell. The velocities are low at the feed entrance side (left), a maximum velocity is located next to the permeate outlet. As the configuration is counter-current, the velocity decreases towards the retentate outlet on the right hand side. It can be also seen the velocities are higher close to the outer shell wall and by moving towards inner fibers the velocity decreases and then increases at the center of the module. This trend can be seen at all the positions due to lower fiber density near the wall (Fig. 6b) and close to the center of the module which leads to larger free flow cross section in these areas and thus to lower resistance to flow. The velocities become zero at the walls because of the no-slip boundary conditions at walls. Comparing the velocities on the retentate and permeate side, the velocities on the permeate side are much smaller and the relative velocities on both sides of the membrane at different positions are almost the same. This justifies the assumption of dividing the flow on the permeate side by number of fibers and considering an equally divided flow in the permeate side for 1D simulations.

#### 6.1.2. Pressure drop

Fig. 11 shows the absolute pressure inside a single fiber along the length for the hydrogen pure gas case for both 1D and 3D cases compared to the Darcy-Weisbach correlation. The pressure drop on the shell side was negligible because of very low flow rates compared to the fiber side. As it can be seen 1D and 3D cases predict almost the same pressure drop of 1.5 kPa. The reason for the 0.5 kPa offset between the pressure

profile lines is that in the 1D case the inlet and also outlet parts of the module are ignored in the simulation and consequently their pressure drops are not considered in the 1D runs.

The Darcy-Weisbach [51] correlation was used for the calculation of pressures drop inside the fiber along the length with assumption of a fixed outlet pressure for both, the 1D and the 3D cases (400 kPa).

$$P = f_D \frac{\rho}{2} \frac{\bar{u}^2}{D} + P_o \quad (11)$$

where  $L$  is the fiber length,  $D$  is fiber diameter,  $\rho$  is the density,  $P_o$  is the outlet pressure and  $\bar{u}$  is the average flow velocity. Since the flow was laminar (Reynolds  $\sim 14$ ) in a smooth pipe, the friction factor ( $f_D$ ) was calculated using [52]:

$$f_D = \frac{64}{Re} \quad (12)$$

As it can be seen in Fig. 11 the pressure drop calculated by correlation is around 0.8 kPa and it is less than the pressure drop from simulations. In the CFD based simulations, more sophisticated boundary conditions and flow phenomena compared to the correlation based approaches (e.g. many of process simulation methods [3]) are applied, e.g. mass transfer. This allows for local calculation of the actual flow velocity and therefore more accurate pressure drop prediction.

#### 6.1.3. Separation model performance

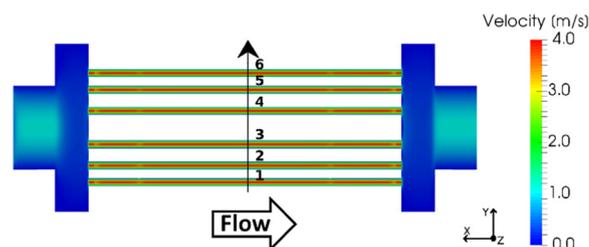
Table 6 shows the comparison of the simulation results for permeate flow in 1D and 3D with experimental data. Reasonable agreement between simulations results (in both 1D and 3D modes) and the experimental data with less than 2% difference can be observed. The reason for the lower 3D simulation results compared to experimental results can be attributed to neglecting complex fiber geometry e.g. twisted fibers, which results in slightly lower membrane area in the CFD mesh compared to the real membrane module. The deviations between 1D simulation data and experiments can also be partially explained by application of the simplifying assumptions (see Section 5 – 1D membrane setup).

### 6.2. Pilot scale module – mixed gas operation

The pilot scale module was simulated using the 1D approach in both co-current and counter-current configurations. In both cases the experimental data have 1% measurement uncertainty in the measured concentration ranges. The simulations and experiments were performed at different stage cuts to evaluate the separation performance of the module.

#### 6.2.1. Co-current flow

Fig. 12 shows the results for module in co-current configuration. As it can be seen the results from the 1D simulations are in good agreement with the experimental data from literature [47]. By increasing the stage cut (decreasing the feed flow rate) the carbon dioxide concentration in the retentate drops from 25% to less than 10% which results in higher



**Fig. 7.** Velocity magnitude contour plots on the symmetry plane and the fibers number on this plane for hydrogen – plot scaled 1:10 in fiber direction. Numbers indicate fiber cross section for detailed evaluation in Fig. 8.

