Politecnico di Torino
Universitat Politècnica de Catalunya

Master of Science in Energy and Nuclear Engineering

Master Thesis

# Development, Validation and Application of a Tritium Transport Model in the Liquid Metal and Permeation across the Membrane of the EU DEMO's PAV

*Supervisors:*
Roberto Bonifetto
Antonio Froio
Lluis Batet
Ronny Rives Sanz

*Student:*
Diana Sgrelli

March 19, 2024

# Abstract

One of the main challenges for the realization of the EU DEMO reactor is to achieve tritium self-sufficiency, for which a suitable design of the Tritium Extraction and Removal System (TERS) able to minimize tritium inventory maximizing its extraction is of crucial importance.

The Permeator Against Vacuum (PAV) is the reference technology for the tritium extraction from the lithium-lead (PbLi) for the EU DEMO Water-Cooled Lithium-Lead Breeding Blanket: it features a shell-and-tube configuration, where tritium dissolved in the liquid PbLi flowing inside the tubes is forced to cross a niobium membrane by pumping vacuum on the secondary side. However, the large uncertainty in several tritium transport properties, particularly regarding solubility values in both PbLi and niobium, poses a significant challenge for the accurate modelling of tritium permeation in the PAV.

In the present work, a detailed CFD model of tritium transport within the PAV is developed using the OpenFOAM software, whose flexibility allowed to craft a set of custom boundary conditions and a solver that are not available in commercial CFD software. The model is validated against experimental data collected on the PAV mock-up at the TRIEX-II facility in ENEA Brasimone research centre. Furthermore, a sensitivity analysis is performed on the key physical parameters – evaluated at the same operating conditions of the experiments – confirming that tritium extraction is limited by the transport in the liquid metal, rather than by diffusion or surface effects in the membrane (i.e. the PAV is operating in Liquid Limited Regime). Consequently, the mass transfer coefficient $h_T$ of hydrogen in PbLi emerges as the main contributor to the model uncertainty and the driver for the permeation efficiency in the PAV. Therefore, the validated model is applied to compute $h_T$, showing a large overestimation by existing correlations, developed for water.

Future work forsees the development of a new correlation for the mass transfer coefficient in liquid metal, which could then be implemented in the system-level GETTHEM code, allowing an extrapolation of the mock-up test results to design the full-size EU DEMO PAV.

# Acknowledgements

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

Decarbonization is the single biggest challenge humanity has to face, crucial to safeguard our environment and social-economic stability against climate change. The energy production sector was accountable for roughly 40% of the global carbon dioxide emission worldwide in 2022 [1], which underscores the necessity to prioritize the development of low-emission energy production technologies. Nuclear fusion power plants are one of the most promising candidate.

Nuclear fusion is the reaction that powers our Sun, alongside with all other stars in the Universe: it is the process by which two light nuclei combine to form a single heavier one. The mass deficiency between *before* and *after* the interaction is converted into energy according to the formula $E = mc^2$. Fusion reactions take place in a state of matter called plasma, which is a extremely energetic and charged gas made of positive ions and free-moving electrons.

A fusion power plant is inherently safe because it lacks the chain reaction typical of fission reactors, mitigating the risk of a runaway process that, if not controlled, could potentially result in a core meltdown. Furthermore, the extreme temperatures and pressures required to sustain fusion reactions naturally limit and control the process: in the event of any operational issues, the reaction stops on its own. Overall, this technology offers a multitude of benefits: is low carbon emission, has a stable and continuous energy production, does not produce long-lived radioactive waste, does not pose a risk of nuclear weapons proliferation and, finally, promotes international scientific collaboration, exemplified by projects like ITER.

The current fusion road-map [2] foresees two main steps: first, the completion of ITER's construction, whose main goal is to study the plasma and prove a net gain in thermal energy of the order of $Q > 10$, and secondly the development of DEMO, a de-facto power plant which will produce electricy.

## 1.1 Energy production from nuclear fusion

To replicate the fusion reactions occurring in the Sun here on Earth it is necessary to engineer a machine capable of energy production: the most popular design up to now is the Tokamak.

A Tokamak is a nuclear fusion reactor that utilizes magnetic confinement to maintain plasma in a toroidal shape, preventing high-energy particles from impacting the chamber walls. The fuel is made of a roughly 50/50 mixture of deuterium ($D$) and tritium ($T$), the two isotopes of hydrogen. The fuel is injected into the plasma chamber and then heated until it reaches appropriate conditions for the $D, T$ nuclei to undergo fusion, producing a helium nucleus (also called $\alpha$ particle) and a neutron $n$ (figure (1.1a)).

$$T + D \longrightarrow \alpha(3.5MeV) + n(14.1MeV) \tag{1.1}$$

The large amounts of energy released in the fusion reactions appear in the form of kinetic energy of the end products. In particular, when the highly energetic neutron, not confined by the magnetic field, collides with the solid wall it transfers energy to the structural materials. A water cooling system extracts the heat, and then the hot water is employed to generate steam, subsequently converted into electricity using a turbine and generator (figure (1.1b)), as in a conventional power plant.



(a) Fusion reaction [3]



(b) Schematic of a fusion power plant, conventional cycle [4]
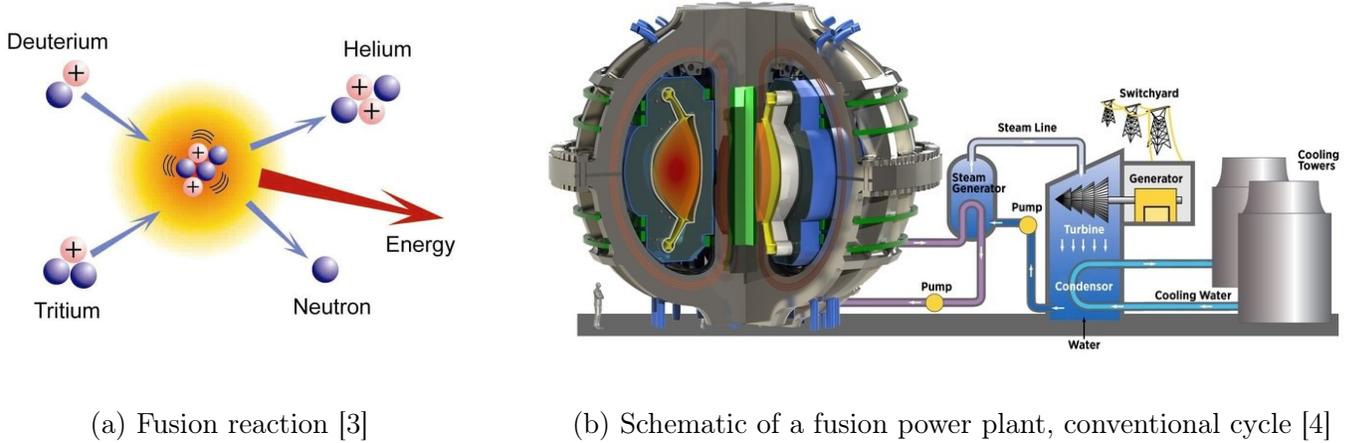
Figure 1.1

## 1.2 The tritium challenge

As noted earlier, fusion reactor are designed to use deuterium and tritium as fuel. Deuterium ($^2_1$D), a stable isotope of hydrogen, is commonly found in water both in form of HDO and $D_2O$, accounting for approximately 0.03% by mass of all the naturally occurring hydrogen in the oceans [5], making it a virtually boundless resource. On the other hand, tritium ($^3_1$T), a radioactive isotope of hydrogen, no longer occurs naturally on Earth. It has a half-life of $t_{1/2} = 12.3$ years, meaning that approximately 5.5% of a given sample of tritium will decay per year. Consequently, any tritium originally present on Earth, which dates back approximately $4 \times 10^9$ years, has already completely decayed. Tritium decays into helium-3 by $\beta$-minus decay:

$$^3_1\text{T} \longrightarrow {}^3_2\text{He} + e^- + \nu_e \tag{1.2}$$

Given the absence of naturally occurring tritium, fusion power plants must produce all the tritium they require.
ITER, over its anticipated 20-year operational period, aims to obtain the tritium fuel from the global inventory, given that it will not implement a breeding blanket. Specifically, it plans to secure 12.3 kg of tritium from CANDU production in Ontario [6]. On the other hand, DEMO is expected to require about 0.3 kg/day of tritium during nominal operation, and no sufficient external source of tritium exists that could supply that demand, since the current global production is around 2 kg/year. Nevertheless, a tritium stockpile is still necessary for the startup phase of DEMO, scheduled for the mid-2050s. However, even this tritium supply may be insufficient due to two significant factors. Firstly, potential delays in ITER could influence DEMO's timeline, while CANDU reactors undergo shutdown and tritium stockpile decay. Secondly, the prospect of multiple fusion reactors increases the demand for tritium: stocks would likely have to be shared.

In general, a fusion power plant designed to operate for 20-40 years cannot rely for the nominal operation phase on a stockpile of tritium, as it would decay significantly before being utilized. Additionally, for safety reasons, minimizing the path of a radioactive gas is crucial to reduce the risk of leakages. To address these concerns DEMO will implement a closed tritium fuel cycle.

### 1.2.1 Tritium fuel cycle

In a closed fuel cycle tritium is produced inside the fusion machine through the use of a Breeding Blanket (BB), reprocessed with the Tritium Extraction and Removal System (TERS) and finally injected back in the plasma chamber.
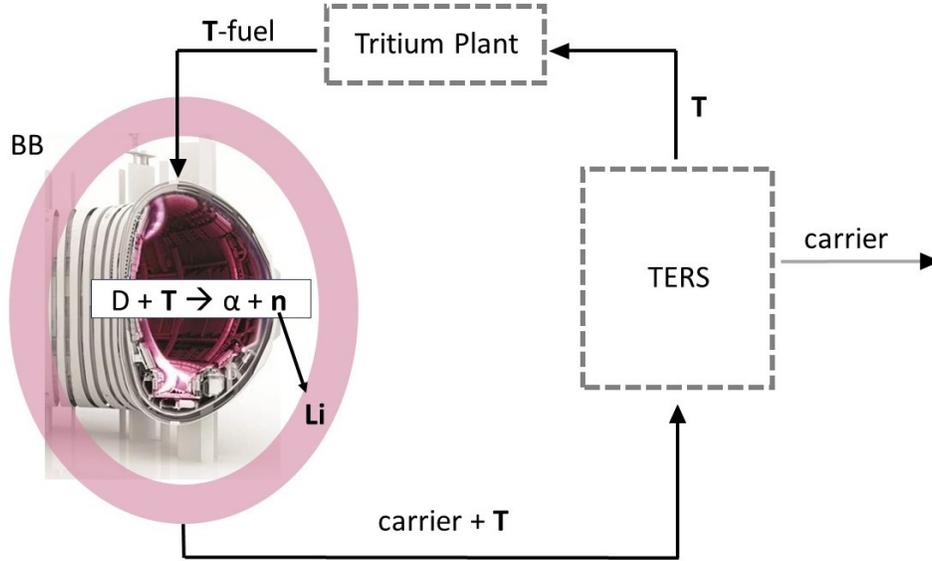


Figure 1.2: Schematic of the tritium fuel cycle

The BB is a structure that surrounds the plasma chamber and has the goal to absorb any high-energy neutron that is generated by the fusion reaction. That has three main functions: to extract the neutron energy for electricity production and to act as a cooling mechanism, to provide neutron shielding and finally, to breed tritium.

Tritium can be produced in the BB using lithium, which has two isotopes that are relevant to this process: lithium-6 (7.6% abundance) and lithium-7 (92.4% abundance) that can be integrated into the BB. As the high-energy neutron born with $E_n = 1.41 \times 10^7 eV$, escapes radially from the plasma, it penetrate into the BB material, where it has a higher likelyhood of interacting with $^7$Li (figure 1.3), producing tritium and helium-4, an inert gas, then the slowed down neutron ($n'$) can interact with $^6$Li, producing again tritium and helium-4.

$$^7_3\text{Li} + n \longrightarrow {}^4_2\text{He} + {}^3_1\text{T} + n' - 2.8 MeV$$
$$^6_3\text{Li} + n' \longrightarrow {}^4_2\text{He} + {}^3_1\text{T} + 4.8 MeV$$
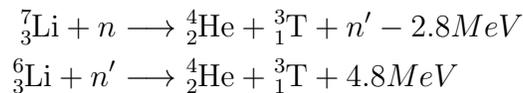
This mechanism allows, in principle, the production of two tritium atoms with a single neutron: an advantageous outcome from the neutron economics prospective.
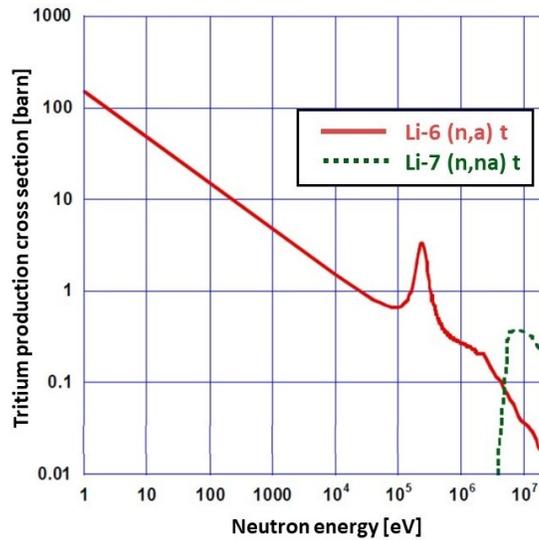
Figure 1.3: Lithium cross section for tritium production [7]

One of the main design parameters of the BB is the Tritium Breeding Ratio (TBR), defined as the ratio of the tritium breeded in the blanket to the tritium burned in plasma. The goal is to have a TBR>1.1, in order to guarantee the self-sufficiency of the reactor and provide a surplus of tritium for the power plant start-up, considering losses for tritium retention in the plant and radioactive decay.

Currently there are two main concepts for the BB design, each featuring a Test Blanket Module (TBM) that will undergo testing in ITER: the Helium-Cooled Pebble Bed (HCPB) BB and the Water-Cooled Lithium Lead (WCLL) BB. This work focus on the the latter. The WCLL BB will implement EUROFER as structural material, liquid metal eutectic Lithium-Lead (PbLi) as breeder, neutron multiplier and tritium carrier, and water at typical Pressurized Water Reactor (PWR) conditions as coolant. The water and PbLi manifolds are integrated with the primary heat transport system and the TERS, respectively (figure 1.4).
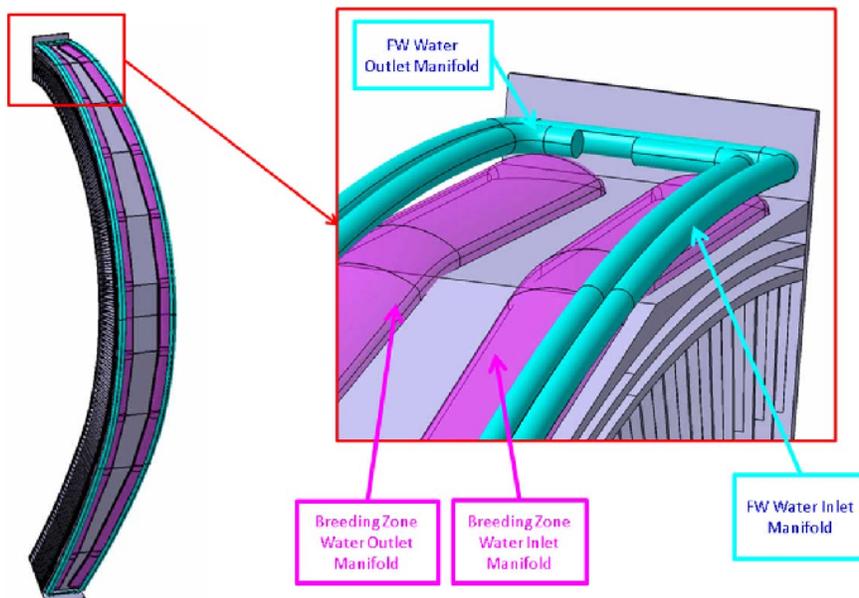


Figure 1.4: Breeding zone and first wall's channels for the WCLL BB [8].

Tritium is produced in the breeding zone through the $^6Li$(n,T) and $^7$Li(n,nT) reactions previously mentioned, with the lithium contained in the liquid metal. The flowing PbLi is then used as a carrier as it transports the tritium from the BB to the TERS.

The TERS is designed to extract tritium from the breeder and direct it to the Tritium Plant for final processing, converting it into a fuel form suitable for injection into the plasma chamber. The tritium extraction process involves separating the tritium from the lithium lead, and the efficiency of this process is crucial for achieving tritium self-sufficiency and maintaining a continuous fuel cycle. Effective tritium extraction from PbLi flows is crucial for PbLi-based BB concepts. The fast removal of dissolved tritium is essential to prevent excessive tritium inventories, ensuring safety, and to maintain tritium self-sufficiency in the plant.

Currently, various technologies are under investigation, and among the most promising for the WCLL BB design is the Permeator Against Vacuum (PAV) machine (see section 2.3), which is the main focus of this study. The PAV operates as an extractor with a shell-and-tube configuration, where the primary side contains the carrier liquid metal, and in the secondary side there is vacuum. The tritium dissolved in the liquid metal circulates through the inner pipes, permeating through a metallic (niobium) membrane driven by the pressure gradient.

## 1.3   Scope of the work

The tritium extraction process in the PAV needs to be investigated in detail, mainly because, according to current literature, the transport properties of tritium in PbLi and niobium have large uncertainties. In particular the solubility values, due to its difficulty in being measured experimentally, can vary up to four orders of magnitude. Furthermore, the condition of the membrane, such as oxidation and cleanliness, can significantly contribute to these uncertainties.

Hence, the main goal of this work is to have a better understanding on how tritium is transported in the PAV.

This entailed several key steps:

- to develop a 3D CFD model with OpenFOAM, in order to study how the tritium is transported both in the liquid metal and in the metallic membrane, taking into account surface processes.

- to validate the model against experimental data collected on the PAV mock-up at the TRIEX-II facility in ENEA Brasimone research centre, where two recent experimental campaigns took place using PbLi and hydrogen.

- to perform a sensitivity analysis over the main physical parameters of both the liquid metal and the membrane, to identify which is the limiting transport phenomena.

- to compute the mass transfer coefficient $h_T$, given the insight gained from the previous step, which confirmed the hypotesis that the PAV operates in Liquid Limited Regime (LLR).

These steps are necessary to develop a comprehensive understanding of tritium transport phenomena within the PAV and therefore properly design the EU DEMO extractor.

# Chapter 2

# Background

The following chapter provides some theoretical background on the mass transfer phenomena, justifies the choice of the employed CFD software and provide a description of the PAV machine.

## 2.1   Mass transfer

Mass transfer is the net movement of mass from one location to another, and it can occur in many processes, such as absorption, evaporation, precipitation or, what is of interest for this case, membrane filtration.

Mass transfer has two main transport mechanis: diffusion and advection. Diffusion is the random movement of atoms or molecules from a region of higher concentration to a region of lower concentration. Advection is the transfer of matter by the bulk fluid motion. The motion of fluids is described by the Navier-Stokes equations, which represent momentum balance for Newtonian fluids and make use of conservation of mass. In the incompressible form are:

$$\frac{\partial U}{\partial t} + \nabla \cdot (UU) = \nabla \cdot (\nu \nabla U) - \frac{1}{\rho}\nabla p + g \tag{2.1}$$

$$\nabla \cdot U = 0 \tag{2.2}$$

Where $U$ [m/s] is the velocity field, $p$ [Pa] is the pressure and $\rho$ [kg/s] and $\nu$ [m$^2$/s] are the fluid's density and kinematic viscosity.

The theoretical limit on how much mass can be transferred in a system is set by thermodynamic equilibrium, which is the state where the system's properties (i.e. temperature, pressure, chemical potential) are uniform. However, the actual rate of mass transfer is influenced by factors beyond thermodynamic equilibrium, including flow patterns and species diffusivities. The rate of mass transfer is quantified using mass transfer coefficients, typically expressed as dimensionless numbers like Reynolds (Re), Sherwood (Sh) and Schmidt (Sc) numbers. This coefficient help characterize and predict mass transfer behavior, aiding in the optimization of mass transfer operations in engineering contexts. A typical correlation for the mass transfer coefficient is in the form of:

$$h_T = \alpha \, Re^\beta \, Sc^\gamma \, D/d \tag{2.3}$$

Where $D$ [m$^2$/s] is the diffusivity, $d$ [m] is the pipe's inner diameter and $\alpha, \beta, \gamma$ are constants that can be derived experimentally or through a parametric study.

An analogy can be drawn between heat and mass transfer, due to similarities in their physical behaviour and mathematical descriptions. In fact Fourier's law for heat transfer and Fick's law for

mass transfer are very similar (eq. (2.5)), since they are both linear approximations of conserved quantities.

*Partial-differential equation (PDE)*     *Constitutive law*

$$\frac{\partial T}{\partial t} = \nabla \cdot (\alpha \nabla T) + S_T \qquad\qquad q = -k\nabla T \qquad\qquad \textbf{heat conduction} \qquad (2.4)$$

$$\frac{\partial C}{\partial t} = \nabla \cdot (D \nabla C) + S_C \qquad\qquad J = -D\nabla C \qquad\qquad \textbf{mass diffusion} \qquad (2.5)$$

Where $C$ [mol/m$^3$] is the concentration of a substance, $T$ [K] is the temperature. $D$ [m$^2$/s] and $\alpha$ [m$^2$/s] are the mass and thermal diffusivity. $k$ [W/m/K] is the thermal conductivity.

## 2.1.1 Passive scalar approach

The passive scalar approach is a simplification used to model the behavior of a scalar quantity, such as temperature or concentration, within a fluid flow. In this approach, the scalar is considered to be "passive" because it does not influence the flow itself; rather, it is passively advected by the velocity field of the fluid. This assumption is valid if the scalar quantity is present in low concentration, thereby having a negligible effect on the thermophysical properties of the fluid. This simplifies the governing equations, since it allows to decouple the scalar transport equation from the equations governing fluid flow, making it possible to solve first for the fluid field and subsequently for the scalar quantity. This results in significative lower computational complexity with respect to the multi-component approach.

The generic transport equation for the scalar quantity $\phi$ (advection-diffusion PDE) is:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (U\phi) = \nabla \cdot (\Gamma \nabla \phi) + S_\phi \qquad (2.6)$$

Which is composed, from left to right, by: the transient term indicating the rate of change in time of $\phi$; the advection term which describes how $\phi$ is trasported by the fluid flow; the diffusion term ($\Gamma$ is the diffusion coefficient) which describes how much $\phi$ dispurses into the surroundings; a source term.