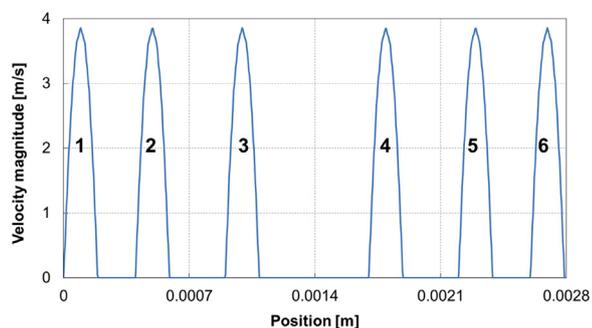


Fig. 8. Velocity magnitude for fibers extracted from line shown in Fig. 7 on the symmetry plane for hydrogen.

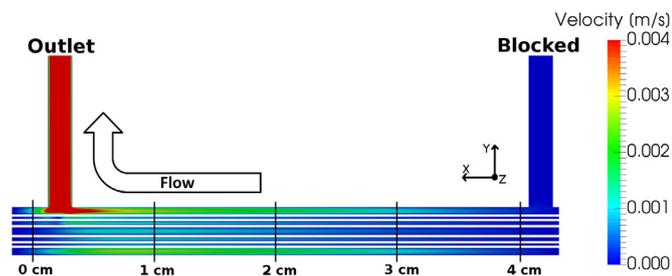


Fig. 9. Velocity magnitude at the permeate side on the symmetry plane for hydrogen – position marks are for data extracted for Fig. 10.

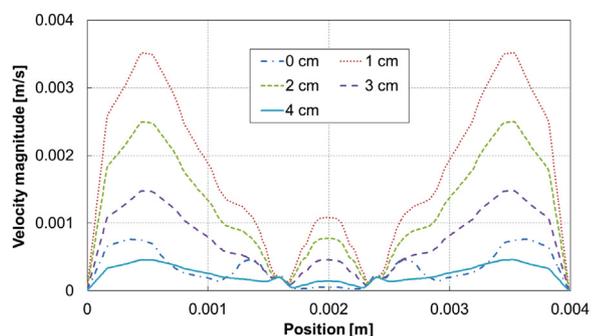


Fig. 10. Velocity magnitude at the permeate side for different positions shown in Fig. 9 on the horizontal lines passing the center of the module for hydrogen.

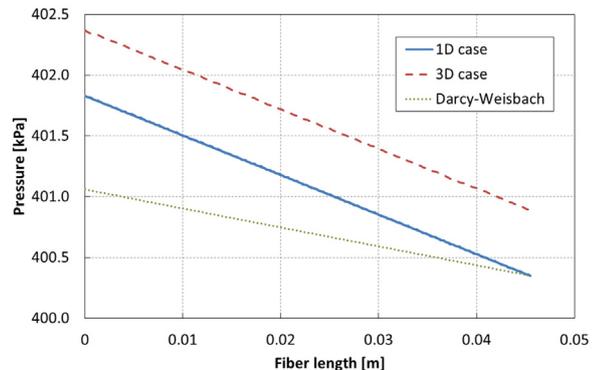


Fig. 11. Comparison of absolute pressures from 1D and 3D simulation for H<sub>2</sub> with pressures calculated using Darcy-Weisbach correlation.

concentration of methane in the retentate (over 90%). Overall, the results from simulation compare reasonably well to the available experimental data.

Table 6

Feed and permeate flow rates for 1D, 3D simulations and experiment for pure gas operation, the relative deviation between the experiment and simulation is reported as percentage.