## 2.2 Methodology: Computational Fluid Dynamics (CFD)

To achieve the primary objectives of the model, a detailed study of tritium transport within the liquid metal is imperative. Therefore, Computational Fluid Dynamics (CFD) software are used to simulate fluid flow and heat transfer phenomena on relatively small domains.

CFD software implement the control volume approach, a numerical technique for solving PDEs. The control volume approach subdivides the problem domain into a collection of discrete control volumes, through which the continuous medium (hypotizing small Knudsen number $K_n << 1$) flows. These control volumes collectively cover the entire domain of interest, and the closed surface enclosing the region is called control surface. Then the governing conservation equations (i.e., the Navier-Stokes equations) are expressed in integral form over each control volume and are numerically approximated both in time and space. Boundary conditions are applied at the interfaces between adjacent control volumes to enforce the physical behavior of the flow variables. The discretized equations for all control volumes form a coupled system of algebraic equations, typically nonlinear, and CFD software solves this system using iterative numerical techniques.

OpenFOAM is built around the finite volume method and uses implicit time discretization schemes to solve the transient term, allowing for larger time steps. In particular, the solver used as base for this problem is *chtMultiRegionFoam* [9]. The solver employs the two following pressure-velocity coupling algorithms to ensure mass conservation and pressure continuity: SIMPLE (Semi-Implicit Method for Pressure-Linked Equations) and PISO (Pressure-Implicit with Splitting of Operators). Moreover, the solver incorporates specialized algorithm to handle conjugate heat transfer between different regions, which ensure proper coupling between fluid flow and heat transfer.

To deal with turbulent flow the Reynolds-averaged Navier–Stokes (RANS) modeling approach is used, which describes the transport of averaged flow quantities, avoiding direct simulation of small-scale turbulent fluctuations. The RANS equations are the time-averaged of the Navier-Stokes equations (eq. 2.2). They implement the Reynolds decomposition technique, where a time-dependent quantity is expressed as superposition of its time-averaged and fluctuating component. The RANS equations are the following:

$$\frac{\partial U}{\partial t} + \text{div}(U\mathbf{U}) = -\frac{1}{\rho}\frac{\partial P}{\partial x} + \nu\text{div}(\text{grad}(U)) + \frac{1}{\rho}\left(\frac{\partial}{\partial x}\left(-\rho u'^2\right) + \frac{\partial}{\partial y}\left(-\rho u'v'\right) + \frac{\partial}{\partial z}\left(-\rho u'w'\right)\right)$$

$$\frac{\partial V}{\partial t} + \text{div}(V\mathbf{U}) = -\frac{1}{\rho}\frac{\partial P}{\partial y} + \nu\text{div}(\text{grad}(V)) + \frac{1}{\rho}\left(\frac{\partial}{\partial x}\left(-\rho u'v'\right) + \frac{\partial}{\partial y}\left(-\rho v'^2\right) + \frac{\partial}{\partial z}\left(-\rho v'w'\right)\right)$$

$$\frac{\partial W}{\partial t} + \text{div}(W\mathbf{U}) = -\frac{1}{\rho}\frac{\partial P}{\partial z} + \nu\text{div}(\text{grad}(W)) + \frac{1}{\rho}\left(\frac{\partial}{\partial x}\left(-\rho u'w'\right) + \frac{\partial}{\partial y}\left(-\rho v'w'\right) + \frac{\partial}{\partial z}\left(-\rho w'^2\right)\right)$$

Where $\mathbf{U}$ is the mean velocity vector with mean components U, V and W, and fluctuating components u', v' and w'. $P$ is the pressure, $\rho$ is the fluid density and $\nu$ is the fluid kinematic viscosity. The RANS-based modeling approach requires two additional equations in order to achieve closure for the unknowns. The main families of two-equations turbulence models are $k$-$\epsilon$ and $k$-$\omega$. They are models that solves transport equations for the turbulent kinetic energy $k$ and the turbulent dissipation rate $\epsilon$, while the turbulence frequency is defined as $\omega = \epsilon/k$. The *chtMultiRegionFoam* solver supports various turbulence models, including $k$-$\omega$, whose transport equations are:

$$\frac{\partial(\rho k)}{\partial t} + \text{div}(\rho k\mathbf{U}) = \text{div}\left(\mu + \frac{\mu_t}{\sigma_k}\right)\nabla k + P_k - \beta^*\rho k\omega$$

$$\frac{\partial(\rho\omega)}{\partial t} + \text{div}(\rho\omega\mathbf{U}) = \text{div}\left(\mu + \frac{\mu_t}{\sigma_\omega}\right)\nabla\omega + \gamma_1\left(2\rho S_{ij} \cdot S_{ij} - \frac{2}{3}\rho\omega\frac{\partial U_i}{\partial x_j}\delta_{ij}\right) - \beta_1\rho\omega^2$$

Where $\mu_t$ is the eddy viscosity and $S_{ij}$ is the mean component of the rate of deformation of a fluid element. The values of the constants $\sigma_k$, $\sigma_\omega$, $\gamma_1$, $\beta_1$, and $\beta$ are typically determined empirically [10].

## 2.3   Technology: Permeator Against Vacuum (PAV)

The Permeator Against Vacuum (PAV) is the chosen technology for the extractor in the WCLL BB of the DEMO reactor. It serves the dual purpose of extracting tritium from the flowing lithium-lead and supplying it to the tritium plant for final processing. The PAV operates by permeating tritium through a membrane that separates the flowing PbLi and vacuum, driven by a pressure gradient to facilitate extraction.

The tritium permeation involves the following steps (figure 2.1), starting from the liquid side: 1) initially the tritium is dissolved in monoatomic form in the PbLi, where it is transported both through diffusion and advection processes; 2) as it reaches the inner membrane , tritium is attracted by the metallic surface and through adsorption and absorption processes it is transported into the

bulk of the membrane; 3) driven by the concentration gradient the atomic tritium diffuses in the solid until it reaches the outer surface, facing the vacuum; 4) here it recombinates into bi-atomic molecules and leave the membrane.
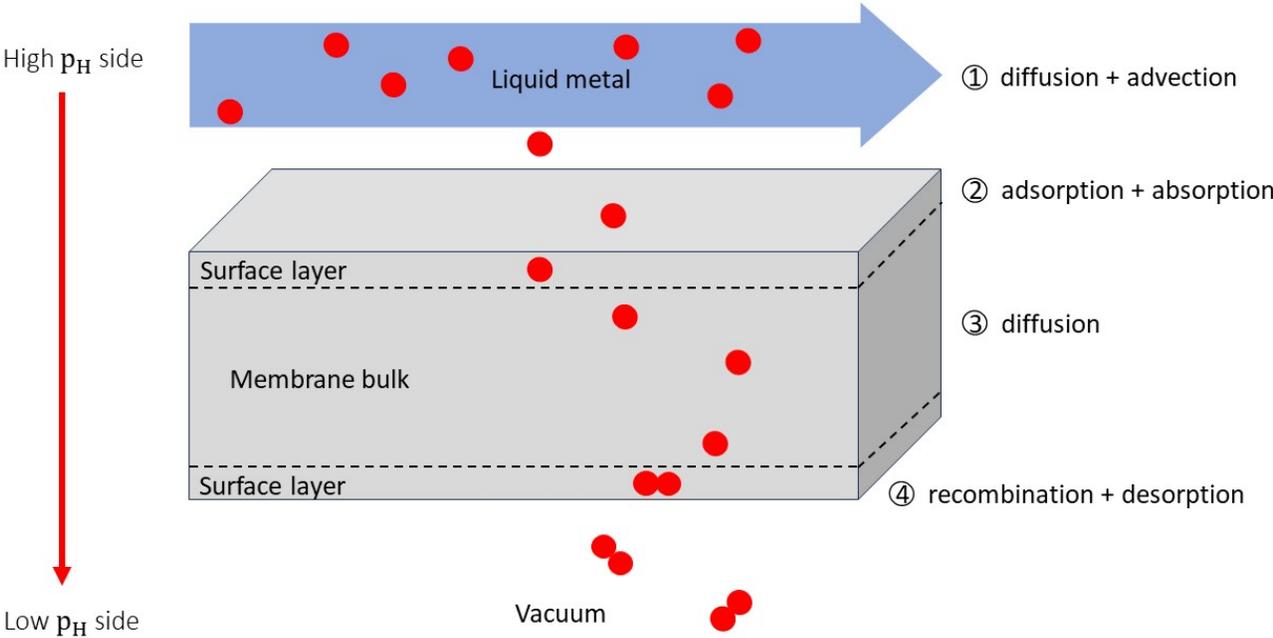


Figure 2.1: Tritium transport process in the PAV

The PAV mock-up (figure (2.2)) adopts a tube-and-shell heat exchanger structure. It consists of a cylindrical vessel housing 16 niobium "U"-shaped pipes, representing the membranes for hydrogen permeation. The 16 pipes are divided into two passages, each one consisting of 8 parallel pipes of different length: 3 short (S), 3 medium (M), and 2 long (L).
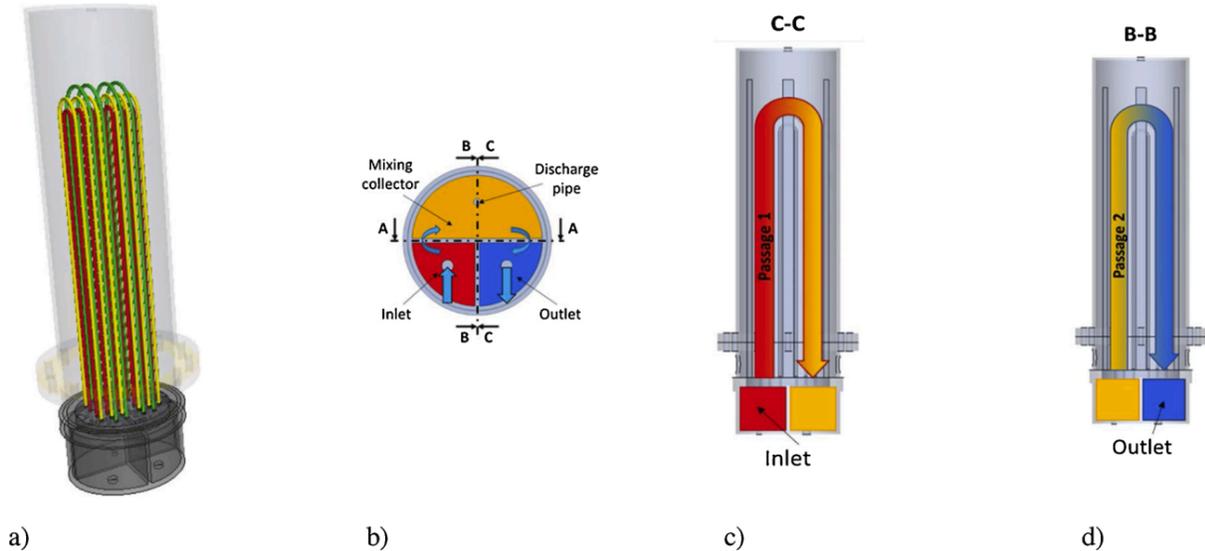


Figure 2.2: Schematic of the PAV's mock-up [11]

Between the first and second passage there is a collector, divided into three parts, which facilitates the distribution of LiPb into the niobium pipes. The choice of niobium is justified by its high permeability and low susceptibility to oxidation. The niobium pipes are welded to an F22 plate, chosen for its corrosion resistance in LiPb environments. The vessel maintains a medium vacuum, while LiPb flows through the niobium pipes.

To maintain a constant temperature for all the niobium pipes the PAV mock-up employs four double-tube infrared lamps made of impermeable quartz as heating system. The lamps, positioned inside the vessel symmetrically, facilitate temperature control.

To gain insights into the behavior of the PAV, the Humrickhouse analytical model [12] is implemented as a first approximation. Figure (2.3) illustrates the axial concentration profile from the inlet (left) to the outlet (right) for both passages. For the first passage, a dummy value of $C_{in,I} = 1 : \mathrm{mol/m^3}$ is assumed as the inlet concentration. Additionally, assuming perfect mixing within the collector, the inlet concentration of the second passage is computed as the arithmetic average of the outlet concentrations of the first passage.
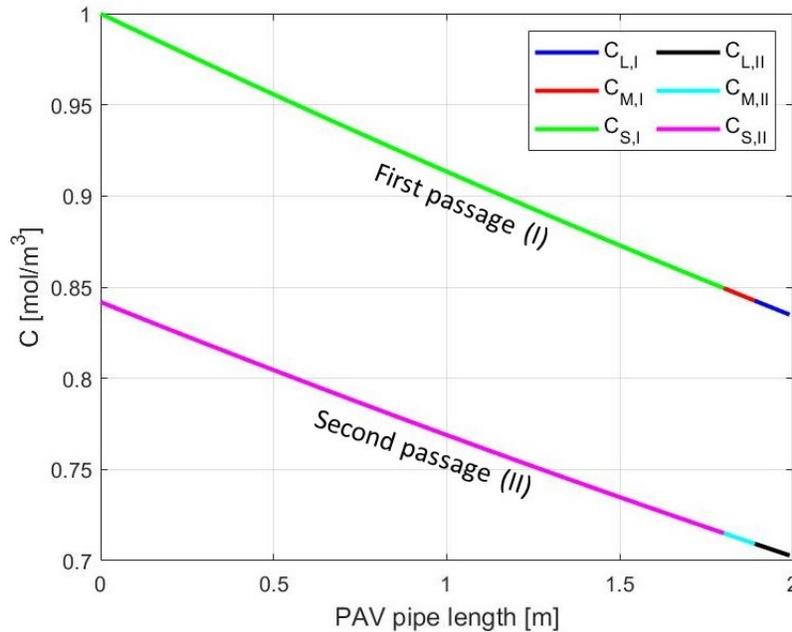


Figure 2.3: Axial profile of the concentration along the PAV piping system

The concentration decays exponentially along the axis for both passages, and it decreases with the pipe length, in fact at the outlet $C_S > C_M > C_L$.

10

# Chapter 3

# Model

The following chapter will provide a detailed description of the model developed for tritium transport in the liquid metal and the membrane. The software used is OpenFOAM-v9 with custom solver and boundary conditions, using as base *chtMultiRegionFoam* and *turbulentTemperatureCoupledBaffleMixed*, respectively.

## 3.1  Hypothesis and governing equations

The main hypothesis are the following:

- The fluid flows in the axial direction with mean velocity $\vec{U}(z) = U\hat{e}_z$, while tritium concentration $C(r, z)$ is transported in the axial direction by the liquid bulk, and in the radial direction through diffusion processes due to the pressure gradient.

- Passive scalar approach: the concentration is considered as a property of the flow. This approach is valid under the assumption that the concentration is low enough so to not influence the flow field.

- Transient model that reach steady-state condition.

- Isothermal condition, with uniform and constant temperature: $T(r, z) = T_{op}$.

- Incompressible flow: $\nabla \rho = 0$

- Negligible effect of gravity force.

- No volumetric source/sink.

- No particle accumulation in the membrane: $J_{in} = J_{out}$ in steady state.

The governing equations are the following:

**For the fluid flow**
Mass conservation equation

$$\nabla U = 0 \tag{3.1}$$

Momentum conservation equation

$$\frac{\partial U}{\partial t} + \nabla \cdot (UU) = \nu \nabla^2 U - \frac{1}{\rho} \nabla p \qquad (3.2)$$

Energy conservation equation

$$T = T_{op} \qquad (3.3)$$

**For concentration**
Scalar transport equation

$$\frac{\partial C}{\partial t} + \nabla \cdot (CU) = D \nabla^2 C \qquad (3.4)$$

## 3.2    Computational domain

The computational domain is selected to minimize computational cost while at the same time ensuring a reliable representation of the overall system. Specifically, the decision was made to model only the linear segment of the U-pipe with medium (M) length. The exclusion of the 180° bend is not expected to have a major impact on the results. Subsequently, exploiting the axial symmetry observed in the geometry, fluid flow and physical phenomena, the domain was further reduced to a wedge-shaped region within the original pipe. This region, highlighted in green in figure (3.1), is characterized by dimensions detailed in table (3.1).

| diameter $d$ [m] | thickness $t$ [m] | length $L_{wedge}$ [m] | angle $\alpha$ [°] |
|---|---|---|---|
| $9.2 \times 10^{-3}$ | $0.4 \times 10^{-3}$ | 0.834 | 10 |

Table 3.1: Geometrical parameters of the wedge
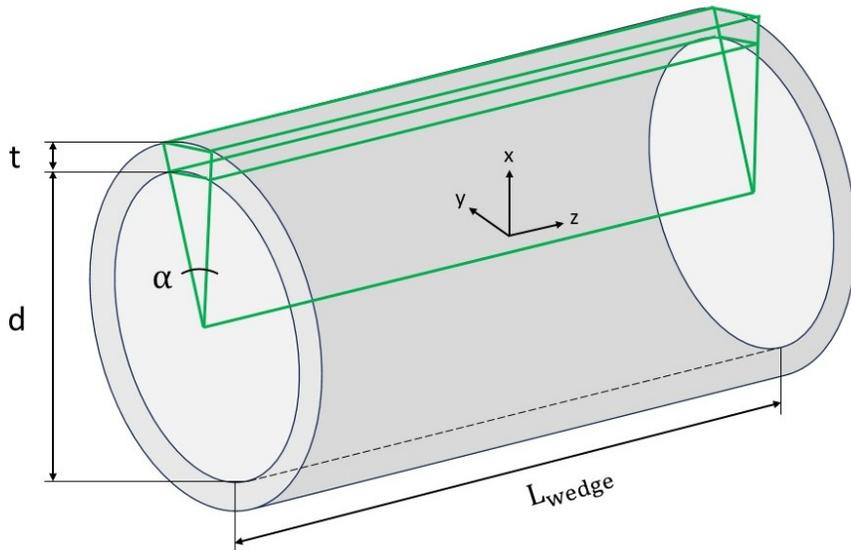


Figure 3.1: Schematic representation of the computational domain

The wedge geometry is made of only one cell across the circumferential direction (figure (3.2a)), which reduces the relevant dimensions to only two: axial and radial. Despite being a 2D geometry, the wedge is a common choice for turbulence modeling in OpenFOAM. This specific domain, coupled with the proper boundary condition called *wedge*, proves effective in replicating turbulence, an inherently 3D phenomena. The *wedge* boundary condition, applied to the inclined side patches, has the advantage of allowing for the rotational flow around the axis. This property arises from not assuming zero normal components at the boundary, allowing flow or scalar flux across it. This is the main difference between a *wedge* boundary condition and a *symmetry* boundary condition. Referring to figure (3.2b) the *wedge* BC transforms cell values $\Psi_P$ to the patch faces using a rotational transformation tensor $R$ by $\Psi_f = G(R_f, \Psi_P)$, where $R_f$ defines a rotation between the unit vector $n$ in the circumferential direction at the cell centre and the unit face normal vector $n_f$ by $n_f = R_f \cdot n$.
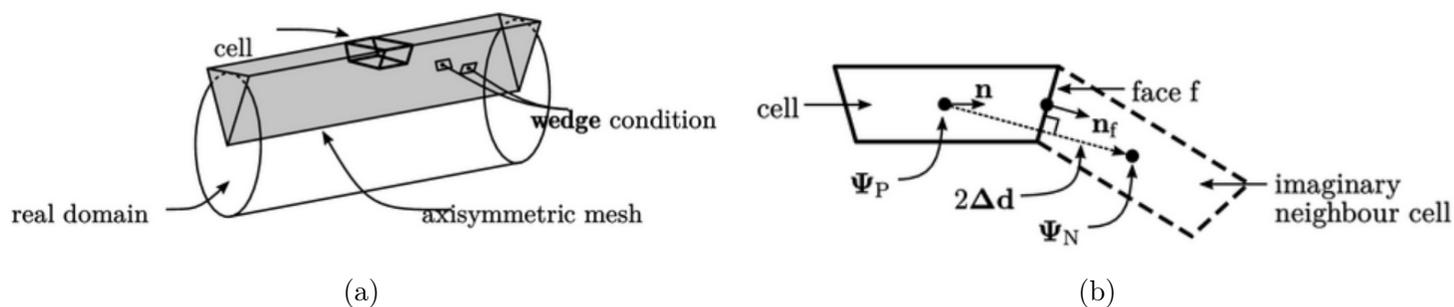


Figure 3.2: How the wedge geometry is able to reproduce 3D phenomena [13]

Therefore, despite being geometrically 2D, the wedge still computes the solution on the lateral cell faces, allowing for 3D modelling.

## 3.3 Mesh

The mesh was build using the *blockMesh* utility, which decompose the domain geometry into a set of hexahedral blocks. Each block of the geometry is identified by 8 vertices, one for each corner. OpenFOAM operates always in 3D cartesian geometry and each block has a local coordinate system $(x_i, y_i, z_i)$, which must be right-handed.

The blocks utilized for the wedge geometry are depicted in figure (3.3), with their corresponding vertex numbering. There are 4 blocks in total: 2 for the liquid region and 2 for the solid one. The blocks constituting the liquid region feature a pair of vertices collapsed on each other to replicate the wedge shape, whereas the solid blocks are standard hexahedra.

The mesh is split in the axial direction with $L_1 + L_2 = L_{wedge}$, where $L_1 = 10$cm. This division allows for mesh customization only in the entry region, which may require an higher mesh refinement. Notably, the faces at $z = L_1$ (in both the solid and liquid regions) share the same vertices, indicating that this face is not a boundary. In contrast, the faces at $x = r_{in}$, between the liquid and solid regions, have distinct vertices. This signify them as physical boundaries, defining two distinct regions with varying properties: *liquidMetal* as the fluid region and *membrane* as the solid region.
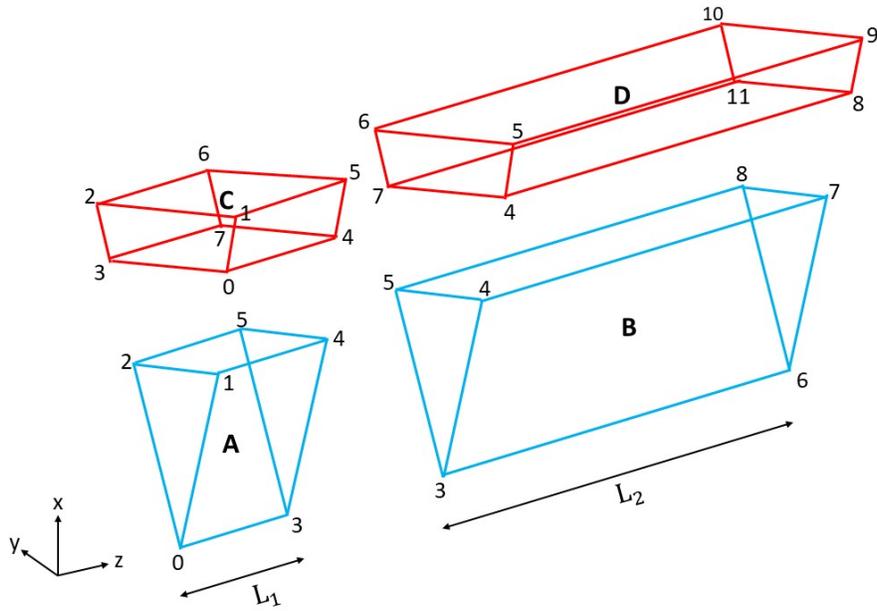
Figure 3.3: Schematic representation of the mesh's blocks with respective vertices.

The implementation of a mesh in two regions requires two *blockMeshDict*: one in the *liquidMetal* region and one in the *membrane* region. The two patches on the shared boundary are called *liquidWall* and *innerWall* (figure (3.4)), and they are connected by the *mappedWall* boundary type, whose implementation is reported in appendix (A.2). This is the base over which the boundary condition *myRobinBC* (detailed in section (3.5.1)) is build on.
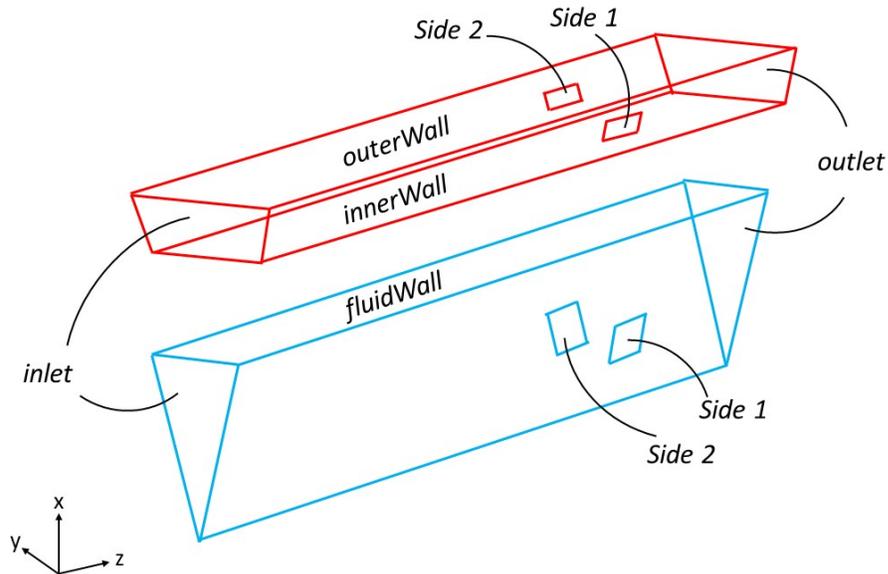


Figure 3.4: Schematic representation of the mesh's patches. Liquid region in blue, solid region in red.

Subsequently the number of cells and the cell expansion ratio must be specified in each direction

14

for each block. The mesh reported in figure (3.5) correspond to the parameters specified in table (3.2).

| Block | $N_{cells,r}$ | $N_{cells,z}$ | $\delta_e/\delta_s$ |
|---|---|---|---|
| A | 25 | 1300 | (0.1 1 7) |
| B | 25 | 900 | (0.1 1 1) |
| C | 5 | 1300 | (3 1 7) |
| D | 5 | 900 | (3 1 1) |

Table 3.2: Block's parameters

The cell expansion ratio (implemented as *simpleGrading*) enables the mesh to be refined in specified directions. It is defined as the ratio of the width of the end cell along one edge of a block to the width of the start cell along that same edge: $\delta_e/\delta_s$. In particular the first two blocks (A and C, corresponding to the entry region) have grading in the axial direction of $\delta_e/\delta_s = 10$. The axial thickness of the last cell is then taken as reference for the other two blocks (B and D) and kept constant ($\delta_e/\delta_s = 1$). In the radial direction again the grading is done in such a way that the last cell of the liquid region has similar radial thickness as the first cell in the solid region.
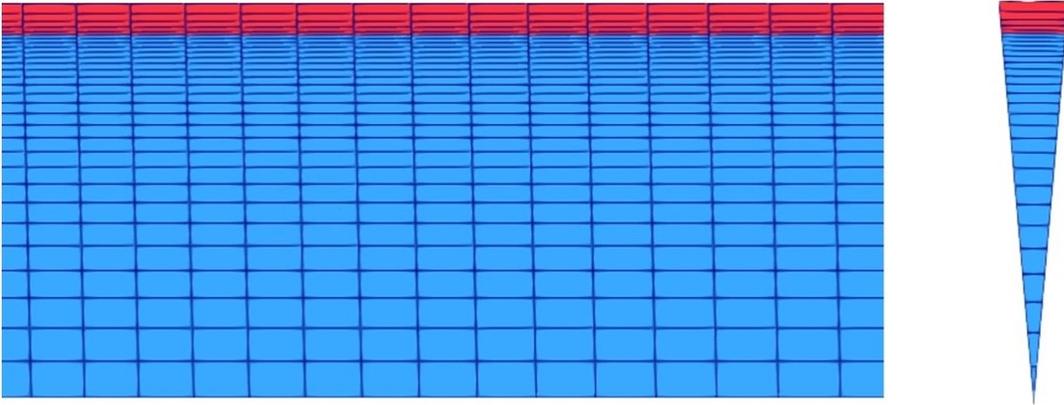


Figure 3.5: Mesh of the wedge. Side view on the left, front view on the right.

As previously mentioned, meshes for axisymmetric problems typically feature only one cell along the circumferential direction. This approach introduces a geometric error because the faces normal to the radial direction are flat. This error reduces with decreasing wedge angle: in practice, the error becomes negligible for an angle of $\alpha = 1°$. The recommendation is to use a wedge with $\alpha \leq 5°$.

## 3.4 Solvers

The solver is based on the *ChtMultiRegionFoam* solver present in OpenFOAM [9]. This is a solver commonly used for turbulent flow with conjugate heat transfer between regions. The solver adopts a segregated solution approach for fluid-solid coupling: Initially, fluid equations are solved using the previous iteration's solid temperatures to set boundary conditions for fluid temperatures. Subsequently, a similar procedure is employed for the solid until convergence is achieved. For each fluid

region the compressible Navier Stokes equation are solved, applying the following algorithm: 1) update the density using the continuity equation, ensuring mass conservation; 2) solve the momentum equation, where a velocity field is computed, typically not satisfying the continuity equation; 3) solve the energy equation, where the temperature is computed with the coupling approach previously described; 4) solve the pressure equation, constructed using continuity and momentum equations to ensure mass conservation and correct the velocity field; 5) correct the density using the new pressure field.

Utilizing the passive scalar approach, the concentration C can be treated as a scalar property of the fluid flow, and therefore it only requires the implementation of the scalar transport equation (eq. 3.4).This process involves the following steps:

- Create the new scalar field for the concentration as volScalarField "CFluid" in *createFluid-Fields.H* for every cells [i] of the mesh

- define the new variable volScalarField& C = CFluid[i] in *setRegionFluidFields.H*

- create the new equation CEqn.H, as reported in appendix (A.2).

- Make sure the solver reads from the thermophysicalProperties dictionary the mass diffusivity D (instead of the thermal conductivity $k$), and specify the units as dimensionedScalar.

- Add the new equation in the PISO and PIMPLE loops as #include "CEqn.H" in *solveFluid.H*.

- Repeat the process for the solid region.

- Compile the solver.

The turbulence model used is the $SST\ k - \omega$ model, which is the one conventionally coupled with the *chtMultiRegionFoam* solver. It combines a $k - \omega$ model next to the wall and a $k - \epsilon$ one in the free stream region. This gives a more accurate description of the flow next to the wall in case of adverse pressure gradients, with respect to $k - \epsilon$ models and the results are less dependent on the free stream (arbitrary) values that are required for the original (Wilcox) $k - \omega$. The presence of blending functions facilitates a smooth transition between the two models. The implementation is OpenFOAM also requires to specify the *eddyDiffusivity* as model and the turbulent Prandtl number $Pr_t = 0.85$.
As wall treatment it is implemented the *alphatWallFunction* on the *fluidWall* patch.

## 3.5   Boundary conditions

The boundary conditions used in the model are listed in table (3.3), referring to the patches in figure (3.4). This specific combination of boundary conditions yields the most stable convergence.

| Liquid | | | | Solid | |
|---|---|---|---|---|---|
| patch | p | U | C | patch | C |
| *inlet* | zeroGradient | codedFixedValue | fixedValue | *inlet* | zeroGradient |
| *outlet* | fixedValue | pressureInletOutletVelocity | inletOutlet | *outlet* | zeroGradient |
| *side 1* | wedge | wedge | wedge | *side 1* | wedge |
| *side 2* | wedge | wedge | wedge | *side 2* | wedge |
| *liquidWall* | fixedFluxPressure | noSlip | myRobinBC | *innerWall* | myRobinBC |
| - | - | - | | *outerWall* | myVacuumBC |

Table 3.3: Boundary conditions

In the context of fluid flow, these conditions follow the conventional practice of alternating between fixing the pressure p and the velocity U at the inlet and outlet. Specifically, the velocity is fixed with a uniform profile at the inlet, while the pressure is set equal to $p = 1 \times 10^5$Pa at the outlet section. The concentration C is also fixed at the inlet, and it is the main input parameter of the model.

The boundary conditions for the concentration at the liquid/solid and solid/vacuum interfaces (respectively called *myRobinBC* and *myVacuumBC*) are more challenging instead. The necessity for specialized boundary conditions, developed for this specific problem, was the primary motivation for adopting OpenFOAM. In the next two sections will delve into the detailed formulation of these conditions.
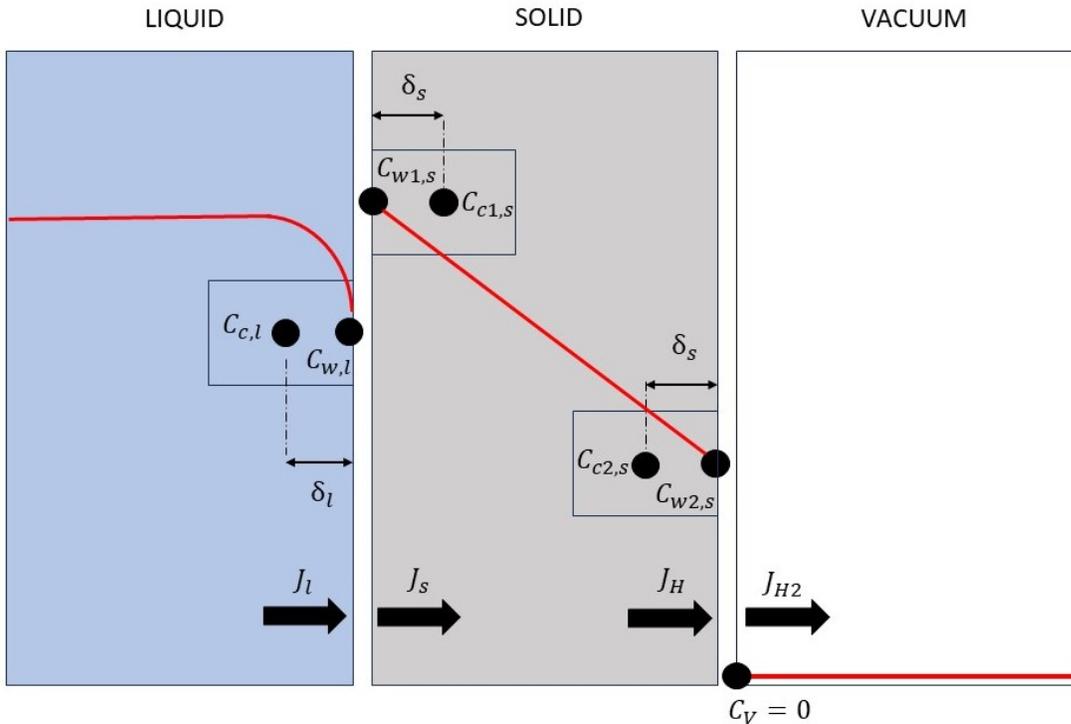


Figure 3.6: 2D computational domain representation.
The pedix "c" refers to the cell-center values while "w" refers to the wall value, "l" and "s" refer to the liquid and solid respectively, and finally "1" and "2" refers to the liquid/solid interface and solid/vacuum interface.

For reference in subsequent calculations, a schematic 2D axial-radial representation of the computational domain is provided in figure (3.6), where the PbLi flow in the liquid region goes from top to bottom, while the concentration flux goes from left to right.

### 3.5.1 Liquid/solid interface

The concentration at the liquid/solid interface is computed by modifying the *turbulentTemperatureCoupledBaffleMixed* boundary condition, which is originally used for temperature and describe the heat transfer across two different regions. This boundary condition is based on the *mixed* BC, which compute the value at the interface as:

$$X_b = f \ X_{ref} + (1 - f)[X_x + \frac{Grad_{ref}(X)}{deltaCoeff}] \tag{3.5}$$
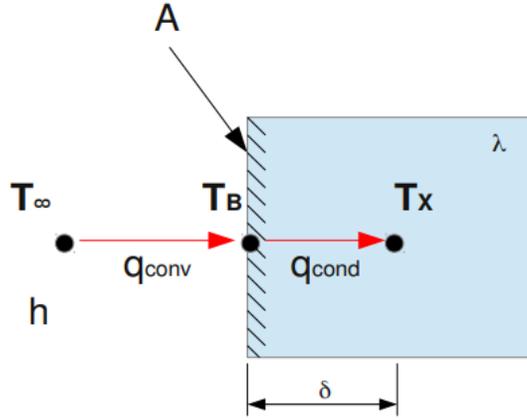


Figure 3.7: Scheme for the OpenFOAM *mixed* BC [14]

Where $X_{ref}$ is a reference value for the variable at boundary, $X_x$ is the value of the variable in the cell center, $Grad_{ref}(X)$ is the reference gradient of the variable, $deltaCoeffs$ is the inverse of the face center to cell center distance ($\frac{1}{\delta}$) and $f$ is a weighted factor that defines the BC type, being a Dirichlet BC when f=1, a Neumann BC when f=0 and a Robin BC when $0 < f < 1$.
The original derivation for the temperature starts from the two following equations:

$$\begin{cases} q_{cond,l} = -q_{cond,s} & \text{conservation of heat flux} \\ T_l = T_s & \text{continuity of temperature} \end{cases} \tag{3.6}$$

The first equation becomes

$$\frac{k_l}{\delta_l}(T_{c,l} - T_l) = -\frac{k_s}{\delta_s}(T_{c,s} - T_s) \tag{3.7}$$

For simplicity of notation we define $\delta_l = \frac{k_l}{\delta_l}$ and $\Delta_s = \frac{k_s}{\delta_s}$, and substituting the second equation and solving for $T_l$ we obtain:

$$T_l = T_{c,s} \ (\frac{\Delta_s}{\Delta_s + \delta_l}) + T_{c,l} \ (\frac{\delta_l}{\Delta_s + \delta_l}) = T_{c,s} \ (\frac{\Delta_s}{\Delta_s + \delta_l}) + T_{c,l} \ (1 - \frac{\Delta_s}{\Delta_s + \delta_l}) \tag{3.8}$$

Which is implemented in OpenFOAM by specifing the following 3 entries:

18

- $X_{ref} = T_{c,s}$

- $f = \frac{\Delta_s}{\Delta_s + \delta_l}$

- $Grad_{ref}(X) = 0$

If concentration replaces temperature as the variable of interest it becomes necessary to account for the jump at the interface. The derivation for this new boundary condition starts from the two following equations:

$$\begin{cases} J_l = -J_s & \text{conservation of particle flux} \\ \frac{C_l}{K_{s,l}} = \frac{C_s}{K_{s,s}} & \text{equilibrium of partial pressure (with Sievert's law)} \end{cases} \tag{3.9}$$

The first equation becomes

$$\frac{D_l}{\delta_l}(C_{c,l} - C_{w,l}) = -\frac{D_s}{\delta_s}(C_{c1,s} - C_{w1,s}) \tag{3.10}$$

Again for simplicity of notation we define $\delta_l = \frac{D_l}{\delta_l}$ and $\Delta_s = \frac{D_s}{\delta_s}$, and substituting the second equation we obtain:

$$C_{w,l} = C_{c1,s}\left(\frac{\Delta_s}{\Delta_s \frac{K_{s,s}}{K_{s,l}} + \delta_l}\right) + C_{c,l}\left(\frac{\delta_l}{\Delta_s \frac{K_{s,s}}{K_{s,l}} + \delta_l}\right) \tag{3.11}$$

Where:

- $X_{ref} = C_{c1,s}$

- $f = \frac{\Delta_s}{\Delta_s \frac{K_{s,s}}{K_{s,l}} + \delta_l}$

However the coefficient for $C_{c,l}$ is not in the form $(1 - f)$. In order to re-write it in a way that resembles eq. (3.5), the following term is defined as $\frac{\delta_l}{\Delta_s \frac{K_{s,s}}{K_{s,l}} + \delta_l} = (1 - f)^*$ and impose that:

$$(1 - f)[C_{c,l} + Grad_{ref}(X)\,\delta_l] = (1 - f)^* C_{c,l} \tag{3.12}$$

Consequently, it is derived:

- $Grad_{ref}(X) = \frac{C_{c,l}}{\delta_l}\left(\frac{(1-f)^*}{(1-f)} - 1\right) = \frac{C_{c,l}}{\delta_l}\left(\frac{\delta_l}{\delta_l + \Delta_s\left(\frac{K_{s,s}}{K_{s,l}} - 1\right)} - 1\right)$

Which is the third and last entry required by the OpenFOAM boundary condition. As a check, it can be verified that if $\frac{K_{s,s}}{K_{s,l}} = 1$ the previous formulation for the temperature is obtained.
The implementation in OpenFOAM is provided in appendix (A.2).

### 3.5.2 Solid/vacuum interface

The concentration at the solid/vacuum interface is computed by imposing the particle conservation between the atomic and molecular fluxes.

$$J_H = -2\,J_{H_2} \tag{3.13}$$

19

Which means that the flux of atomic hydrogen in the solid membrane is equal to twice the flux of molecular hydrogen that recombines before detaching from the membrane and diffusing into the vacuum.

$$J_s = -2\, J_{rec} \tag{3.14}$$

$$\frac{D_s}{\delta_s}\,(C_{c2,s} - C_{w2,s}) = -2\, k_r\, C_{w2,s}^2 \tag{3.15}$$

From which we obtain:

$$C_{w2,s} = \frac{\Delta_s}{4\, k_r}\left(1 \pm \sqrt{1 - 8\,\frac{k_r\, C_{c2,s}}{\Delta_s}}\right) \tag{3.16}$$

However, the implementation in OpenFOAM of the above formula is rather complex. Therefore the implementation is done by steps and, starting from the same balance (eq. 3.13), we can define:

$$J_{ricombination} = k_r\, C_{c2,s}^2$$
$$J_s = -2\, J_{ricombination}$$
$$\nabla C = -J_s/D_s$$
$$C_{w2,s} = C_{c2,s} + \nabla C/\delta_s$$

Which is done directly in the *0/membrane/C* file using *codedFixedValue*, as reported in appendix (A.2).

# Chapter 4

# Results

The following chapter will present the main results and applications derived from the utilization of the previously described model.

## 4.1    Hydraulic results

In sections (4.1) and (4.2), some qualitative results, including velocity, pressure, and concentration profiles, will be showcased. The computational domain employed consists of a wedge with a length of $L = 0.834$ m, representing the straight portion of the U-pipe. The properties utilized are outlined in table (4.2), corresponding to the experimental conditions of test (2.1) as detailed in the table (4.1).

| $T$ [K] | $p$ [Pa] | $\dot{m}$ [kg/s] |
|---|---|---|
| 723 | 241 | 1.20 |

Table 4.1: Conditions used in sections (4.1),(4.2)

| $K_{s,PbLi}$ [mol/m$^3$/Pa$^{1/2}$] [15] | $K_{s,Nb}$ [mol/m$^3$/Pa$^{1/2}$] [16] | $D_{PbLi}$ [m$^2$/s] [17] | $D_{Nb}$ [m$^2$/s] [18] |
|---|---|---|---|
| $1.10 \times 10^{-3}$ | $1.00 \times 10^{2}$ | $4.85 \times 10^{-9}$ | $9.12 \times 10^{-9}$ |

Table 4.2: Main properties used in sections (4.1),(4.2)

The Reynolds number for the liquid metal flow in the wedge is computed as:

$$Re = \frac{D\,\bar{U}\,\rho}{\mu} \tag{4.1}$$

Where $D$[m] is the pipe inner diameter, $\bar{U}$ [m/s] is the average liquid velocity and $\rho$[kg/s], $\mu$[Pa s] are the lithium-lead density and dynamic viscosity.
In the experimental conditions $Re = 1.3 \times 10^4 > Re_{cr}$ ensuring that the flow is in turbulent conditions.
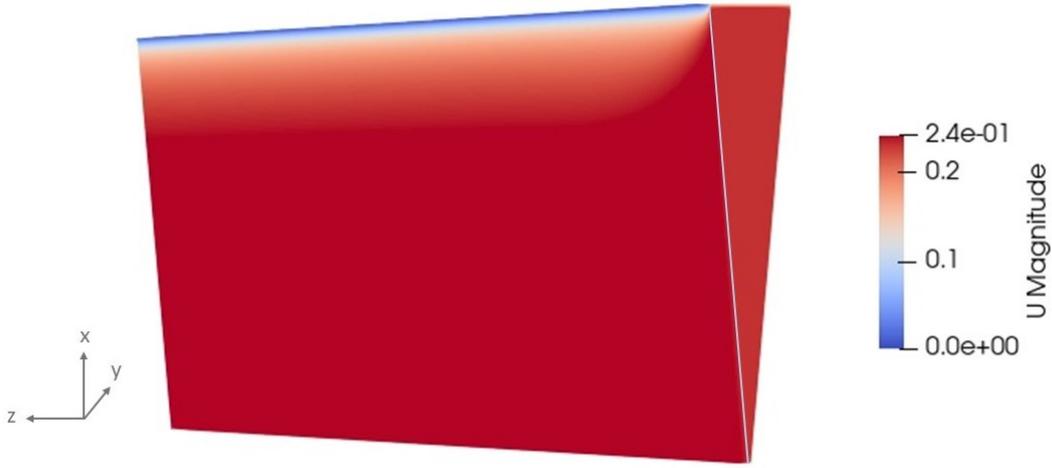
Figure 4.1: Velocity distribution in the wedge

Since the computational domain includes the inlet of the pipe, it is essential to consider the impact of the entry region. The radial velocity profile is evaluated at eight distinct axial coordinates, as illustrated in figures (4.2a) and (4.2b), where $r = 0$ mm correspond the the pipe centerline, while $r = 4.6$ mm to the pipe wall.