Species	Feed flow [kg/s]	Permeate flow 1D [kg/s]	Permeate flow 3D [kg/s]	Permeate flow experiment [kg/s]
H <sub>2</sub>	$5.57 \times 10^{-7}$	$1.96 \times 10^{-9}$ (1.0%)	$1.93 \times 10^{-9}$ (0.5%)	$1.94 \times 10^{-9}$
CO <sub>2</sub>	$1.39 \times 10^{-5}$	$1.94 \times 10^{-7}$	$1.90 \times 10^{-7}$ (1.6%)	$1.93 \times 10^{-7}$
CH <sub>4</sub>	$5.56 \times 10^{-6}$	$1.62 \times 10^{-8}$ (0.6%)	$1.59 \times 10^{-8}$ (1.2%)	$1.61 \times 10^{-8}$

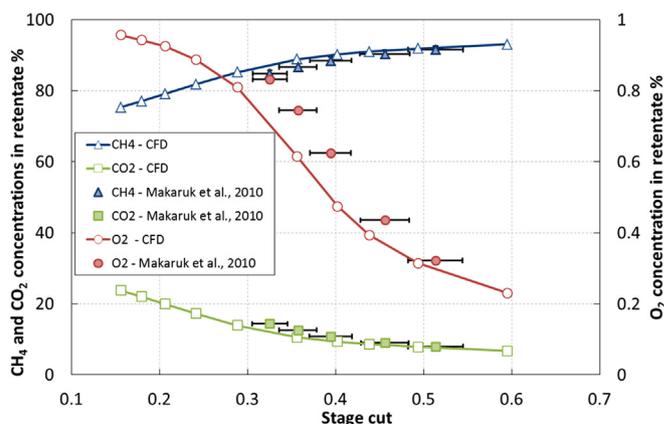


Fig. 12. Comparison of gas concentrations at different stage cuts in the retentate for co-current module: one-dimensional CFD code and experimental data [47] – oxygen is shown on the secondary y-axis.

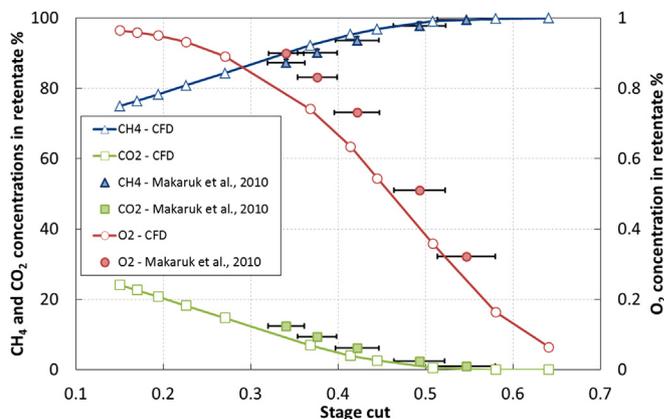


Fig. 13. Comparison of gas concentrations at different stage cuts in the retentate for counter-current module: one-dimensional CFD code and experimental data [47] – oxygen is shown on the secondary y-axis.

### 6.2.2. Counter-current flow

Similar simulations were performed on the same module also in counter-current mode. As it can be seen from the results in Fig. 13, similar to co-current operation mode, the CO<sub>2</sub> concentration in retentate decreases and CH<sub>4</sub> concentration increases with increasing the stage cut from 0.15 to 0.65. However, in counter-current operation mode CO<sub>2</sub> can be almost completely removed from retentate and the CH<sub>4</sub> purity of around 100% can be achieved. Counter-current configuration is preferential for practical application because of the optimum utilization of driving force e.g. in biogas upgrading where more carbon dioxide can be removed from mixture to obtain higher methane purity grades. Again good agreement can be seen between simulation and

experiment.

## 7. Conclusion

In this study a new CFD algorithm for modeling membranes is introduced. This algorithm is based on a multi-region approach for modeling different membrane module compartments as separate regions e.g. retentate and permeate which are connected using the membrane boundary. Mass and heat transfer through the membrane boundary can be handled for any number of components (species). Each region can have its own thermo-physical properties, turbulence models etc. An implementation flow chart is suggested for modular implementation of the suggested algorithm for easy expandability of the available models. The high flexibility of the algorithm allows the use in 1D or 3D membrane modeling – depending on the desired results. The implementation was done based on the open source CFD code OpenFOAM®.