(a) Not normalized

(b) Normalized
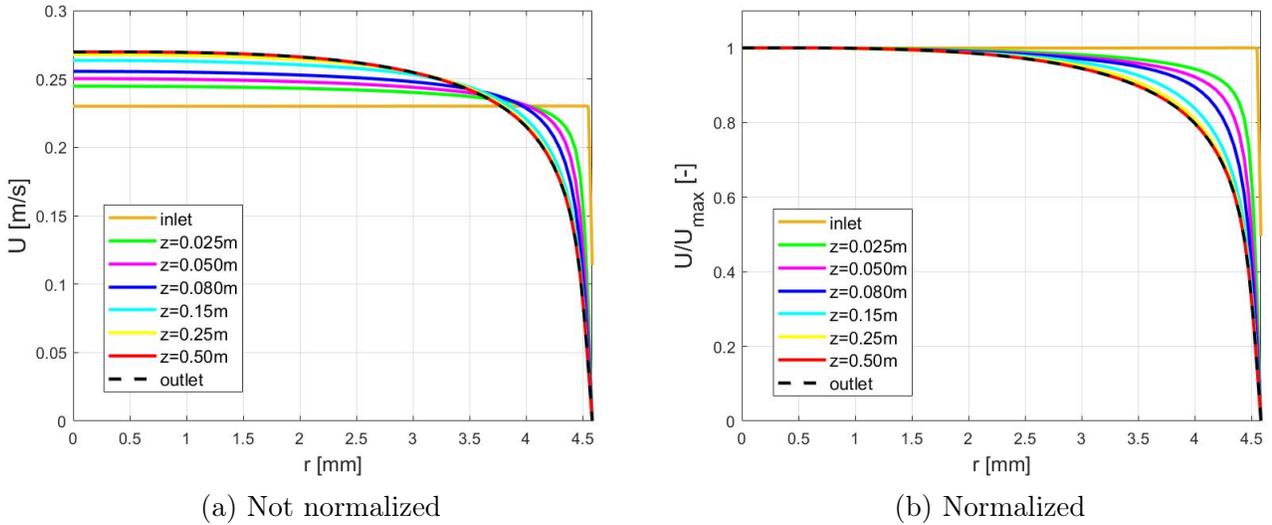
Figure 4.2: Radial velocity profiles at different axial coordinates

These figures clearly depict the uniform inlet profile imposed as a boundary condition (but extracted at $z = 1 \times 10^{-6}$ m). As the profiles evolve, they transition into the characteristic turbulent velocity profile: a flat top with a sharp gradient near the walls. The length of the entry region is approximately $L_{ER} = 0.30$m, beyond which the profile remains constant.

The pressure drop can be computed analytically as:

$$\Delta p = f \, \frac{L}{D} \frac{\rho \, \bar{U}^2}{2} \tag{4.2}$$

Where the friction factor $f = 0.028$ is found using the Moody's diagram with the entries $(Re, \epsilon) = (1.3 \times 10^4, 3 \times 10^{-5})$, resulting in $\Delta p = 665$ Pa. Instead the numerical pressure drop along the centerline is $p_{inlet} - p_{outlet} = \Delta p = 652$ Pa, resulting in a 2% error.

22

(a)                                           (b)
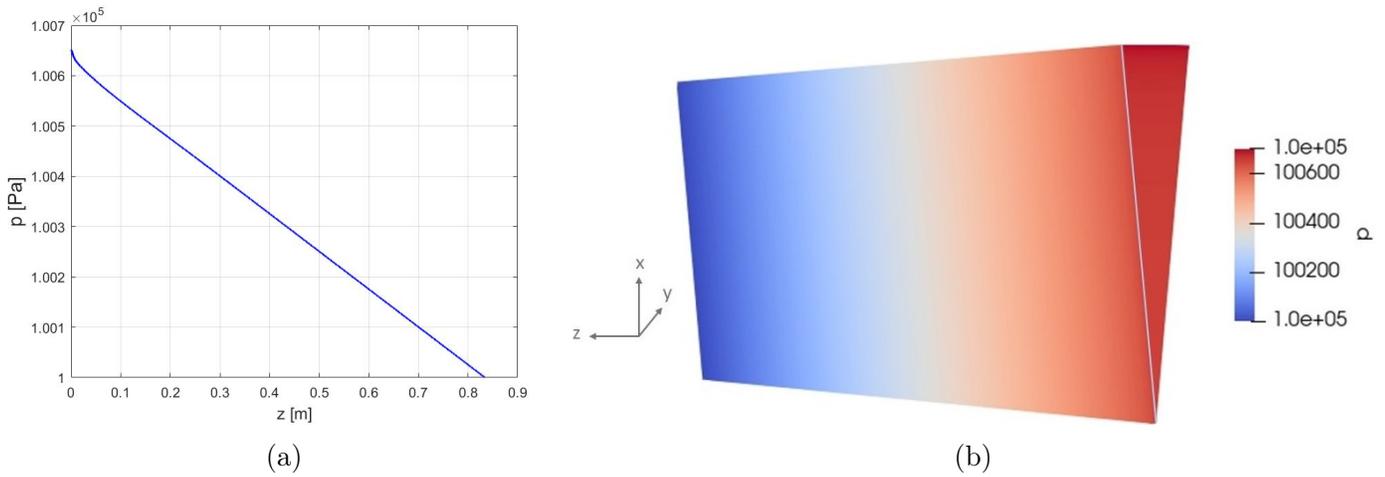
Figure 4.3: Pressure axial profile

## 4.2 Concentration

The concentration $C\,[\text{mol/m}^3]$ map is shown in figure (4.4).
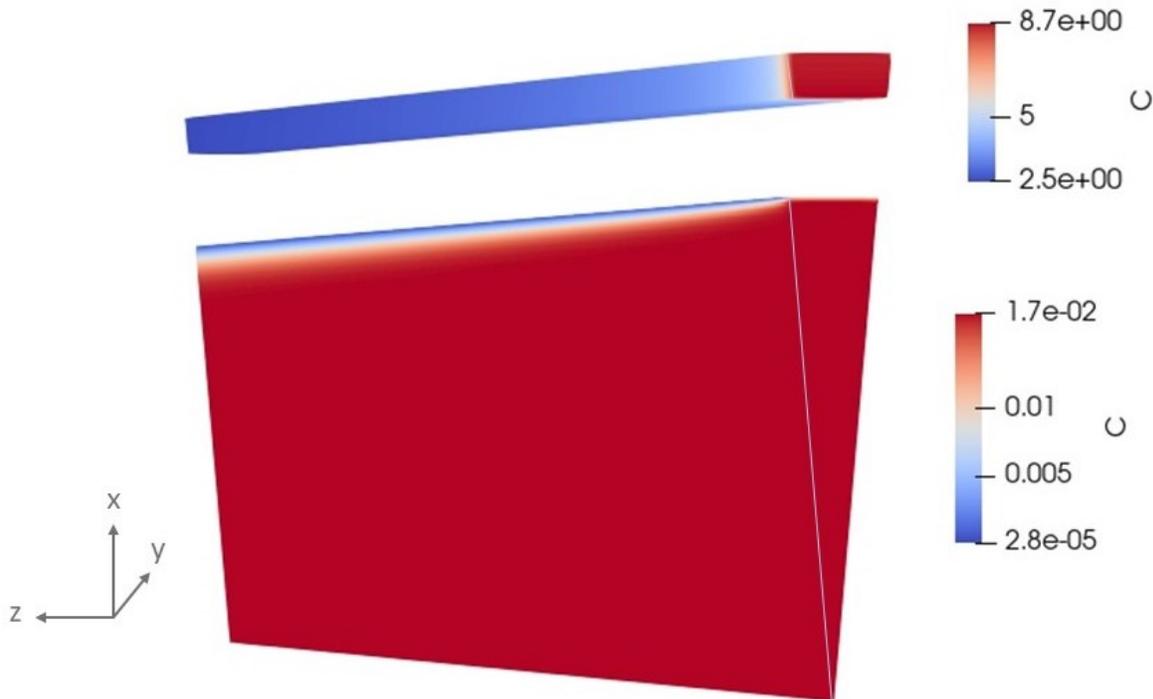


Figure 4.4: Concentration distribution in the wedge

The concentration radial profile is shown in figure (4.5a), while figure (4.5b) provides a zoom-in on the interface. The profile resemble what we expect from literature (figure (4.6)): the bulk concentration is uniform until very close to the wall, where it sharply decreases up $C_{w,l} \simeq 3 \times 10^{-5}$. In the solid region instead, complying with Fick's law, a linear profile emerges.

23
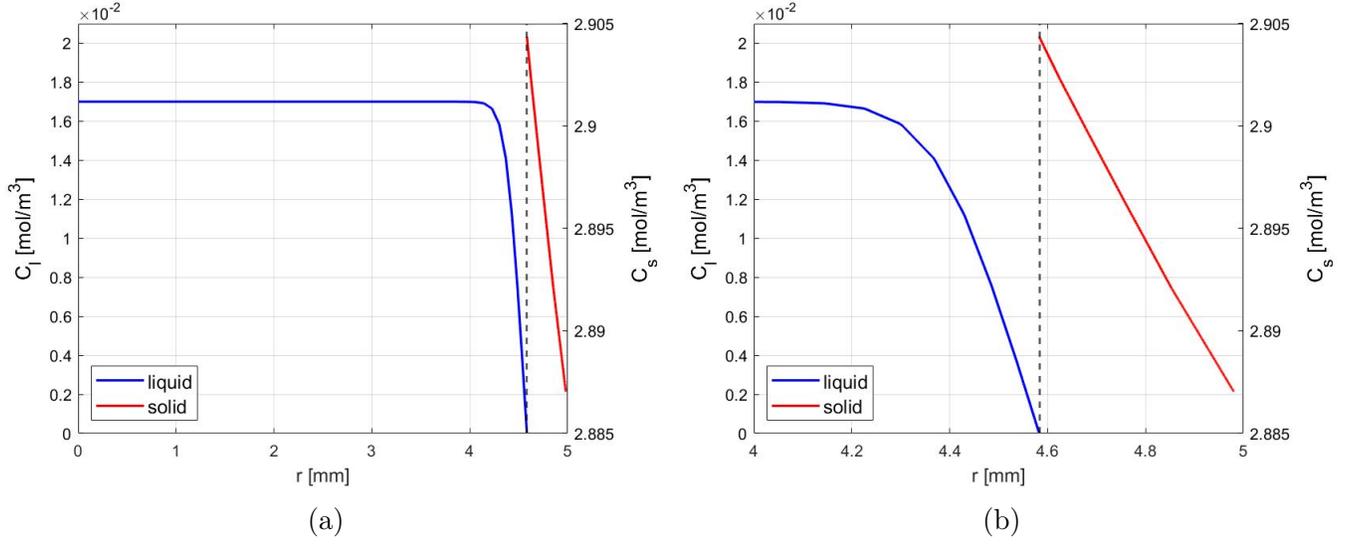
Figure 4.5: Concentration radial profile in the liquid and solid region
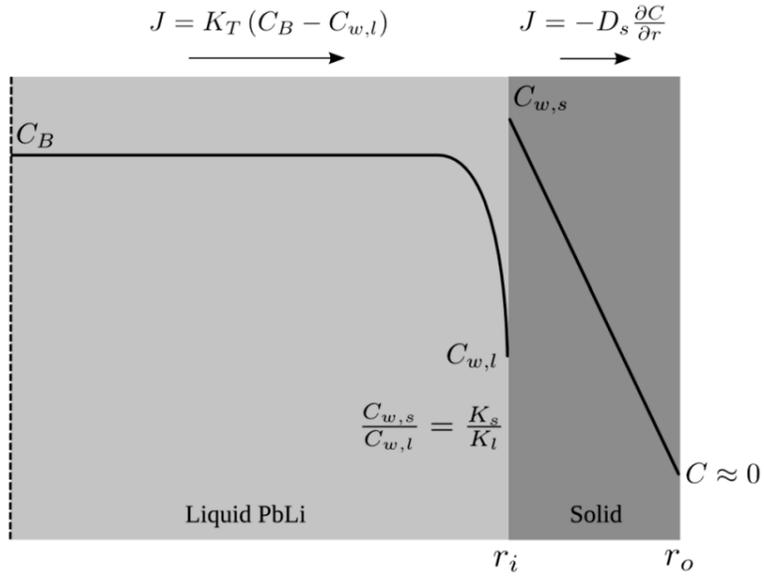


Figure 4.6: Schematic of tritium transport processes in the radial direction in a vacuum permeator tube, from Humrickhouse analytical model [12]

At the interface, the concentration jump $\Delta C = C_{w,s} - C_{w,l}$ becomes evident. The discontinuity arises from the difference in solubility between the two regions. Assuming continuity of partial pressure and applying Sieverts law:

$$\begin{cases} p_{H,l} = p_{H,s} \\ C = K_s \sqrt{p_H} \end{cases} \tag{4.3}$$

Results in:

$$\frac{C_{w,s}}{C_{w,l}} = \frac{K_{s,s}}{K_{s,l}} \simeq 9 \times 10^4 \tag{4.4}$$

This indicates that the actual concentration jump is several orders of magnitude higher than one depicted in the plots, as they have been scaled for better comprehension.

24

In figure (4.7) are represented the radial profiles of concentration in the liquid region at different axial coordinates. It is evident that the concentration never reaches a fully developed condition, as it keeps changing axially.



(a) Not normalized



(b) Normalized
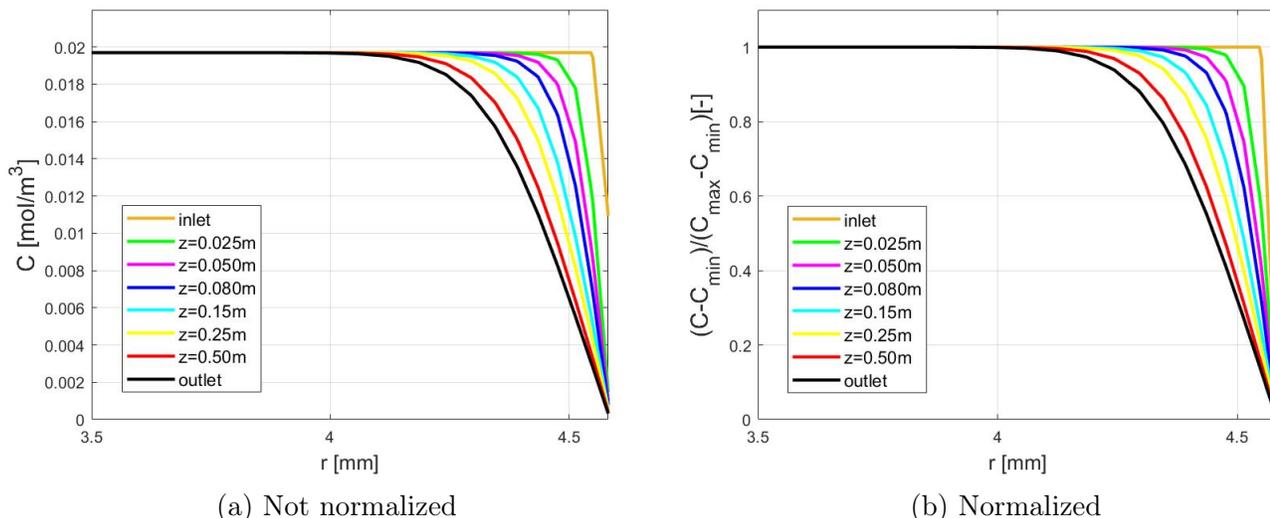
Figure 4.7: Concentration radial profile in the liquid region at different axial coordinates

## 4.3 Verification and Validation

Verification and validation serve as the primary procedures to assess the accuracy, reliability, and predictive capability of the numerical model.

Verification involves ensuring the correctness of numerical algorithms implemented in the code, confirming that it operates as intended. Validation assesses the degree to which a model is an accurate representation of the real world for the intended application. The objective is to confirm the model's predictive capabilities by comparing its outcomes with experimental data. In the validation process, the agreement between model output and experimental data is measured by quantifying the difference between them. This agreement is then expressed in a statistical way, typically as an error accompanied by a confidence interval.

### 4.3.1 Grid convergence study

The verification process involves quantifying the discretization error in the simulations through a progressive refinement of the mesh. As the cells decrease in size, the spatial discretization error is expected to asymptotically approach zero. In this study, the Grid Convergence Index (GCI) method is employed, which involves conducting simulations on two or more progressively finer grids. The GCI serves as a parameter indicating the deviation of the solution from its asymptotic value and is calculated following the procedure recommended in [19]:

$$GCI_{21} = \frac{F_s \, |\epsilon_{21}|}{r_{21}^c - 1} \tag{4.5}$$

Where $F_s$ is the Factor of Safety and it is assumed equal to 1.25, $\epsilon_{21}$ is the relative error between the two finest meshes, c is the observed convergence order, and $r_{21}$ is the ratio between $h_2$ and $h_1$. The latter requires to define the average mesh cell size $h$ [m] as:

$$h = \left(\frac{V_{meshed}}{N_{cells,tot}}\right)^{1/3} \tag{4.6}$$

Where $V_{meshed}$ is the meshed liquid volume and $N_{cells,tot}$ is the number of liquid cells. The relative numerical error is given by $u_{num} = GCI/k$ with expansion factor $k = 1.15$.

To conduct this convergence study, four meshes have been created for the liquid region, with main parameters summarized in table 4.3.

| Mesh | $N_{cells,r}$ | $N_{cells,z,1}$ | $N_{cells,z,2}$ | $N_{cells,tot}$ | $h$ [m] | average $\delta_z/\delta_r$ |
|------|------|------|------|------|------|------|
| 1 | 15 | 800 | 530 | $2.0 \times 10^4$ | $4.2 \times 10^{-4}$ | 2.0 |
| 2 | 25 | 1300 | 900 | $5.5 \times 10^4$ | $3.0 \times 10^{-4}$ | 2.1 |
| 3 | 40 | 2200 | 1500 | $1.5 \times 10^5$ | $2.3 \times 10^{-4}$ | 2.0 |
| 4 | 65 | 3300 | 2200 | $3.6 \times 10^5$ | $1.6 \times 10^{-4}$ | 2.1 |

Table 4.3: Mesh refinement parameters

The refinement level was selected to meet three specific criteria: ensuring that the ratio $r = h_{coarse}/h_{fine}$ remains above 1.3, maintaining a nearly constant average cell aspect ratio ($\delta_z/\delta_r$) and have a smooth transition between blocks with different mesh refinement. The latter requirement is implemented by ensuring $\delta_{e,1} = \delta_{s,2}$ for all the meshes. The reported values result in the following ratios for consecutive pairs of meshes: $r_{12} = 1.40, r_{23} = 1.39, r_{34} = 1.34$.

The parameters used to estimate numerical uncertainty are:

- Pressure drop $\Delta p$: calculated as $p_{in} - p_{out}$, where pressure is computed as the surface average over the respective inlet and outlet areas.

- Average wall concentration $\bar{C}_w$: computed as the surface average over the liquid wall area.

- Velocity at the pipe centerline in the outlet section $U(r = 0, z = z_{out})$: this point is assumed to represent the location of maximum velocity.
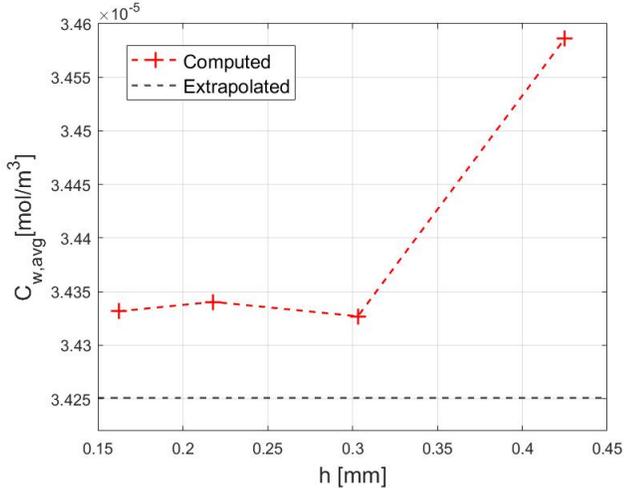
(a)  Average wall concentration
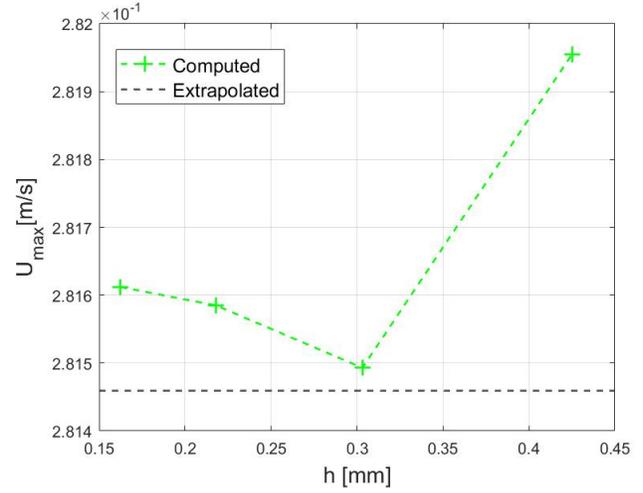


(b)  Maximum velocity
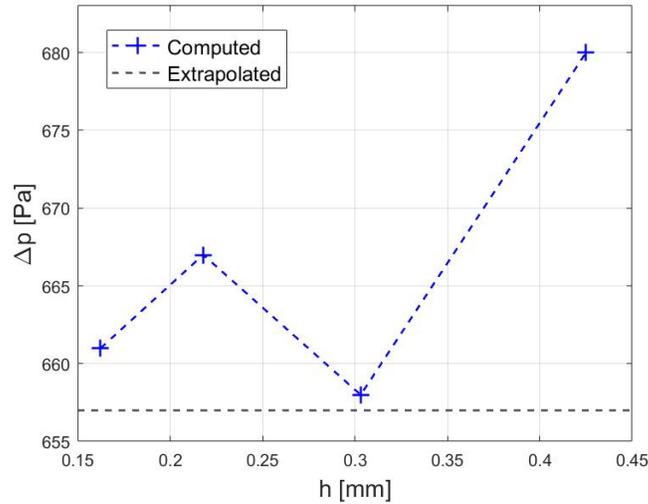


(c)  Pressure drop

Figure 4.8: GCI analysis performed on three parameters, as function of the average cell size $h$.

The mesh selected is the third finest one, which yield an error with respect to the extrapolated value of: 0.15% for $\Delta p$, 0.01% for $\bar{C}_w$ and 0.22% for $U_{max}$.

## 4.3.2   Experimental campaigns

The TRIEX-II experimental facility operates at ENEA Brasimone research center (Camugnano, Italy) and it is designed to investigate tritium extraction processes in a PAV mock-up. In 2023 two experimental campaigns were conducted using hydrogen as permeating gas and lithium-lead as carrier, to study the behavior of tritium extraction.

The first experimental campaign is made of three tests (test 1,2 and 3) performed at higher temperature T=450°C and different partial pressures. The second experimental campaign instead is made of four tests (test 4,5,6 and 7) performed at a lower temperature T=355°C and different hydrogen partial pressures. All the results are listed in tables (4.4) and (4.5) respectively.

| Test | | $p_{\text{in}}$ [Pa] | $T$ [°C] | $\dot{m}$ [kg/s] | $K_{s,PbLi}$ [mol/m³/Pa$^{1/2}$] | $\Phi$ [mol/s] |
|---|---|---|---|---|---|---|
| 1 | 1 | $170 \pm 0.52$ | $451 \pm 2.56$ | $1.19 \pm 0.10$ | $8.22 \times 10^{-4} \pm 1.59 \times 10^{-4}$ | $4.58 \times 10^{-8} \pm 2.75 \times 10^{-9}$ |
|   | 2 | $170 \pm 0.39$ | $452 \pm 2.24$ | $1.19 \pm 0.10$ | $8.71 \times 10^{-4} \pm 1.63 \times 10^{-4}$ | $4.67 \times 10^{-8} \pm 2.58 \times 10^{-9}$ |
| 2 | 1 | $241 \pm 0.83$ | $452 \pm 2.26$ | $1.19 \pm 0.10$ | $1.46 \times 10^{-4} \pm 2.76 \times 10^{-5}$ | $3.00 \times 10^{-8} \pm 1.67 \times 10^{-9}$ |
|   | 2 | $239 \pm 0.60$ | $450 \pm 2.21$ | $1.19 \pm 0.10$ | $1.32 \times 10^{-4} \pm 2.36 \times 10^{-5}$ | $2.56 \times 10^{-8} \pm 1.18 \times 10^{-9}$ |
|   | 3 | $241 \pm 0.82$ | $450 \pm 2.25$ | $1.18 \pm 0.10$ | $8.83 \times 10^{-5} \pm 1.96 \times 10^{-5}$ | $1.73 \times 10^{-8} \pm 1.51 \times 10^{-9}$ |
|   | 4 | $244 \pm 0.53$ | $451 \pm 2.21$ | $1.18 \pm 0.10$ | $6.00 \times 10^{-5} \pm 1.10 \times 10^{-5}$ | $1.15 \times 10^{-8} \pm 5.90 \times 10^{-10}$ |
| 3 | 1 | $359 \pm 0.86$ | $451 \pm 2.22$ | $1.20 \pm 0.10$ | $7.87 \times 10^{-5} \pm 1.21 \times 10^{-5}$ | $3.08 \times 10^{-8} \pm 6.84 \times 10^{-10}$ |
|   | 2 | $366 \pm 1.33$ | $451 \pm 2.22$ | $1.20 \pm 0.10$ | $9.72 \times 10^{-5} \pm 1.78 \times 10^{-5}$ | $3.75 \times 10^{-8} \pm 1.88 \times 10^{-9}$ |
|   | 3 | $366 \pm 0.90$ | $451 \pm 2.21$ | $1.20 \pm 0.10$ | $1.01 \times 10^{-4} \pm 2.05 \times 10^{-5}$ | $3.95 \times 10^{-8} \pm 2.80 \times 10^{-9}$ |

Table 4.4: First experimental campaign [20]

| Test | $p_{\text{in}}$ [Pa] | $T$ [°C] | $\dot{m}$ [kg/s] | $K_{s,PbLi}$ [mol/m³/Pa$^{1/2}$] | $\Phi$[mol/s] |
|---|---|---|---|---|---|
| 4 | $114 \pm 0.33$ | $356 \pm 1.84$ | $1.28 \pm 0.10$ | $2.10 \times 10^{-5} \pm 4.52 \times 10^{-6}$ | $3.37 \times 10^{-9} \pm 3.00 \times 10^{-10}$ |
|   | $110 \pm 0.33$ | $356 \pm 1.82$ | $1.28 \pm 0.10$ | $2.02 \times 10^{-5} \pm 3.00 \times 10^{-6}$ | $2.98 \times 10^{-9} \pm 6.94 \times 10^{-11}$ |
| 5 | $129 \pm 0.40$ | $355 \pm 1.84$ | $1.22 \pm 0.10$ | $1.22 \times 10^{-5} \pm 4.28 \times 10^{-6}$ | $2.00 \times 10^{-9} \pm 4.36 \times 10^{-10}$ |
|   | $133 \pm 0.38$ | $355 \pm 1.83$ | $1.22 \pm 0.10$ | $1.61 \times 10^{-5} \pm 2.73 \times 10^{-6}$ | $2.90 \times 10^{-9} \pm 1.13 \times 10^{-10}$ |
| 6 | $184 \pm 0.47$ | $355 \pm 1.86$ | $1.22 \pm 0.10$ | $1.36 \times 10^{-5} \pm 2.60 \times 10^{-6}$ | $3.00 \times 10^{-9} \pm 1.80 \times 10^{-10}$ |
|   | $184 \pm 0.49$ | $355 \pm 1.85$ | $1.22 \pm 0.10$ | $1.33 \times 10^{-5} \pm 3.12 \times 10^{-6}$ | $2.90 \times 10^{-9} \pm 3.03 \times 10^{-10}$ |
| 7 | $226 \pm 0.48$ | $355 \pm 1.84$ | $1.22 \pm 0.10$ | $1.13 \times 10^{-5} \pm 2.01 \times 10^{-6}$ | $2.96 \times 10^{-9} \pm 1.41 \times 10^{-10}$ |
|   | $225 \pm 0.47$ | $355 \pm 1.87$ | $1.22 \pm 0.10$ | $1.02 \times 10^{-5} \pm 1.54 \times 10^{-6}$ | $2.64 \times 10^{-9} \pm 5.47 \times 10^{-11}$ |

Table 4.5: Second experimental campaign [20]

Since each test is made of multiple measures, an averaged value was computed to use it as representative. As example, in fig 4.9 is shown the average permeated flux for test 2: in blue are the results for each measure, while in red is the average value used as representative, which was computed as:

$$E[x] = \frac{1}{N} \sum_{i=1}^{N} x_i \tag{4.7}$$

$$\sigma[x] = \sqrt{\frac{1}{N} \sum_{i=1}^{N} |x_i - E[x]|^2} \tag{4.8}$$
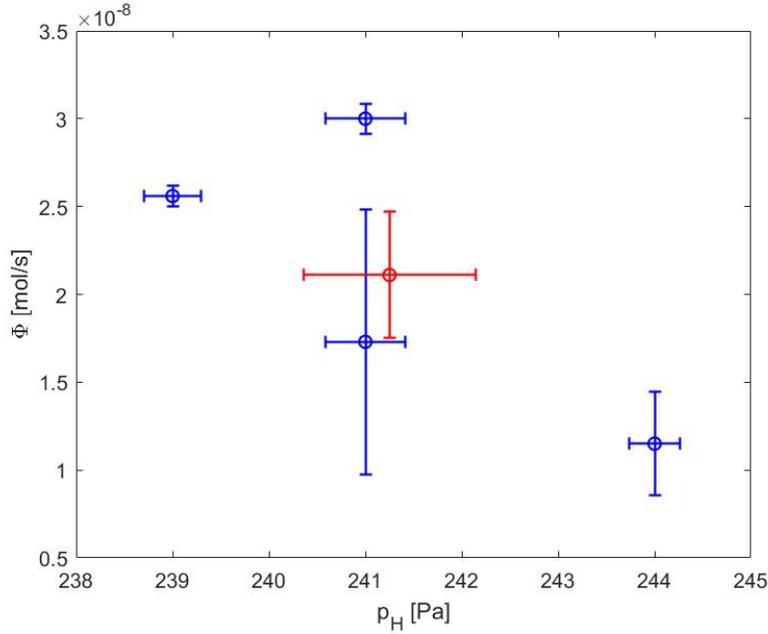
Figure 4.9: Average flux of test 2, the remaining are in appendix A.1

The plots below show the average global permeated flux for each test. As anticipated, the flux is notably affected by the temperature (figure 4.10b), with the first experimental campaign conducted at higher temperatures exhibiting higher values with respect to the second one.
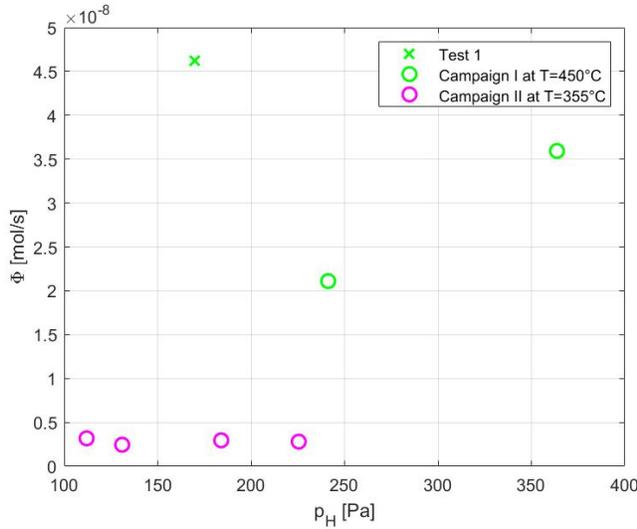


(a) Experimental flux trend with partial pressure
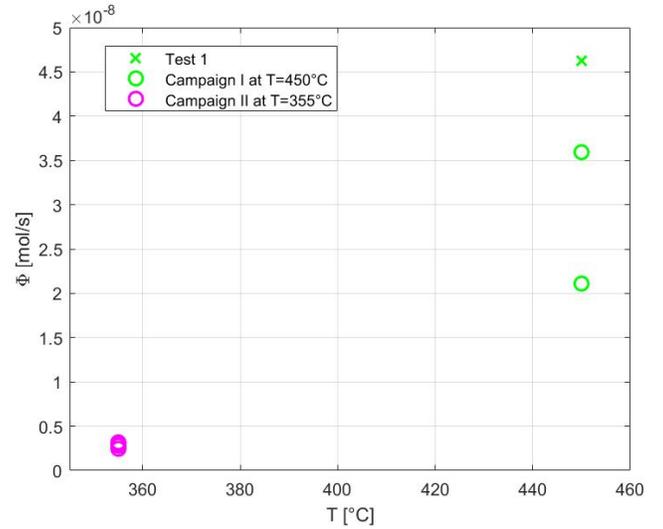


(b) Experimental flux trend with temperature

Test 1 has been excluded from the validation process detailed in section (4.3.3) since its value is out of scale. This deviation resulted in an unphysical trend for the permeated flux, as illustrated in figure (4.10a). In fact, the permeated flux is expected to increase together with the hydrogen partial pressure. Assuming a constant sink term for all experiments ($p_{vacuum} \approx 0$ constant, as stated in [20]), the vacuum concentration remains constant. Consequently, with an increase in the inlet partial pressure and, by extension, an increase in the inlet concentration governed by Sievert's law ($C \propto p_H^{0.5}$), the concentration gradient rises. This gradient serves as the driving force for the

permeation flux, implying an increase in the flux. This relationship is evident in figure (4.11), where a clear correlation exists between the inlet concentration $C_0$ and the permeated flux $\Phi$ (computed in this case using OpenFOAM).
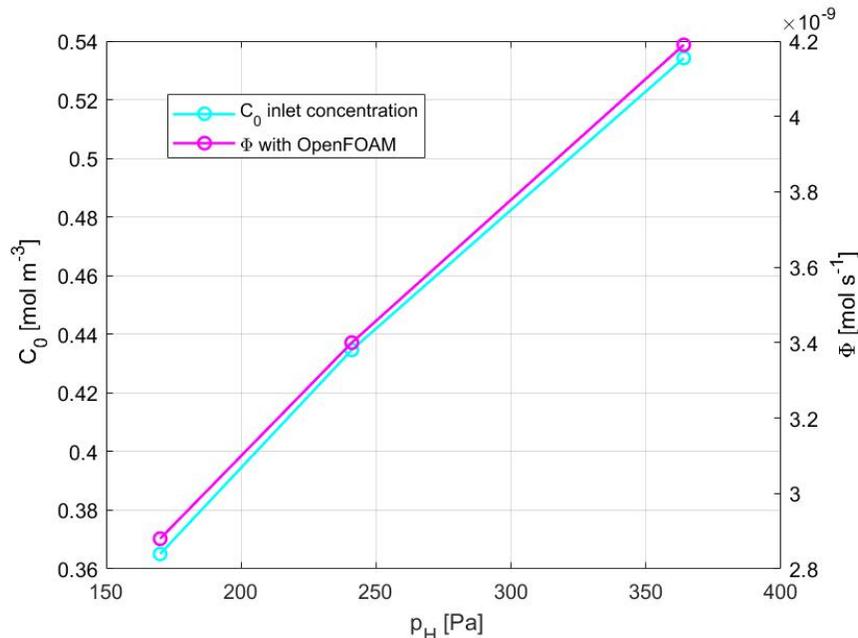


Figure 4.11: Relationship between inlet concentration and permeated flux

### 4.3.3   Comparison with experimental results

This section delves into the validation process, which involves a comparison between the numerical model's output and real-world measurements obtained through experiments. Specifically, we compare the simulation results from OpenFOAM with the experimental data derived from the TRIEX-II PAV mock-up at the Brasimone research center, as detailed in Section (4.3.2). The parameter over which the comparison is conducted is the global permeated flux $\Phi$ [mol/s].
The computational domain is adjusted to resemble as closely as possible the geometry of the PAV (figure 4.12)), comprising of 16 U-pipes in two passages (I and II). Each passage consists of 8 parallel pipes: 3 short (S), 3 medium (M), and 2 long (L). While the various lengths are taken into account, for the sake of simplicity, the U-bend is disregarded and the pipes are treated as straight.
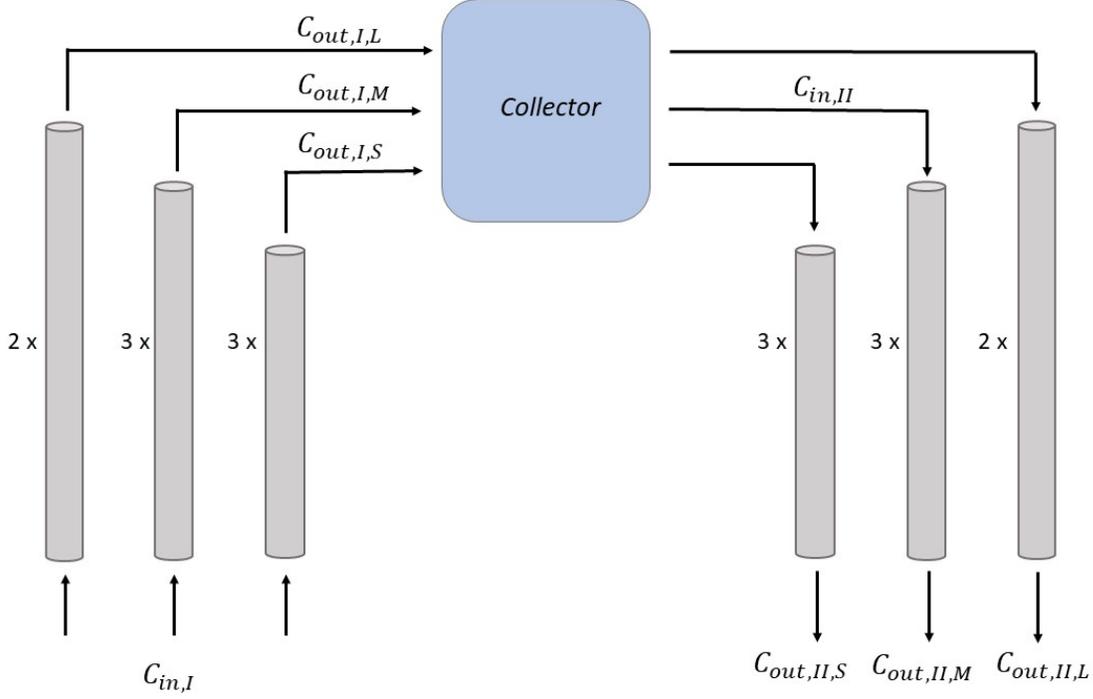
Figure 4.12: Representation of the computational domain adaptation to simulate the PAV's piping system

Additionally, the experimental conditions were replicated, covering temperature, partial pressure, mass flow rate and PbLi solubility. The latter is a critical parameter due to its high uncertainty, and it is derived from the experiments as:

$$K_{s,PbLi} = \frac{\Phi \, \rho}{\eta \, \sqrt{p_H} \, \dot{m}_{PbLi}} \tag{4.9}$$

However, the lithium-lead diffusivity $D_{PbLi}$ was not directly determined in the experiments. Therefore, it was calculated using values from [21], as they yielded the best agreement.
To replicate each experimental test, the following steps are necessary:

1. Set up 3 simulations with computational domain of length: $L_S$, $L_M$ and $L_L$ respectively.

2. Compute the inlet concentration for the first passage (I), which is equal for all the 3 pipes, as:

$$C_{in,I} = K_{s,PbLi}\sqrt{p_H} \tag{4.10}$$

3. From $\dot{m}$ compute with an iterative process the inlet velocity for each pipe: $U_S$, $U_M$, $U_L$, taking into account the different distributed pressure losses due to the different lengths (see appendix A.3).

4. Set the properties according to the specified temperature.

5. Run the 3 simulations until convergence.

31

6. Extract the outlet concentration $C_{out,I}$ from each pipe (computed as mass flow average) and compute the inlet concentration for the second passage (II), as:

$$C_{in,II} = \frac{\sum_{i=1}^{3} C_{out,I,i}}{3} \quad \text{with} \quad C_{out,I,i} = \frac{\int_{A_{out}} \rho U_i C_i \, dA_{out}}{\int_{A_{out}} \rho U_i \, dA_{out}} \tag{4.11}$$

where $i = S, M, L$. The inlet concentration for the second passage is determined by taking the arithmetic average of the outlet concentrations from the first passage. This approach is employed to simulate the presence of a collector, which provides mixing in the system.

7. Run the 3 simulations for the second passage, with $C_{in,II}$ and the respective inlet velocity, until convergence.

An example of parameters used to reproduce one test is in table (4.6).

| Simulation | Passage | Length | $C_{in}[mol/m^3]$ | $U[m/s]$ | other parameters |
|:---:|:---:|:---:|:---:|:---:|:---:|
| 1 | I | $L_S$ | 1.07e-2 | 2.37e-1 | |
| 2 | I | $L_M$ | 1.07e-2 | 2.31e-1 | $p, T, \dot{m}, K_{s,PbLi} \rightarrow experiments$ |
| 3 | I | $L_L$ | 1.07e-2 | 2.24e-1 | |
| 4 | II | $L_S$ | 1.03e-2 | 2.37e-1 | |
| 5 | II | $L_M$ | 1.03e-2 | 2.31e-1 | $\rho, \mu, D, K_{s,Nb} \rightarrow literature$ |
| 6 | II | $L_L$ | 1.03e-2 | 2.24e-1 | |

Table 4.6: Simulations performed for test 1.1

Once the 6 simulations are done, the permeated flux is extracted from each one to computes the global PAV permeated flux, as:

$$\Phi_{PAV,OF} = [3(\Phi_{S,I} + \Phi_{S,II}) + 3(\Phi_{M,I} + \Phi_{M,II}) + 2(\Phi_{L,I} + \Phi_{L,II})]\frac{360}{10} \quad [mol/s] \tag{4.12}$$

Taking advantage of the axialsymmetry of the problem, the compensation for the computational domain's limitation to a 10° wedge is achieved by multiplying for a factor of 360/10.
The next step involves assigning an error bar to the recently computed value. The numerical error bar is determined as a combination of uncertainties of 6 input parameters: $T, p, \dot{m}, \rho_{PbLi}, D_{PbLi}, K_{s,PbLi}$. The errors for $p, T, \dot{m}, K_{s,PbLi}$ are provided from the experiments, while the errors for $\rho_{PbLi}, D_{PbLi}$ are found in [22] and [17] respectively. Mean values, uncertainties, and the resulting error bar for the experimental test 2.1 are provided as an example in table (4.7).
For additional tests, refer to appendix A.4.

| Input parameter | Mean value | Error | min/max | Effect on $\Phi$ | $\Phi_{PAV,OF,min}$ | $\Phi_{PAV,OF,max}$ |
|---|---|---|---|---|---|---|
| p [Pa] | 241 | 0.34% | 240.2 241.8 | ↑ | $p_{min}$ | $p_{max}$ |
| T [°C] | 452 | 0.50% | 449.7 454.3 | ↑ | $T_{min}$ | $T_{max}$ |
| $\dot{m}$ [kg/s] | 1.19 | 8.4% | 1.09 1.29 | ↑ | $\dot{m}_{min}$ | $\dot{m}_{max}$ |
| $K_{s,PbLi}$ [mol/m$^3$/Pa$^{0.5}$] | $1.46 \times 10^{-4}$ | 19% | $1.18 \times 10^{-4}$ $1.73 \times 10^{-4}$ | ↑ | $K_{s,min}$ | $K_{s,max}$ |
| $D_{PbLi}$ [m$^2$/s] | $4.91 \times 10^{-9}$ | 10% | $4.35 \times 10^{-9}$ $5.43 \times 10^{-9}$ | ↑ | $D_{min}(T_{min})$ | $D_{max}(T_{max})$ |
| $\rho_{PbLi}$ [kg/m$^3$] | 9657 | 0.30% | 9625 9689 | ↓ | $\rho_{max}(T_{min})$ | $\rho_{min}(T_{max})$ |

Table 4.7: Error bar for test 2.1

The lower and upper extremes of the error bar ($\Phi_{PAV,OF,min}$ and $\Phi_{PAV,OF,max}$) are obtained by implementing the combination of input parameters that results in the minimum (and maximum) value of the permeated flux, following the steps described above. This ensures that the flux resulting from any other combination of these parameters falls inside the error bar.
The comparison between the numerical permeated flux and the experimental one is shown below, for the first (figure (4.13a)) and second (figure (4.13b)) experimental campaign.