A workflow for performing 1D simulation using the developed algorithm and solver is presented. The validity of the new algorithm and developed solver was tested by simulating two modules with available experimental data. A small hollow fiber module with 30 fibers to allow also fast enumeration in full resolved 3D geometry was simulated both in 1D and 3D for gas permeation of three pure gases (H<sub>2</sub>, CO<sub>2</sub>, CH<sub>4</sub>). Simulations were compared to the experiments performed by authors and a good agreement was observed with less than 2% deviation. In the studied module the flow rates were higher at walls and in the center of the module since fibers density was lower in these zones. In the second case a pilot scale membrane module with 800 hollow fibers and 0.38 m length was simulated in 1D only due to unreasonably high computational demand in 3D. In this simulation the separation performance of the module for a biogas mixture were studied. Different stage cuts as well as co- and counter-current configurations were investigated. It was observed that simulation results were in good agreement with experimental data. By varying the stage cut from 0.15 to 0.65 it was confirmed that in both configurations CO<sub>2</sub> concentration decreased in the retentate flow. In counter current configuration the CO<sub>2</sub> concentration can be reduced down to almost 0% which allows for the production of technically pure CH<sub>4</sub> at the retentate outlet.

Successful implementation and good agreement between simulations and experiments demonstrate that the same algorithm can be used for detailed 3D analysis of a module (investigation of geometrical effects like spacers or flow effects like concentration polarization) and also at the same time can be employed for modeling much bigger modules in 1D for fast and efficient process optimization (e.g. outlet positioning, feed flow rate adjustment or operating pressures changes).

Future work will cover implementation of other membrane separation processes (e.g. pervaporation, nanofiltration, reverse osmosis) with different membrane types and the used of the code with alternative geometries such as hollow fibers, flat sheet, spiral wound and cushion type. It will also be demonstrated how the 1D results can be used to make 3D simulations more efficient by using them as initial conditions. Also combining the 1D and 3D capabilities of the algorithm can provide a promising tool for investigating the inlet and outlet sections in detail (3D) combined with transmembrane flux in lower detail (1D) for sake of accuracy and speed.

## Acknowledgement

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. All the authors are employed by TU Wien.

Authors want to thank for all the help and support from Hamidreza Norouzi and Zsolt Harsfalvi.