(a) First experimental campaign at $T = 450$°C

(b) Second experimental campaign at $T = 355$°C
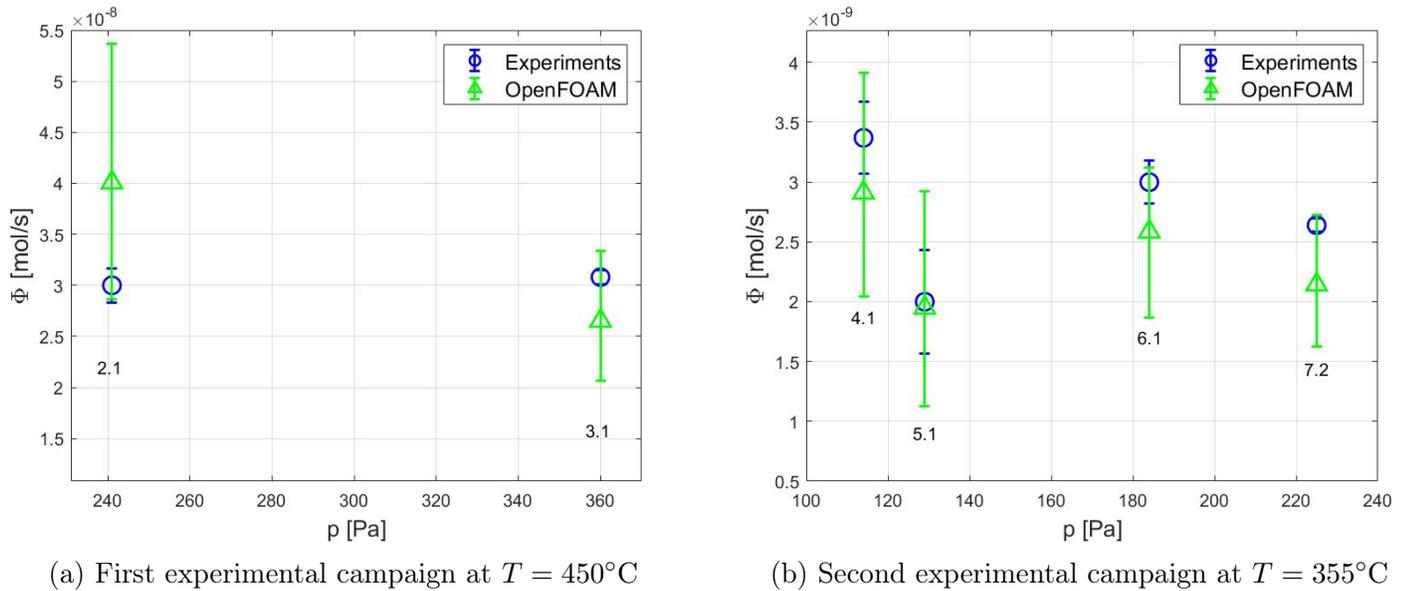
Figure 4.13

The overlapping error bars indicate that the difference between the numerical and experimental flux is not statistically significant. This implies the model effectively predicts experimental values and their overall trend.
Validation is a crucial step for future applications of the model, ensuring that decisions are made with a reliable understanding of the system.

## 4.4 Sensitivity analysis

This chapter is dedicated to the sensitivity analysis study, which investigates how different values of an independent variable affect a particular dependent variable, under a given set of assumptions. In this specific case the goal was to determine how diffusivity, solubility and recombination constant $(D, K_s, k_r)$ effect the global permeated flux $\Phi$. An extensive literature review was conducted on the solubility and diffusivity parameters for both niobium and lithium-lead, and the results are in tables 4.8,4.10, 4.12 and 4.14.

The simulations are carried out for three distinct partial pressure values, namely $p_H = [170, 241, 364]$ Pa, taken from the first experimental campaign (see table 4.4), and at an intermediate temperature between the two campaigns of T= 400°C = 673 K. The computational domain used here is the one discussed in chapter (3.2), therefore considering only the straight portion of the U-pipe.

### 4.4.1 Solubility $K_s$

Regarding the solubility of hydrogen in lithium-lead seven distinct values were identified, within the range of $[7.2 \times 10^{-4}, 8.4 \times 10^{-2}]$ (see table 4.8), while all other parameters are kept constant (see table 4.9).

The results from [23] and [24] were re-elaborated by E. Mas de les Valls et al. (2008)[25]

| $K_{s,PbLi}$ [mol/m$^3$/Pa$^{1/2}$] | range [K] | Reference |
|---|---|---|
| $1.5 \times 10^{-3} \, e^{(-4180/R/T)} = 7.2 \times 10^{-4}$ | $573 < T < 873$ | [21] Edao (2011) |
| $1.4 \times 10^{-3} \, e^{(-1350/R/T)} = 1.1 \times 10^{-3}$ | $508 < T < 700$ | [15] Reiter (1991) |
| $2.6 \times 10^{-2} \, e^{(-9000/R/T)} = 5.3 \times 10^{-3}$ | $550 < T < 900$ | [23] Chan (1984) |
| $5.0 \times 10^{-2} \, e^{(-6100/R/T)} = 1.7 \times 10^{-2}$ | $508 < T < 1040$ | [24] Schumacher (1990) |
| $2.4 \times 10^{-1} \, e^{(-12844/R/T)} = 2.4 \times 10^{-2}$ | $623 < T < 923$ | [26] Aiello (2005) |
| $5.6 \times 10^{-2}$ | $573 < T < 723$ | [27] Katsuta (1985) |
| $8.4 \times 10^{-2}$ | $523 < T < 723$ | [28] Tosti (2013) |

Table 4.8: Literature review for hydrogen solubility in PbLi at T=673 K

| $K_{s,Nb}$ [mol/m$^3$/Pa$^{1/2}$] | $D_{PbLi}$ [m$^2$/s] | $D_{Nb}$ [m$^2$/s] | $k_r$[m$^4$/s/mol] |
|---|---|---|---|
| $4.4 \times 10^1$ | $1.6 \times 10^{-9}$ | $9.1 \times 10^{-9}$ | $6.0 \times 10^{-8}$ |

Table 4.9: Invariant parameters

Regarding the hydrogen solubility in niobium instead only three values were found in literature, therefore the range of study has been extended within $[1 \times 10^1, 1 \times 10^3]$ as shown in the table 4.10. All other parameters are kept constant (see table 4.11).

| $K_{s,Nb}[mol/m^3/Pa^{1/2}]$ | range [K] | Reference |
|---|---|---|
| $1 \times 10^1$ | - | - |
| $1.3 \times 10^{-1}e^{(4240/T)} = 7.0 \times 10^1$ | $T > 400$ | [18] Veleckis (1969) |
| $1 \times 10^2$ | $450 < T < 715$ | [16] Komiya (2005) |
| $1.3 \times 10^{-1}e^{(5550/T)} = 4.9 \times 10^2$ | $T > 400$ | [18] Steward (1975) |
| $1 \times 10^3$ | - | - |

Table 4.10: Literature review for hydrogen solubility in Niobium at T=673K

| $K_{s,PbLi}$ [mol/m³/Pa$^{1/2}$] | $D_{PbLi}$ [m²/s] | $D_{Nb}$ [m²/s] | $k_r$[m⁴/s/mol] |
|---|---|---|---|
| $2.8 \times 10^{-2}$ | $1.6 \times 10^{-9}$ | $9.1 \times 10^{-9}$ | $6.0 \times 10^{-8}$ |

Table 4.11: Invariant parameters

The results are presented in the figures below. In figure (4.14a), a linear relationship is observed between the permeated flux and the solubility of the liquid metal. As expected, the flux increases with both hydrogen partial pressure and solubility. Notably, the impact of hydrogen partial pressure on the flux is more pronounced at higher solubility values. On the other hand, figure (4.14b) illustrates that there is minimal change in the permeated flux attributed to niobium solubility, despite the broad range covered in the study.



(a) PbLi          (b) Niobium
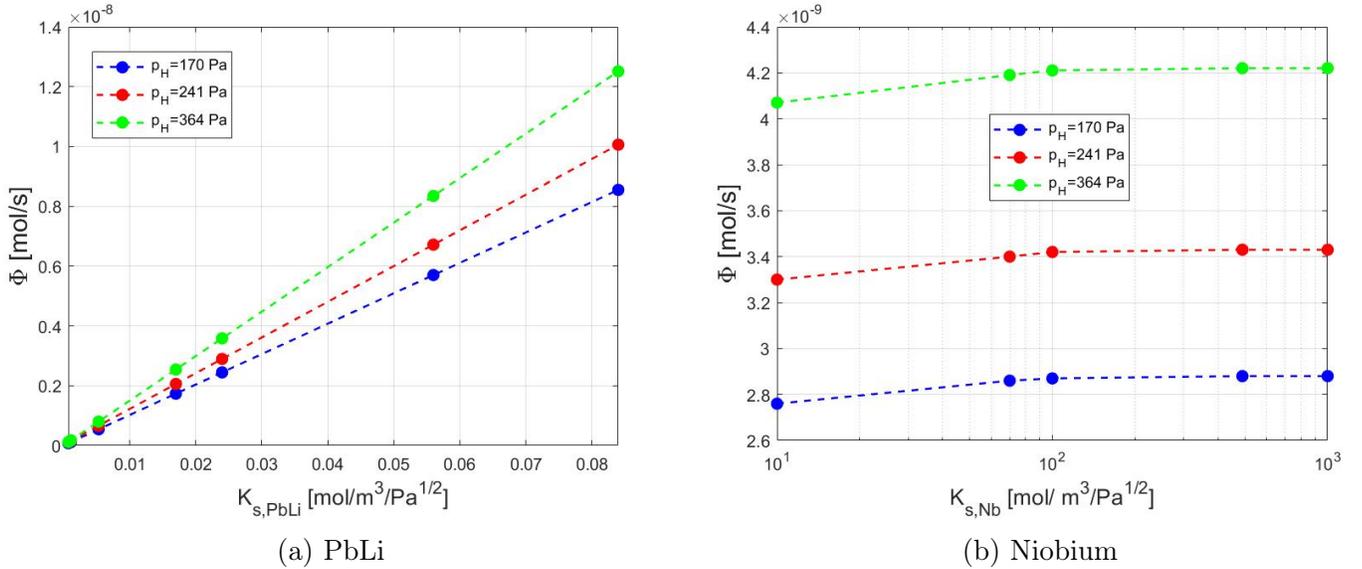
Figure 4.14: Effect of the solubility on the permeated flux

For a more comprehensive understanding of the impact of lithium-lead solubility, the results are also presented on a logarithmic scale (figure 4.15). As expected, the liquid metal solubility exerts a significant influence on the model, resulting in a flux variation spanning two orders of magnitude.
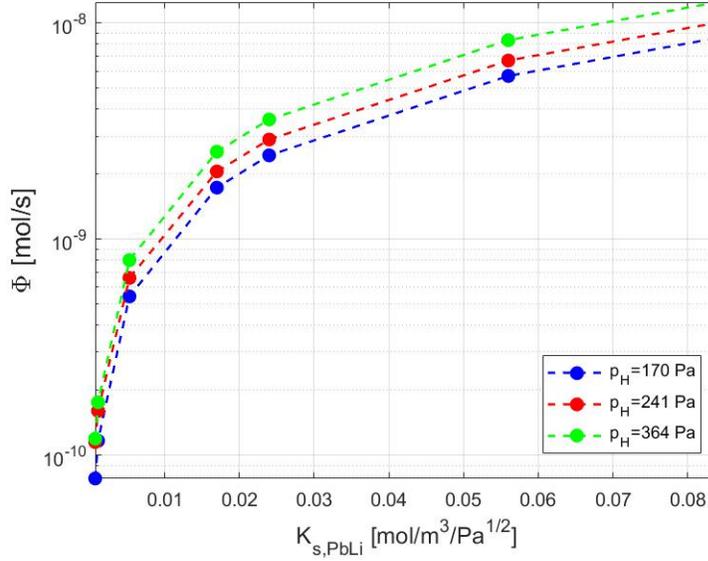
Figure 4.15: Effect of the solubility of hydrogen in lithium-lead on the permeated flux (logarithmic scale)

## 4.4.2 Diffusivity $D$

Concerning hydrogen diffusivity in lithium-lead, a literature review identified three distinct values. The study range has been expanded in the lower limit, yielding a range of $[5 \times 10^{-10}; 4.9 \times 10^{-9}]$ (see table 4.12), while keeping all other parameters constant (see table 4.17).

| $D_{PbLi}[m^2/s]$ | range [K] | Reference |
|---|---|---|
| $5.0 \times 10^{-10}$ | - | - |
| $4.0 \times 10^{-8} \, e^{(-19500/R/T)} = 1.2 \times 10^{-9}$ | $508 < T < 700$ | [15] Reiter (1991) |
| $7.8 \times 10^{-10} \, \sqrt{3} = 1.4 \times 10^{-9}$ | $T = 673$ | [29] Shibuya (1987) |
| $2.5 \times 10^{-7} \, e^{(-27000/R/T)} \, \sqrt{3} = 3.5 \times 10^{-9}$ | $573 < T < 973$ | [17] Terai (1992) |
| $8.2 \times 10^{-8} \, e^{(-15800/R/T)} = 4.9 \times 10^{-9}$ | $573 < T < 873$ | [21] Edao (2011) |

Table 4.12: Literature review for hydrogen diffusivity in PbLi at T=673 K.

| $K_{s,PbLi}$ [mol/m$^3$/Pa$^{1/2}$] | $K_{s,Nb}$ [mol/m$^3$/Pa$^{1/2}$] | $D_{Nb}$ [m$^2$/s] | $k_r$[m$^4$/s/mol] |
|---|---|---|---|
| $2.8 \times 10^{-2}$ | $4.4 \times 10^{1}$ | $9.1 \times 10^{-9}$ | $6.0 \times 10^{-8}$ |

Table 4.13: Invariant parameters

Concerning hydrogen diffusivity in niobium, a literature review identified four distinct values. Again, the study range has been expanded in the lower limit, yielding a range of $[1 \times 10^{-10}; 8.2 \times 10^{-9}]$ (see table 4.14), while keeping all other parameters constant (see table 4.15).

| $D_{Nb}[m^2/s]$ | range [K] | Reference |
|---|---|---|
| $1 \times 10^{-10}$ | - | - |
| $8.0 \times 10^{-10}$ | $645 < T < 800$ | [30] Zhang (2010) |
| $6.0 \times 10^{-9}$ | $623 < T < 873$ | [31] Steward (1983) |
| $5.0 \times 10^{-8} \, e^{(-1230/T)} = 8.0 \times 10^{-9}$ | $223 < T < 873$ | [18] Volkl (1975) |
| $5.4 \times 10^{-8} \, e^{(-1265/T)} = 8.2 \times 10^{-9}$ | $235 < T < 830$ | [32] Cantelli (1969) |

Table 4.14: Literature review for hydrogen diffusivity in Niobium at T=673K

| $K_{s,PbLi}$ [mol/m$^3$/Pa$^{1/2}$] | $K_{s,Nb}$ [mol/m$^3$/Pa$^{1/2}$] | $D_{PbLi}$ [m$^2$/s] | $k_r$[m$^4$/s/mol] |
|---|---|---|---|
| $2.8 \times 10^{-2}$ | $4.4 \times 10^1$ | $1.6 \times 10^{-9}$ | $6.0 \times 10^{-8}$ |

Table 4.15: Invariant parameters

The results are presented in the figures below. Figure (4.16a) shows that the permeated flux exhibits a significant dependency on the diffusivity of PbLi. This shows that, under equivalent flow conditions, diffusion-driven transport within the liquid metal remains an important transport mechanism. In figure (4.16b) instead all niobium diffusivity values yield almost identical permeated flux, indicating that the model is not sensitive to this parameter.



(a) PbLi
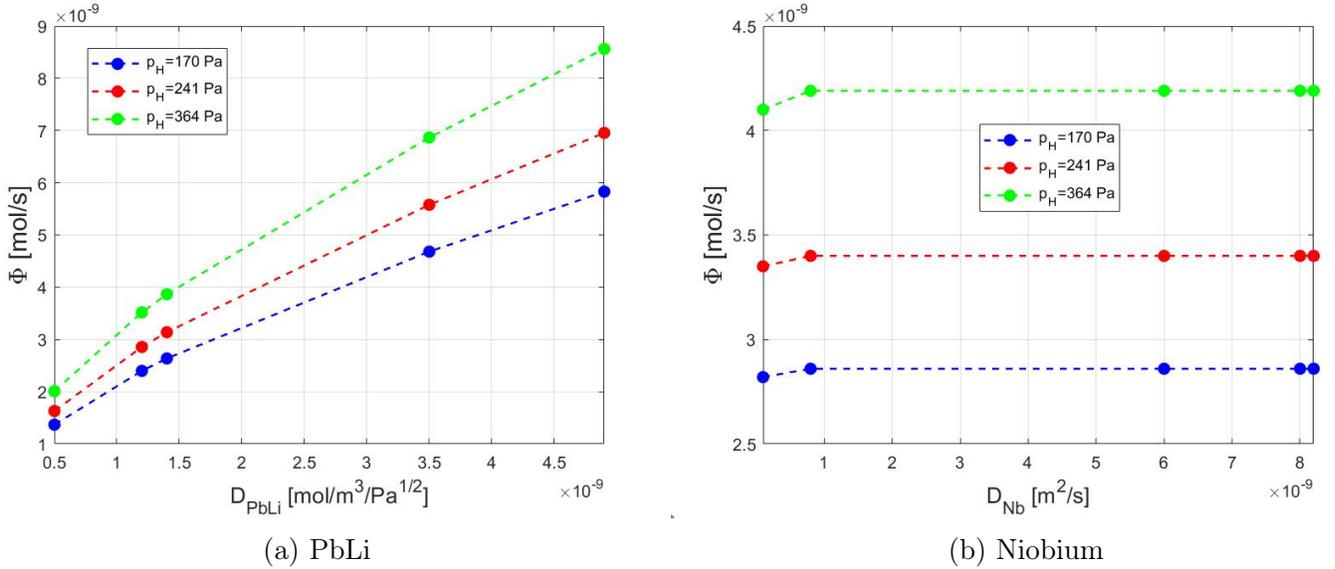
(b) Niobium

Figure 4.16: Effect of the diffusivity on the permeated flux

### 4.4.3 Recombination constant $k_r$

Limited literature exists on the recombination constant for niobium. The value of $k_r = 1.5 \times 10^{-8}$ m$^4$/mol/s is taken from the experimental results reported in [33] and [34]. The range of study is extended both in the upper and lower limit obtaining $[1.0 \times 10^{-10}, 1.0 \times 10^{-6}]$, while all other parameters are kept constant (see table 4.16).

| $K_{s,PbLi}$ [mol/m$^3$/Pa$^{1/2}$] | $K_{s,Nb}$ [mol/m$^3$/Pa$^{1/2}$] | $D_{PbLi}$ [m$^2$/s] | $D_{Nb}$ [m$^2$/s] |
|---|---|---|---|
| $2.8 \times 10^{-2}$ | $4.4 \times 10^{1}$ | $1.6 \times 10^{-9}$ | $9.1 \times 10^{-9}$ |

Table 4.16: Invariant parameters
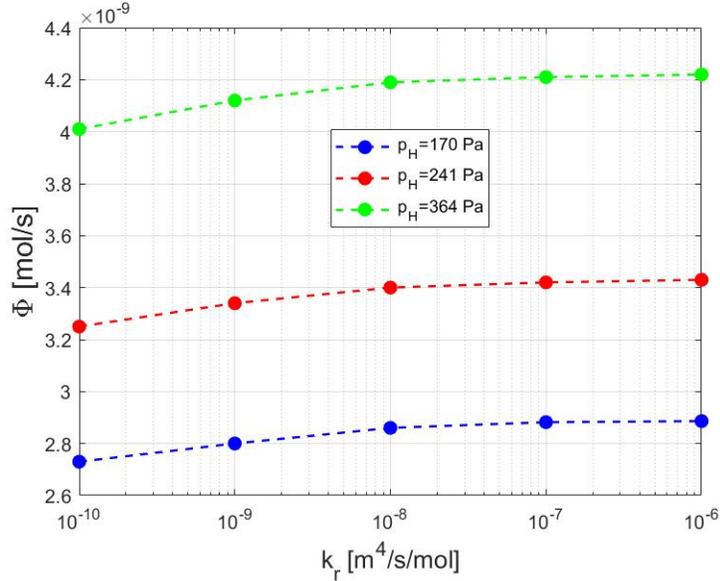


Figure 4.17: Effect of the recombination constant on the permeated flux

As we can see from figure (4.17) the recombination constant does not play a significant role, as the permeate flux is only minimally influenced by it, despite the wide range of study.

It's important to note that while the evaluation of all properties is done within the temperature validity range, the same cannot be asserted for partial pressure. Notably, existing literature often specify (if any) pressure validity ranges with lower limits around $p_H = 1000$ Pa, while in this study the experimental pressure range is $p_H = [114, 364]$ Pa. However, this discrepancy is not anticipated to disprove the overall conclusion of the study.

The main outcome of the sensitivity analysis is that the model has a strong dependency only on the liquid metal properties, while the membrane does not effect the permeated flux in a significant way.

## 4.5 Permeation regime

The permeation regime identifies the limiting transport mechanism for the hydrogen isotopes, where the term "limiting" denotes the slowest process which governs the system.

There are three distinct permeation regimes:

1. Diffusion Limited Regime (DLR): when the permeation process is limited by the atomic diffusion of hydrogen in the niobium membrane.

2. Surface Limited Regime (DLR): when the permeation process is limited by surface effects, such as dissociation and recombination, occurring on the membrane's surface.

3. Liquid Limited Regime (LLR): when the permeation process is limited by the transport of atomic hydrogen within the liquid.

The permeation regime can be identified through two adimensional parameters (figure (4.18a)):

$$W = \frac{k_r \, K_{s,Nb} \, t \, \sqrt{p_H}}{D_{Nb}} \qquad \text{permeation parameter}$$

$$\xi = \frac{D_{Nb} \, K_{s,Nb}}{h_T \, K_{s,PbLi} \, t} \qquad \text{partition parameter}$$

In the case of *liquid-solid-gas* systems (like the PAV) there are two possible scenarios:
For $\xi >> 1$ and $W > 1$ the regime is LLR, where the mass transport in the liquid is the rate-limiting phenomenon and the flux is independent of the membrane properties.
For $\xi >> 1$ and $W << 1$ the regime lies between the LLR and SLR, called mixed-regime.
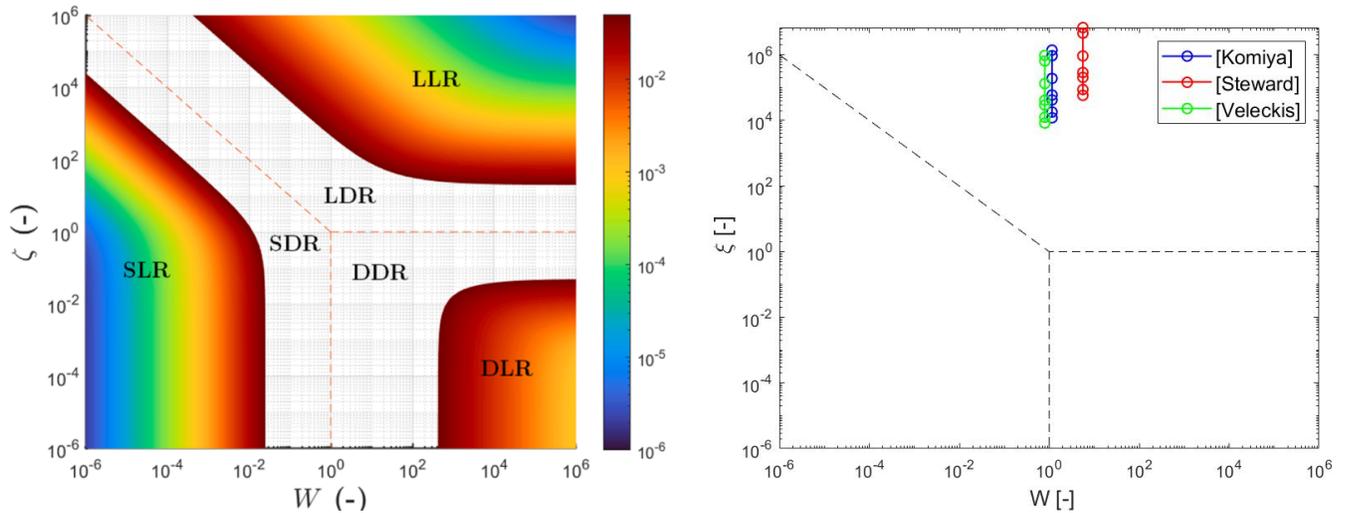


(a) Permeation regimes as functions of $\xi$ and $W$ [35]    (b) $\xi$ and $W$ computed with properties from (4.4)

Figure 4.18: Permeation regime

In figure (4.18b) are shown the values of $\xi$ and $W$ computed using solubility and diffusivity discussed in the previous chapter. It is clear that the PAV is operating in Liquid Limited Regime. This results is coherent with the conclusions drawn from the sensitivity analysis study, indicating a strong dependency on the liquid metal properties. This implies that the transport of hydrogen in PbLi is the slowest phenomena, and therefore is the one driving the permeation efficiency of the PAV.

## 4.6   Mass transfer coefficient $h_T$

The mass transfer coefficient quantify the mass transfer from the PbLi bulk to the fluid region near the membrane wall. It is defined as:

$$\Phi = \bar{h}_T \, A_s (C_{bulk} - C_{w,l}) \tag{4.13}$$

Where $\Phi$ [mol/s] is the total permeated flux, $A_s$ [m$^2$] is the surface area of the liquid/solid interface, $C_{bulk}$[mol/m$^3$] is the liquid bulk concentration and $C_{w,l}$[mol/m$^3$] is the liquid concentration at the liquid/solid interface.

The average and local mass transfer coefficients are related by:

$$\bar{h}_T = \frac{1}{A_s} \int_{A_s} h_T \, dA_s \tag{4.14}$$

The mass transfer coefficient is computed from the simulations as in eq. (4.13). The concentrations $C_{bulk}$ and $C_{w1,s}$ are computed as mass flow average and as surface average on the liquid side, respectively:

$$C_{bulk} = \frac{\int_A \rho \, U \, C \, dA}{\int_A \rho \, U \, dA} \qquad \text{and} \qquad C_{w,l} = \frac{1}{A_s} \int_{A_s} C \, dA_s \tag{4.15}$$

Where $A[m^2]$ is the flow area.

The implementation on OpenFOAM is illustrated in the following table.

|  |  | $C_{bulk}$ |  | $C_{w,l}$ |
|---|---|---|---|---|
| $h_T$ | type | *surfaceFieldValue* | type | *surfaceFieldValue* |
|  | operation | *areaAverage* | operation | *patchAverage* |
|  | regionType | *sampledSurface* | patch | *liquidWall* |
|  | weightField | *rhoU* | weightField | - |
| $\bar{h}_T$ | type | *volFieldValue* | type | *surfaceFieldValue* |
|  | operation | *volAverage* | operation | *patchAverage* |
|  | region | *liquidMetal* | patch | *liquidWall* |
|  | weightField | *rhoU* | weightField | - |

Table 4.17: OpenFOAM implementation for the bulk and surface concentrations

The local coefficient is evaluated at seven distinct axial positions: $z = [0\ 2.5\ 5.0\ 8.0\ 25.0\ 50.0\ 83.4]$cm and the profile is reported in figure (4.19). A notable reduction is observed in the entry region, where the mass transfer coefficient undergoes a reduction of up to 87%. As a result, 63% of the hydrogen flux (considering only the straight portion of the U-tube) permeates within the initial 30 cm of the pipe.

$$\frac{\Phi(0 < z < z_{FD})}{\Phi(0 < z < 0.834m)} \cdot 100 = 63\% \tag{4.16}$$
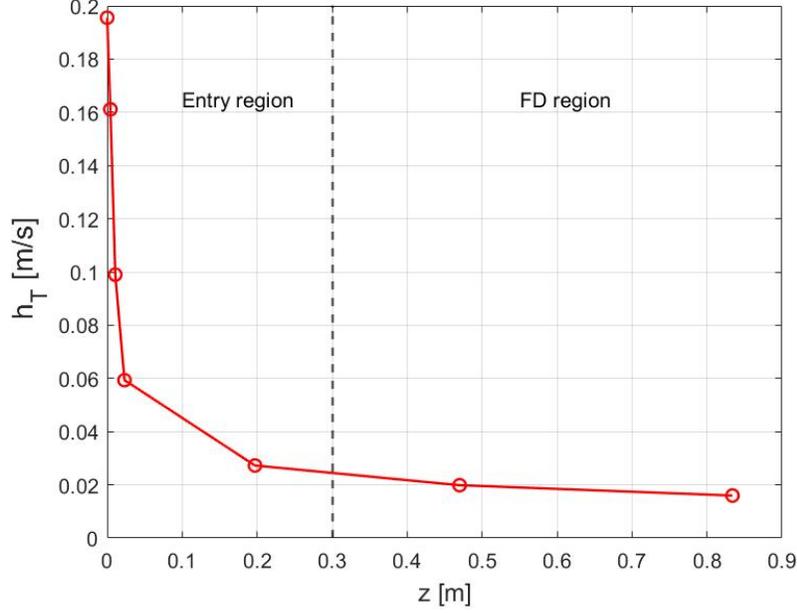
Figure 4.19: Axial variation of the mass transfer coefficient

The mass transfer coefficient is mainly driven by the ratio $h_T \sim \frac{\Phi}{C_{bulk}}$, given that the surface concentration is orders of magnitude smaller than the bulk one (figure (4.20a)). As show in figure (4.20b) the ratio $\frac{\Phi}{C_{bulk}}$ is higher in the entry region, then the flux rapidly converges toward an asymptotic value, while the bulk concentration continues to decrease.
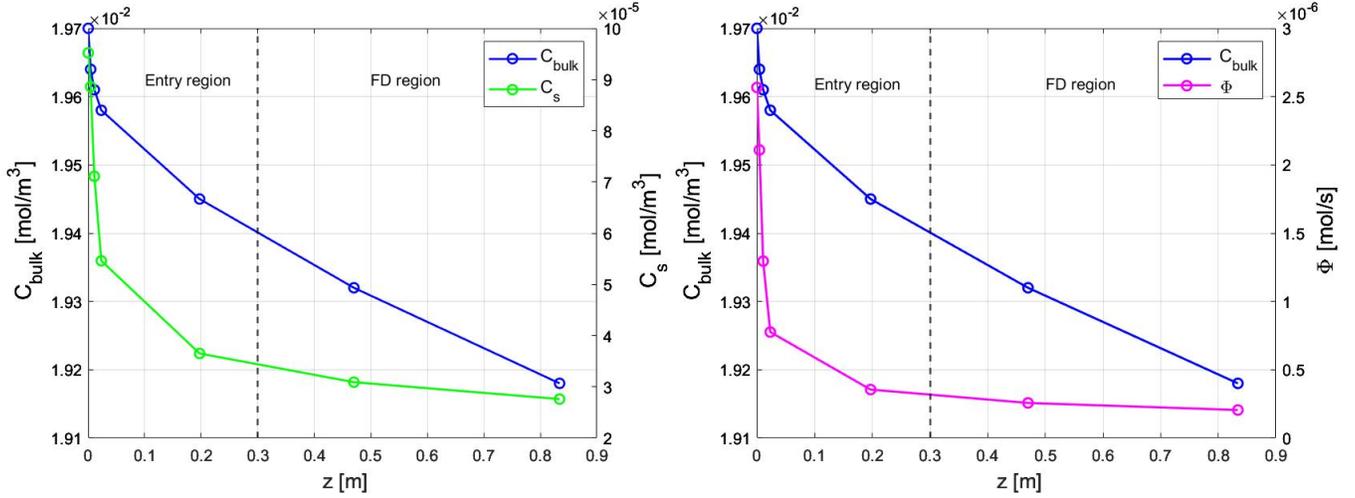


(a) Axial variation of bulk and surface concentration   (b) Axial variation of bulk concentration and flux

Figure 4.20

Subsequently, the average mass transfer coefficient was compared with the one derived from analytical correlations, which are developed for mass transfer in water:

$$\bar{h}_T = \frac{D_{PbLi}}{d} \, Sh = \frac{D_{PbLi}}{d} \, 0.0096 \, Re^{0.913} \, Sc^{0.346} \qquad \text{[36] Harriot (1965)} \qquad (4.17)$$

$$\bar{h}_T = \frac{D_{PbLi}}{d} \, Sh = \frac{D_{PbLi}}{d} \, 0.023 \, Re^{0.8} \, Sc^{0.4} \qquad \text{[37] Incropera (2011)} \qquad (4.18)$$

41

The Sherwood adimensional number (Sh) represents the ratio of convective to diffusive mass transfer and it is the mass transfer equivalent of the Nusselt number. It can be expressed as a combination of the Reynolds and Prandtl number, as $Sh = \alpha\, Re^{\beta}\, Pr^{\gamma}$. The Schmidt adimensional number (Sc) represents the ratio of momentum to mass transfer, as $Sc = \nu/D$.

As shown in figure (4.21) the mass transfer coefficient obtained through OpenFOAM exhibits significantly lower values compared to the analytical counterpart. The discrepancy is to be expected, given that they are referring to two very different liquids.
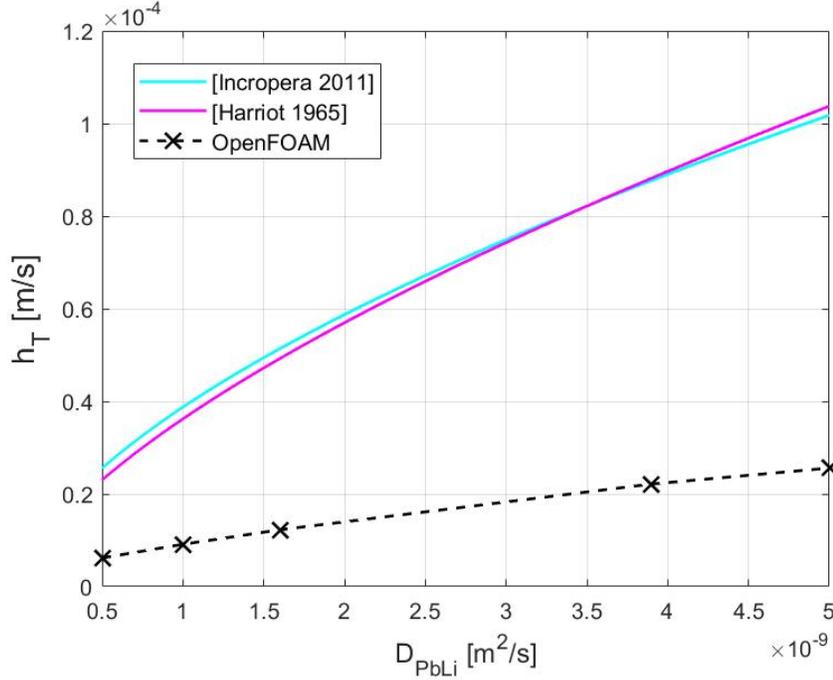


Figure 4.21: $\bar{h}_T$ computed from numerical simulations (for liquid metal) and analytical correlations (for water)

The substantial deviation from existing correlations designed for water is the reason behind the ongoing effort to formulate a new correlation for the mass transfer coefficient for liquid metals (refer to Chapter (5)).

Furthermore, the average mass transfer coefficient was compared with the one obtained from the GETTHEM PAV model [38] [39]. GETTHEM is a dynamic system-level model implemented using the Modelica language, where $\bar{h}_T$ is obtained through a calibration process of the TRIEX-II PAV mock-up experimental data. The same properties were used for this comparison and, together with the results, they are listed in table (4.18).

| Campaign | Test | $K_{s,PbLi}$ [mol/m$^3$/Pa$^{1/2}$] | $\bar{h}_T$ [m/s] OpenFOAM | $\bar{h}_T$ [m/s] GETTHEM |
|---|---|---|---|---|
| $T_1 = 450°C$ | 2.1 | $1.46 \times 10^{-4}$ | $1.91 \times 10^{-5}$ | $1.54 \times 10^{-5}$ |
| | 3.1 | $7.87 \times 10^{-5}$ | $1.95 \times 10^{-5}$ | $2.53 \times 10^{-5}$ |
| $T_2 = 355°C$ | 4.1 | $2.10 \times 10^{-5}$ | $1.24 \times 10^{-5}$ | $1.84$ |
| | 5.1 | $1.22 \times 10^{-5}$ | $1.20 \times 10^{-5}$ | $1.77 \times 10^{-5}$ |
| | 6.1 | $1.36 \times 10^{-5}$ | $1.20 \times 10^{-5}$ | $2.00 \times 10^{-5}$ |
| | 7.2 | $1.02 \times 10^{-5}$ | $1.19 \times 10^{-5}$ | $2.15 \times 10^{-5}$ |

Table 4.18: Properties used for the OpenFOAM/GETTHEM comparison.
$K_{s,PbLi}$ computed from experiments, $D_{PbLi}$ from Terai [17], $K_{s,Nb}$ and $D_{Nb}$ both from Steward [18]

This comparison is of particular interest due to the different approaches employed by the two models. OpenFOAM, being a CFD software, focus on a detailed description of geometry and liquid dynamics, whereas system-level code concentrate on capturing overall system behavior, emphasizing efficiency and speed in analyzing the system-level responses.
The results are shown in figure (4.22) which shows the good agreement between the two models and the large overestimation of the mass transfer coefficient by existing correlations.
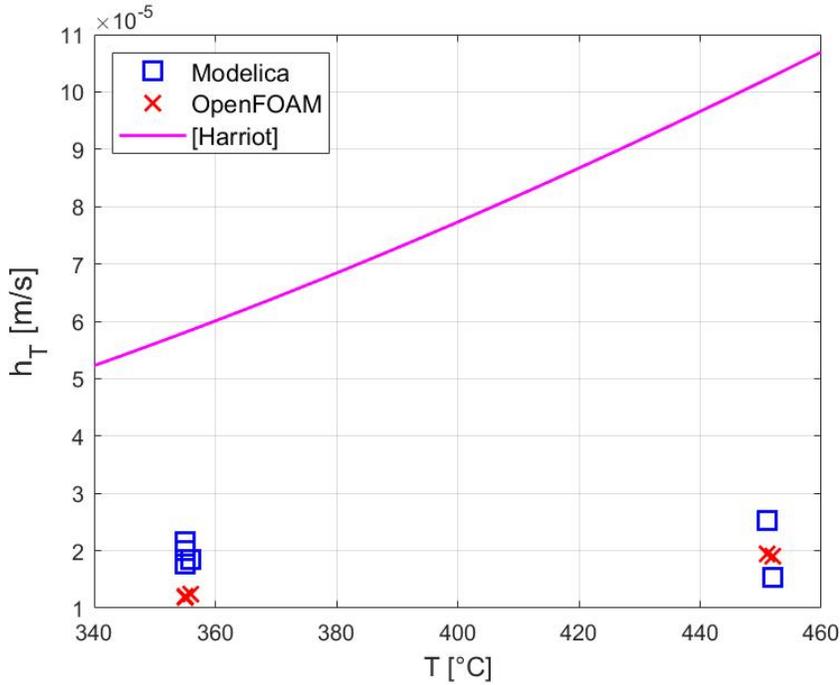


Figure 4.22: $\bar{h}_T$ with OpenFOAM and GETTHEM [38]

Figure (4.23) illustrates the substantial influence of temperature and PbLi diffusivity on $h_T$. While the comparison with GETTHEM code employs the Terai diffusivity, during the validation process the Edao diffusivity showed a better agreement. Additionally, there appears to be no discernible dependence on the hydrogen partial pressure from the OpenOAM's results.
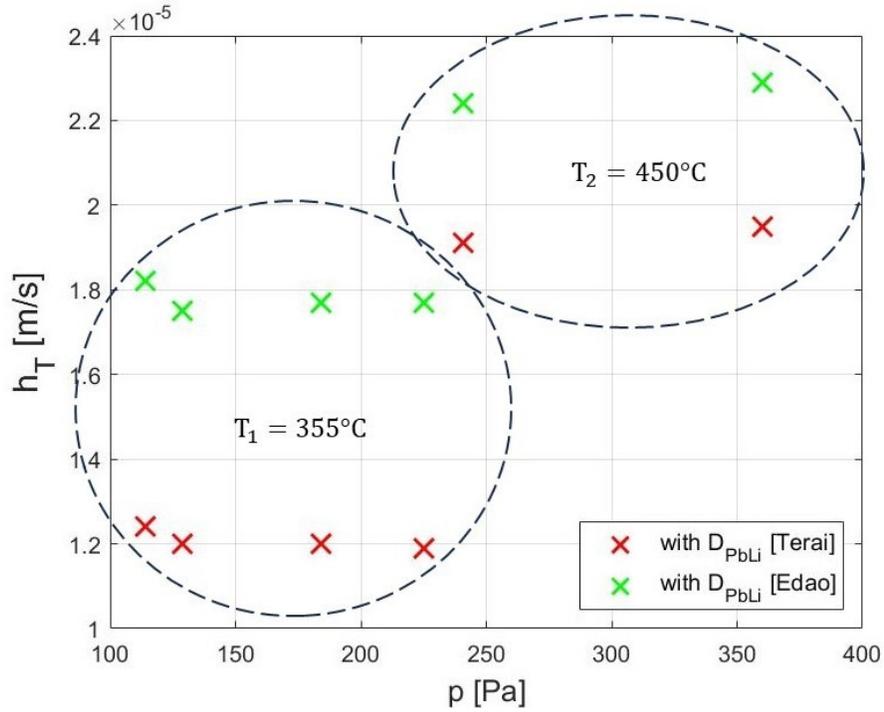
Figure 4.23: $\bar{h}_T$ with OpenFOAM and different $D_{PbLi}$

It is interesting to notice that the mass transfer coefficient is independent from PbLi solubility: this is due to the linear relationship between solubility (and therefore concentration) and the permeated flux, as show in figure (4.14a). The linearity emerges because the recombination phenomenon, responsible for non-linearity since $J_{rec} \propto C^2$, does not play a significant role in this case, as was previously assesed (figure (4.17)).
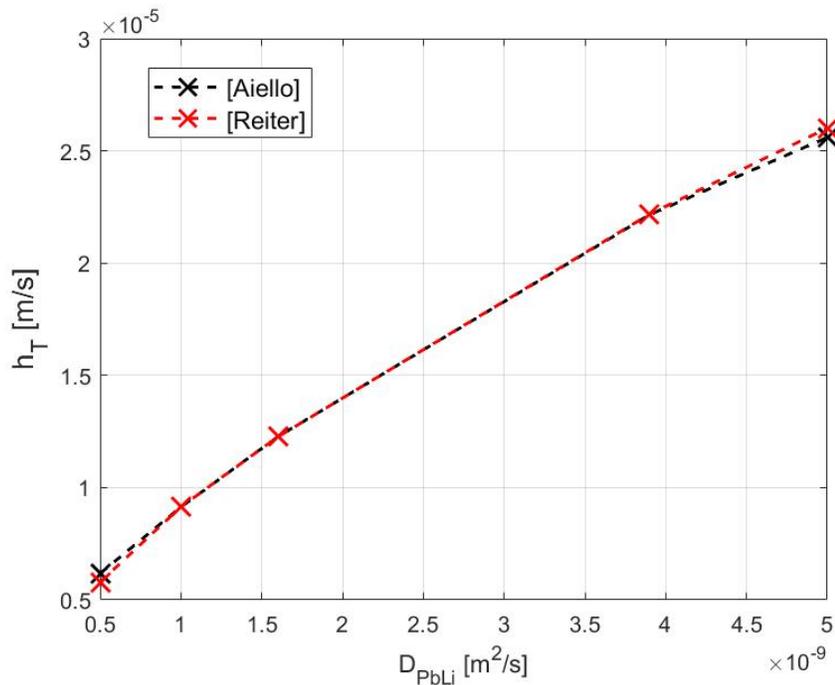


Figure 4.24: $\bar{h}_T$ computed with two different solubilities: Aiello [26] and Reiter [15]

This behaviour is in good agreement with Humrickhouse analytical model [12]:

$$C(z) = C_{in} \ \exp\left(-\frac{4 \, h_T}{U \, d} \frac{\xi}{1+\xi} \, z\right) \tag{4.19}$$

$$J = h_T \, C_{bulk} \frac{\xi}{1+\xi} \tag{4.20}$$

The model state that if $\xi >> 1$, and therefore $\frac{\xi}{1+\xi} \to 1$, the problem simplifies as follows:

$$C(z) \simeq C_{in} \ \exp\left(-\frac{4 \, h_T}{U \, d} \, z\right) \tag{4.21}$$

$$J \simeq h_T \, C_{bulk} \tag{4.22}$$

In the system under study $\xi \approx 1 \times 10^5$. Therefore, the flux per unit area can be approximated as $J \simeq h_T C_{bulk}$, which is supported by the fact that $C_{w1,s} << C_{bulk}$. Hence, $h_T \simeq \frac{J}{C_{bulk}}$ remains constant regardless of solubility values.