## References

- [1] R. Qi, M.A. Henson, Membrane system design for multicomponent gas mixtures via mixed-integer nonlinear programming, *Comput. Chem. Eng.* 24 (2000) 2719–2737.
- [2] M.M. Gimmelshtein, R. Semiat, Investigation of flow next to membrane walls, *J. Membr. Sci.* 264 (2005) 137–150.
- [3] J. Marriott, E. Sørensen, A general approach to modelling membrane modules, *Chem. Eng. Sci.* 58 (2003) 4975–4990.
- [4] G. Barbieri, F.P. Di Maio, Simulation of the methane steam re-forming process in a catalytic Pd-membrane reactor, *Ind. Eng. Chem. Res.* 36 (1997) 2121–2127.
- [5] S.P. Kaldis, G.C. Kapantaidakis, G.P. Sakellariopoulos, Simulation of multi-component gas separation in a hollow fiber membrane by orthogonal collocation—hydrogen recovery from refinery gases, *J. Membr. Sci.* 173 (2000) 61–71.
- [6] S.S.M. Lock, K.K. Lau, F. Ahmad, A.M. Shariff, Modeling, simulation and economic analysis of CO<sub>2</sub> capture from natural gas using cocurrent, countercurrent and radial crossflow hollow fiber membrane, *Int. J. Greenh. Gas Control* 36 (2015) 114–134, <http://dx.doi.org/10.1016/J.IJGGC.2015.02.014>.
- [7] F. Ahmad, K.K. Lau, S.S.M. Lock, S. Rafiq, A.U. Khan, M. Lee, Hollow fiber membrane model for gas separation: process simulation, experimental validation and module characteristics study, *J. Ind. Eng. Chem.* 21 (2015) 1246–1257, <http://dx.doi.org/10.1016/J.JIEC.2014.05.041>.
- [8] I. Rossetti, M. Compagnoni, M. Torli, Process simulation and optimization of H<sub>2</sub> production from ethanol steam reforming and its use in fuel cells. 2. Process analysis and optimization, *Chem. Eng. J.* 281 (2015) 1036–1044, <http://dx.doi.org/10.1016/J.CEJ.2015.08.045>.
- [9] X. He, C. Fu, M.-B. Hägg, Membrane system design and process feasibility analysis for CO<sub>2</sub> capture from flue gas with a fixed-site-carrier membrane, *Chem. Eng. J.* 268 (2015) 1–9, <http://dx.doi.org/10.1016/J.CEJ.2014.12.105>.
- [10] N.M. Mazlan, D. Peshev, A.G. Livingston, Energy consumption for desalination — a comparison of forward osmosis with reverse osmosis, and the potential for perfect membranes, *Desalination*. 377 (2016) 138–151, <http://dx.doi.org/10.1016/J.DESAL.2015.08.011>.
- [11] F. Ahmad, K.K. Lau, A.M. Shariff, G. Murshid, Process simulation and optimal design of membrane separation system for CO<sub>2</sub> capture from natural gas, *Comput. Chem. Eng.* 36 (2012) 119–128.
- [12] E. Pellerin, E. Michelitsch, K. Darcovich, S. Lin, C.M. Tam, Turbulent transport in membrane modules by CFD simulation in two dimensions, *J. Membr. Sci.* 100 (1995) 139–153.
- [13] M. Rezakazemi, Z. Niazi, M. Mirfendereski, S. Shirazian, T. Mohammadi, A. Pak, CFD simulation of natural gas sweetening in a gas-liquid hollow-fiber membrane contactor, *Chem. Eng. J.* 168 (2011) 1217–1226.
- [14] G.A. Fimbres-Weihs, D.E. Wiley, Review of 3D CFD modeling of flow and mass transfer in narrow spacer-filled channels in membrane modules, *Chem. Eng. Process. Process Intensif.* 49 (2010) 759–781.
- [15] D.A. Zaidiza, J. Billaud, B. Belaisaoui, S. Rode, D. Roizard, E. Favre, Modeling of CO<sub>2</sub> post-combustion capture using membrane contactors, comparison between one-and two-dimensional approaches, *J. Membr. Sci.* 455 (2014) 64–74.
- [16] F. Li, W. Meindersma, A.B. De Haan, T. Reith, Optimization of commercial net spacers in spiral wound membrane modules, *J. Membr. Sci.* 208 (2002) 289–302.
- [17] E. Piron, E. Latrille, F. Rene, Application of artificial neural networks for crossflow microfiltration modelling: “black-box” and semi-physical approaches, *Comput. Chem. Eng.* 21 (1997) 1021–1030.
- [18] N. Ashton, V. Skaperdas, Verification and validation of OpenFOAM for high-lift aircraft flows, *AIAA J.*, 2017.
- [19] J. Fang, J.J. Cambareri, C.S. Brown, J. Feng, A. Gouws, M. Li, I.A. Bolotnov, Direct numerical simulation of reactor two-phase flows enabled by high-performance computing, *Nucl. Eng. Des.* 330 (2018) 409–419.
- [20] S. Patankar, *Numerical Heat Transfer and Fluid Flow*, CRC press, Boca Raton, Florida, United States, 1980.
- [21] R. Ghidossi, D. Veyret, P. Moulin, Computational fluid dynamics applied to membranes: state of the art and opportunities, *Chem. Eng. Process. Process Intensif.* 45 (2006) 437–454.
- [22] R.B. Ferreira, D.S. Falcão, V.B. Oliveira, A.M.F.R. Pinto, Numerical simulations of two-phase flow in proton exchange membrane fuel cells using the volume of fluid method – a review, *J. Power Sources* 277 (2015) 329–342, <http://dx.doi.org/10.1016/J.JPOWSOUR.2014.11.124>.
- [23] M.M.A. Shirazi, A. Kargari, A.F. Ismail, T. Matsuura, Computational fluid dynamic (CFD) opportunities applied to the membrane distillation process: State-of-the-art and perspectives, *Desalination*. 377 (2016) 73–90, <http://dx.doi.org/10.1016/J.DESAL.2015.09.010>.
- [24] T. Brinkmann, J. Pohlmann, U. Withalm, J. Wind, T. Wolff, Theoretical and experimental investigations of flat sheet membrane module types for high capacity gas separation applications, *Chem. Ing. Tech.* 85 (2013) 1210–1220.
- [25] M. Scholz, T. Harlacher, T. Melin, M. Wessling, Modeling gas permeation by linking nonideal effects, *Ind. Eng. Chem. Res.* 52 (2012) 1079–1088.
- [26] H. Jasak, A. Jemcov, Z. Tukovic, et al., OpenFOAM: A C++ library for complex physics simulations, in: *Proceedings of the International Workshop on Coupled Methods in Numerical Dynamics*, 2007, pp. 1–20.
- [27] W.J. Schell, Commercial applications for gas permeation membrane systems, *J. Membr. Sci.* 22 (1985) 217–224.
- [28] R.Y.M. Huang, *Pervaporation Membrane Separation Processes*, Elsevier Science Ltd, Amsterdam, Netherlands, 1991.
- [29] H. Jasak, Handling parallelisation in openfoam, in: *Proceedings of the Cyprus Advanced HPC Workshop*, 2012.
- [30] H.K. Versteeg, W. Malalasekera, *An Introduction to Computational Fluid Dynamics:*

- The Finite Volume Method, Pearson Education, London, England, United Kingdom, 2007.
- [31] I.E. Barton, Comparison of SIMPLE-and PISO-type algorithms for transient flows, *Int. J. Numer. Methods Fluids* 26 (1998) 459–483.
- [32] R.I. Issa, A.D. Gosman, A.P. Watkins, The computation of compressible and incompressible recirculating flows by a non-iterative implicit scheme, *J. Comput. Phys.* 62 (1986) 66–82.
- [33] R. Courant, K. Friedrichs, H. Lewy, Über die partiellen Differenzgleichungen der mathematischen Physik, *Math. Ann.* 100 (1928) 32–74.
- [34] S.P. Nunes, K.-V. Peinemann, *Membrane Technology*, Wiley Online Library, Hoboken, New Jersey, United States, 2001.
- [35] G.-J. Su, Modified law of corresponding states for real gases, *Ind. Eng. Chem.* 38 (1946) 803–806.
- [36] W. Sutherland, LII. The viscosity of gases and molecular force, *Lond. Edinb. Dublin Philos. Mag. J. Sci.* 36 (1893) 507–531.
- [37] F.D. Rossini, F.D. Rossini, *Selected Values of Chemical Thermodynamic Properties*, US Government Printing Office, Washington, DC, 1952.
- [38] E. Ryckebosch, M. Drouillon, H. Vervaeren, Techniques for transformation of biogas to biomethane, *Biomass Bioenergy* 35 (2011) 1633–1645.
- [39] M. Miltner, A. Makaruk, M. Harasek, Review on available biogas upgrading technologies and innovations towards advanced solutions, *J. Clean. Prod.* (2017).
- [40] L. Shao, B.T. Low, T.-S. Chung, A.R. Greenberg, Polymeric membranes for the hydrogen economy: contemporary approaches and prospects for the future, *J. Membr. Sci.* 327 (2009) 18–31.
- [41] L.M. Robeson, Polymer membranes for gas separation, *Curr. Opin. Solid State Mater. Sci.* 4 (1999) 549–552.
- [42] R.W. Baker, *Membrane technology*, *Encycl. Polym. Sci. Technol.* (2003).
- [43] A. Fluent, *12.0 Documentation*, Ansys Inc., 2009.
- [44] R.I. Issa, Solution of the implicitly discretised fluid flow equations by operator-splitting, *J. Comput. Phys.* 62 (1986) 40–65.
- [45] T.L. Bergman, F.P. Incropera, *Fundamentals of Heat and Mass Transfer*, John Wiley & Sons, Hoboken, New Jersey, United States, 2011.
- [46] L. Pauling, *General Chemistry*, Courier Corporation, Mineola, New York, United States, 1988.
- [47] A. Makaruk, M. Harasek, Numerical algorithm for modelling multicomponent multipermeator systems, *J. Membr. Sci.* 344 (2009) 258–265.
- [48] PermSelect, PDMSXA – 10 membrane module, 2018. <[http://permselect.com/files/Data\\_Sheet\\_Tiny\\_Rev-15-12.pdf](http://permselect.com/files/Data_Sheet_Tiny_Rev-15-12.pdf)> (Accessed 9 April 2018).
- [49] P. Schretter, *Simulation of Membrane Modules With OpenFOAM®* (Master thesis), Vienna University of Technology, Vienna, Austria, 2016.
- [50] C.J. Greenshields, *Openfoam user guide*, OpenFOAM Found. Ltd, Version. 3, 2015.
- [51] G.O. Brown, The history of the Darcy-Weisbach equation for pipe flow resistance, in: *Proceedings of the Environmental and Water Resources History*, 2003: pp. 34–43.
- [52] N.H. Chen, An explicit equation for friction factor in pipe, *Ind. Eng. Chem. Fundam.* 18 (1979) 296–297.