According to Humrickhouse, when $\xi >> 1$, the permeation flux is primarily limited by mass transport in PbLi. Therefore, to enhance permeation, a focus on improving $h_T$ is essential rather than on membrane permeability or thickness.

# Chapter 5

# Conclusions and future work

The present work is composed of three primary parts. First, the development of a CFD model using OpenFOAM to simulate tritium transport within the liquid metal and the membrane of the PAV. Secondly, the model is validated against experimental data obtained from the PAV mock-up at the TRIEX-II facility in ENEA Brasimone research center. Lastly, the model is applied to carry out a sensitivity analysis and compute the mass transfer coefficient.

The model consist of a 10° axially symmetric wedge geometry representing the straight segment of the U-pipe. It utilizes the *chtMultiRegionFoam* solver, which solves the conjugate heat transfer between the solid and liquid regions. To address the specifics of the problem, a scalar transport equation for concentration is added to the solver, following the passive scalar approach. At the liquid/solid interface, the boundary condition (BC) is adapted from the *myTurbulentTemperature-CoupledBaffleMixed* BC, modified to account for the concentration jump resulting from the solubility discontinuity in the two materials. On the solid/vacuum interface, the boundary condition is customized using the *codedFixedValue* option by imposing flux continuity.

The model validation is carried out using the global permeated flux as a comparison parameter, replicating the PAV piping system and experimental conditions. Other unspecified values (notably the PbLi diffusivity $D_{PbLi}$) are taken from the literature. The numerical error bar is determined as a combination of uncertainties from the six input parameters. The overlapping of the experimental and numerical error bars indicates that the difference between the numerical and experimental flux is not statistically significant, demonstrating that the model effectively predicts the experimental values. Consequently, the model can be reliably used to gain insights into the PAV behavior.

A sensitivity analysis [39] is first performed on key physical parameters representing the primary source of uncertainty: diffusivity $D$, solubility $K_s$, and recombination constant $k_r$ for both PbLi and niobium. The results reveal a strong dependency of the model only on the properties of the liquid metal, with the membrane having minimal effect on the permeated flux. This confirms that tritium extraction is primarily limited by transport within the liquid metal, rather than diffusion or surface effects in the membrane, concluding that the PAV operates in Liquid Limited Regime (LLR) [35]. Consequently, the main parameter driving PAV permeation efficiency is the mass transfer coefficient of hydrogen in PbLi. Specifically, the latter shows a significative sensibility on the lithium-lead diffusivity and temperature. When the mass transfer coefficient is compared with existing analytical correlations, developed for water, it shows a significant overestimation of the coefficient.

Therefore, future activities foresee the application of the validated model to develop a new correlation for the mass transfer coefficient of hydrogen in the PbLi. This correlation could then be implemented in the system-level GETTHEM code [38], allowing an extrapolation of the mock-up test results to design the full-size EU DEMO PAV.

This thesis work could be further improved by, firstly, modeling the U-bend of the pipe to have a more accurate description of the fluid flow and, secondly, by investigating the impact of the inlet region on permeation and potentially exploring the implementation of turbulence promoters.

# Appendix A

# Appendix

## A.1  Permeated flux from experiments
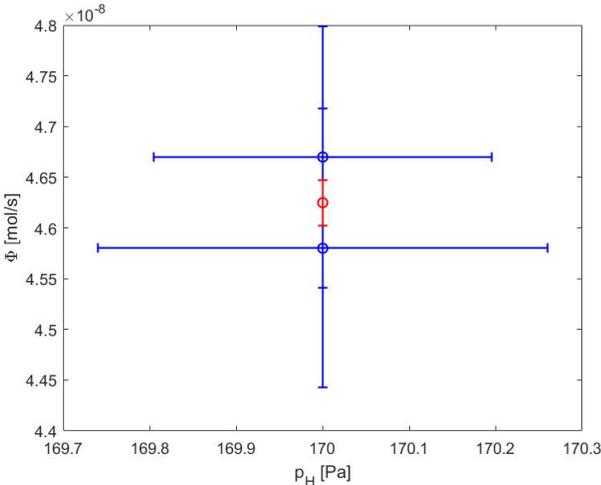


(a) Test 1
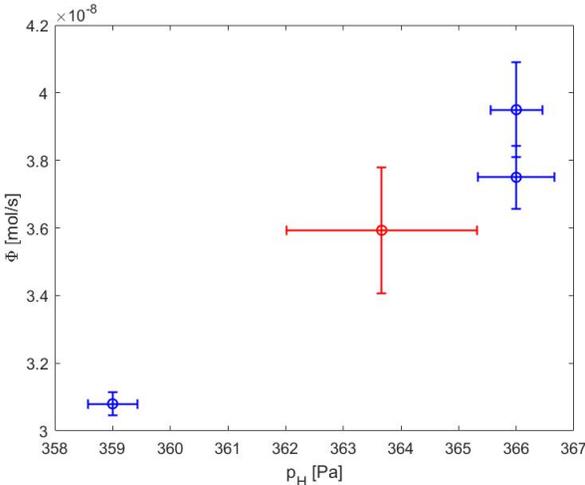
(b) Test 3
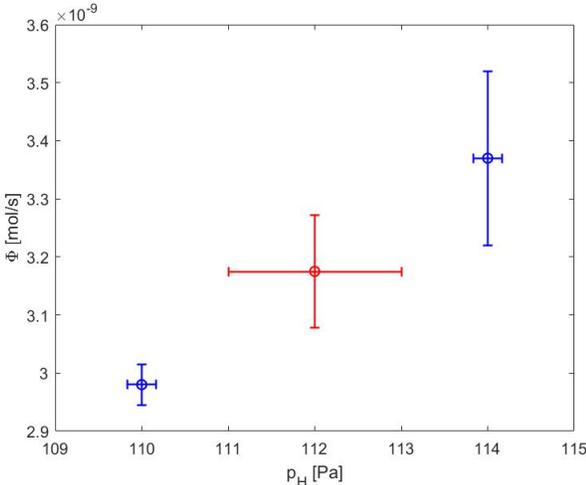


(a) Test 4
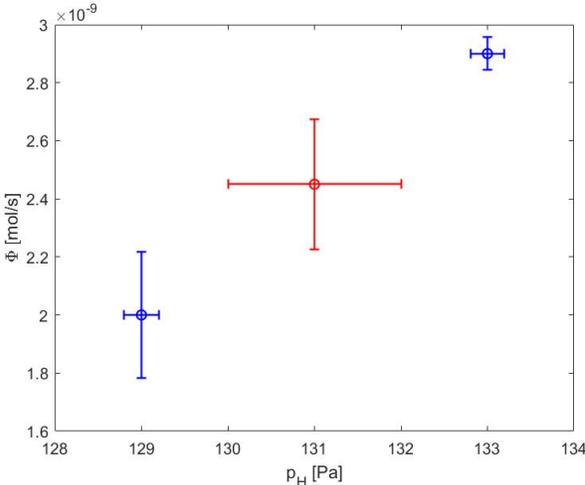
(b) Test 5
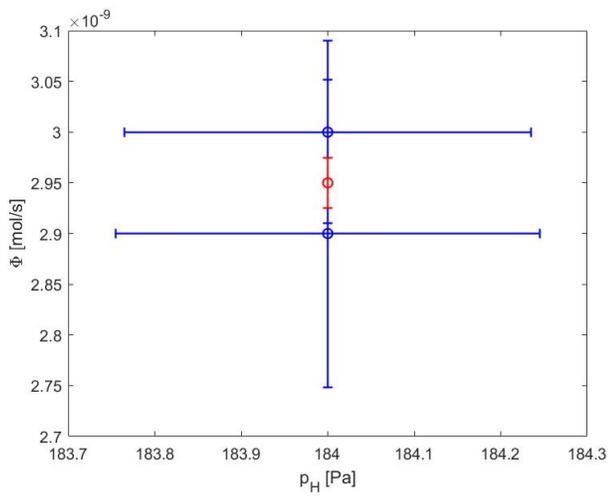
(a) Test 6
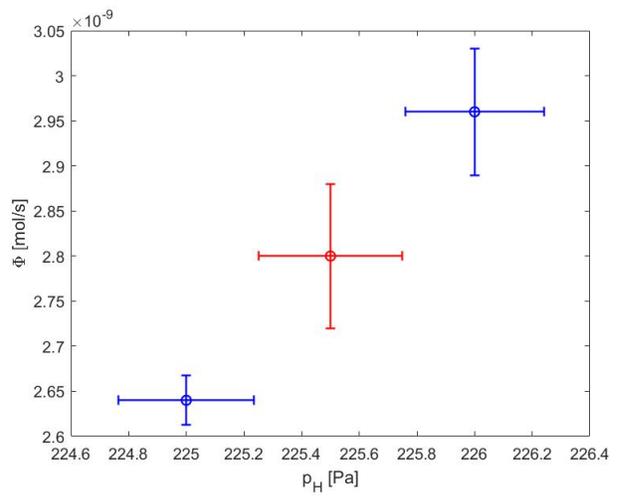


(b) Test 7

## A.2 OpenFOAM code

Implementation in OpenFOAM of different patches on a shared boundary between two regions.
In *system/liquidMetal/blockMeshDict*

```
1  fluidWall
2      {
3          type            mappedWall;
4          sampleMode      nearestPatchFaceAMI;
5          sampleRegion    membrane;
6          samplePatch     innerWall;
7
8        faces
9        (
10           (1 4 5 2)
11           (4 7 8 5)
12       );
13     }
```

In *system/membrane/blockMeshDict*

```
1      innerWall
2      {
3          type            mappedWall;
4          sampleMode      nearestPatchFaceAMI;
5          sampleRegion    liquidMetal;
6          samplePatch     fluidWall;
7
8          faces
9          (
10            (0 3 7 4)
11            (4 7 11 8)
12       );
13     }
```

New transport equation for the concentration $C$.
In *mychtMultiRgionFoam/fluid*

```
1  IOdictionary thermophysicalProperties(
2    IOobject("thermophysicalProperties",runTime.constant(),fluidRegions[i],
3          IOobject::MUST_READ,IOobject::NO_WRITE,
4          false)
5    );
6  dimensionedScalar D (thermophysicalProperties.lookup("D"));
7
8        fvScalarMatrix CEqn
9        (
10           fvm::ddt(rho,C)
11         + fvm::div(phi,C)
12         - fvm::laplacian(rho*D,C)
13       );
14
15       CEqn.relax();
16
```

```
17          fvConstraints.constrain(CEqn);
18
19          CEqn.solve("C");
20
21          fvConstraints.constrain(C);
```

In *mychtMultiRgionFoam/solid*

```
1  IOdictionary thermophysicalProperties(
2    IOobject("thermophysicalProperties",runTime.constant(),solidRegions[i],
3            IOobject::MUST_READ,IOobject::NO_WRITE,
4            false)
5    );
6
7  dimensionedScalar D (thermophysicalProperties.lookup("D"));
8
9          fvScalarMatrix CEqn
10         (
11             fvm::ddt(rho,C)
12           - fvm::laplacian(rho*D,C)
13         );
14
15         CEqn.relax();
16
17         fvConstraints.constrain(CEqn);
18
19         CEqn.solve("C");
20
21         fvConstraints.constrain(C);
```

Implementation of *myRobinBC*, compiled inside the solver. To be concise, only the "Member Functions" portion in reported, as it was the only part modified.

```
1
2  // * * * * * * * * * * * * * * * Member Functions   * * * * * * * * * * * * * * //
3
4  void myTurbulentTemperatureCoupledBaffleMixedFvPatchScalarField::updateCoeffs()
5  {
6      if (updated())
7      {
8          return;
9      }
10
11     // Since we're inside initEvaluate/evaluate there might be processor
12     // comms underway. Change the tag we use.
13     int oldTag = UPstream::msgType();
14     UPstream::msgType() = oldTag+1;
15
16     // Get the coupling information from the mappedPatchBase
17     const mappedPatchBase& mpp = refCast<const mappedPatchBase>(patch().patch())
       ;
18     const polyMesh& nbrMesh = mpp.sampleMesh();
19     const label samplePatchi = mpp.samplePolyPatch().index();
20     const fvPatch& nbrPatch = refCast<const fvMesh>(nbrMesh).boundary()[
       samplePatchi];
```

51

```cpp
21
22      // Calculate the temperature by harmonic averaging
23      // ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
24
25      typedef myTurbulentTemperatureCoupledBaffleMixedFvPatchScalarField thisType;
26
27      const fvPatchScalarField& nbrTp =
28          nbrPatch.lookupPatchField<volScalarField, scalar>(TnbrName_);
29
30      if (!isA<thisType>(nbrTp))
31      {
32          FatalErrorInFunction
33              << "Patch field for " << internalField().name() << " on "
34              << patch().name() << " is of type " << thisType::typeName
35              << endl << "The neighbouring patch field " << TnbrName_ << " on "
36              << nbrPatch.name() << " is required to be the same, but is "
37              << "currently of type " << nbrTp.type() << exit(FatalError);
38      }
39
40      const thisType& nbrField = refCast<const thisType>(nbrTp);
41
42      // Swap to obtain full local values of neighbour internal field
43      tmp<scalarField> nbrIntFld(new scalarField(nbrField.size(), 0.0));
44      tmp<scalarField> nbrKDelta(new scalarField(nbrField.size(), 0.0));
45
46      if (contactRes_ == 0.0)
47      {
48          nbrIntFld.ref() = nbrField.patchInternalField();
49          const dictionary &thermophysicalProperties = db().lookupObject<
    IOdictionary>("thermophysicalProperties");
50          //const dimensionedScalar D = dimensionedScalar("D", dimensionSet
    (0,2,-1,0,0,0,0),thermophysicalProperties);
51          const dimensionedScalar nbrD = dimensionedScalar("D", dimensionSet
    (0,2,-1,0,0,0,0),nbrMesh.thisDb().lookupObject<IOdictionary>("
    thermophysicalProperties"));
52          //nbrKDelta.ref() = nbrField.kappa(nbrField)*nbrPatch.deltaCoeffs();
53          nbrKDelta.ref() = nbrD.value()*nbrPatch.deltaCoeffs();  // Modified by
    Diana 26/06/2023
54      }
55      else
56      {
57          nbrIntFld.ref() = nbrField;
58          nbrKDelta.ref() = contactRes_;
59      }
60
61      mpp.distribute(nbrIntFld.ref());
62      mpp.distribute(nbrKDelta.ref());
63      const dictionary &thermophysicalProperties = db().lookupObject<IOdictionary
    >("thermophysicalProperties");
64      const dimensionedScalar D = dimensionedScalar("D", dimensionSet
    (0,2,-1,0,0,0,0),thermophysicalProperties);
65      tmp<scalarField> myKDelta = D.value()*patch().deltaCoeffs();   // Modified
    by Diana 26/06/2023
66
67 ///////////// added by Diana 22/06/2023 --> Get info on the fluid side  +
    properties
68      tmp<scalarField> inv_Delta = patch().deltaCoeffs(); // 1/delta_l [m^-1]
```

```cpp
69      tmp<scalarField> inv_nbrDelta = nbrPatch.deltaCoeffs();
70      const tmp<scalarField>& intFld = this->patchInternalField(); // C_l (
    patchField)
71      scalar Ks = readScalar(thermophysicalProperties.lookup("Ks"));
72      scalar nbrKs = readScalar(nbrMesh.thisDb().lookupObject<IOdictionary>("
    thermophysicalProperties").lookup("Ks"));
73
74 // Modified by Diana 22/06/2023
75      this->refValue() = nbrIntFld();
76      this->refGrad() = intFld() * inv_Delta * (myKDelta() / (myKDelta() +
    nbrKDelta() * (nbrKs/Ks-1)) - 1);
77      this->valueFraction() = nbrKDelta() / (nbrKDelta() * nbrKs/Ks + myKDelta());
78
79      Info << "The value of D is: " << D.value() << endl;
80      Info << "The value of nbrD is: " << nbrD.value() << endl;
81      Info << "The value of Ks is: " << Ks << endl;
82      Info << "The value of nbrKs is: " << nbrKs << endl;
83
84      mixedFvPatchScalarField::updateCoeffs();
85
86 // Modified by Diana 29/06/2023, it's the concentration flux [mol/s]
87          scalar Q = gSum(D.value()*patch().magSf()*snGrad());
88
89          Info<< patch().boundaryMesh().mesh().name() << ':'
90              << patch().name() << ':'
91              << this->internalField().name() << " <- "
92              << nbrMesh.name() << ':'
93              << nbrPatch.name() << ':'
94              << this->internalField().name() << " :"
95              << " C at the fluid/solid wall [mol/m^3] "
96              << " min:" << gMin(*this)
97              << " max:" << gMax(*this)
98              << " avg:" << gAverage(*this)
99              << endl;
100
101      Info << "The flux Phi_H [mol/s] at the fluid/solid wall is : " << Q << endl
    ;
102
103      // Restore tag
104      UPstream::msgType() = oldTag;
105 }
106
107
108 void myTurbulentTemperatureCoupledBaffleMixedFvPatchScalarField::write
109 (
110      Ostream& os
111 ) const
112 {
113      mixedFvPatchScalarField::write(os);
114      writeEntry(os, "Tnbr", TnbrName_);
115      writeEntry(os, "thicknessLayers", thicknessLayers_);
116      writeEntry(os, "kappaLayers", kappaLayers_);
117
118      temperatureCoupledBase::write(os);
119 }
120
121
```

```
122  // * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
123
124  makePatchTypeField
125  (
126      fvPatchScalarField,
127      myTurbulentTemperatureCoupledBaffleMixedFvPatchScalarField
128  );
129
130  // * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
131  }
132  }
```

Implementation of *myVacuumBC* in *0/membrane/C*

```
1      outerWall
2      {
3          type              codedFixedValue;
4          value             uniform 0;
5          name              myVacuumBC;
6
7          code
8          #{
9                  const dictionary &thermophysicalProperties = db().lookupObject<
    IOdictionary>("thermophysicalProperties");
10             const dimensionedScalar D = dimensionedScalar("D", dimensionSet
    (0,2,-1,0,0,0,0),thermophysicalProperties);
11             scalar Kr_H = readScalar(thermophysicalProperties.lookup("Kr_H"));
12             const tmp<scalarField> delta = patch().deltaCoeffs();
13             const tmp<scalarField> intFld = this->patchInternalField();
14
15             scalarField Jmolecular = Kr_H*intFld()*intFld();
16             scalarField Jatomic = 2*Jmolecular;
17             scalarField newGrad= - Jatomic*(1.0/D.value());
18
19             (*this)== max(intFld() + newGrad/delta(),0.0);
20          #};
21
22      }
```

## A.3   How to compute the inlet velocity

The first (I) and second (II) passages in the PAV are each made of 8 parallel pipes: 3 short (S) + 3 medium (M) + 2 long (L). The inlet velocity is computed taking into account the different distributed pressure losses due to different lengths. The flow in parallel pipes has the following characteristics:

$$\begin{cases} \Delta p_S = \Delta p_M = \Delta p_L & \text{same pressure drop} \\ \dot{m}_{tot} = 3\dot{m}_S + 3\dot{m}_M + 2\dot{m}_L & \text{mass flow rate conservation} \end{cases} \tag{A.1}$$

The mass flow rate is $\dot{m}_i = \rho U_i A$ and the pressure drop is $\Delta p_i = f_i \frac{L_i}{D} \frac{\rho U_i^2}{2}$ where $i = S, M, L$. The friction factor $f_i$ is computed iteratively with the Colebrook correlation [37], assuming the roughness $\epsilon = 3e - 7m$ (for niobium tubes).
After some algebraic passage the three velocities are the following:

$$\begin{cases} U_L = \dfrac{\alpha}{1 + 3/2\,\beta\,\gamma + 3/2\,\gamma} \\ U_M = \gamma\,U_L \\ U_S = \beta\,U_M \end{cases} \tag{A.2}$$

with the parameters: $\alpha = \frac{\dot{m}_{tot}}{2\rho\,A}$    $\beta = \sqrt{\frac{f_M}{f_S}\frac{L_M}{L_S}}$    $\gamma = \sqrt{\frac{f_L}{f_M}\frac{L_L}{L_M}}$

The velocities are found with two nested iterative loops implemented in a MATLAB script: the outer one on the velocity and the inner one on the friction factor.

```matlab
%% Iterative cycle to find how the mass flow split in the // pipes according to
    their different length
% "1"= short, "2"= medium, "3"= long
% General data
T= 355+273; %[K]
pp= 129; %[Pa]
m_flow_tot=1.22; %[kg/s] TOT mass flow rate
D= 9.2e-3; %[m] pipe diameter
A_flow= pi*D^2/4; %[m^2] flow area
L_short= 1.805; %[m]
L_medium= 1.899;
L_long= 1.994;
L= [L_short L_medium L_long];
e= 3e-7; %[m] roughness from Goodfellow
e_rel= e/D; % relative roughness [-]
delta_p= 7e2*ones(size(L)); %[Pa] pressure drop between inlet/outlet

% make a guess on the friction factor and velocity
ff_guess= 0.1*ones(size(L));
U_guess= sqrt(2*D.*delta_p./(ff_guess.*L.*rho));

i=1; toll= 1e-3;
err_f= 2*toll; err_U= err_f;

while max(err_U)>toll
    i=i+1;

    Re= rho*D.*U_guess./mu;

```

```matlab
29        j=0;
30         while max(err_f)>toll %cycle to find friction factor with Colebrook (
    implicit)
31            j=j+1;
32            ff= (-2*log10(e_rel/3.71 + 2.51./Re./sqrt(ff_guess))).^-2; % find new ff
33            err_f= norm(ff-ff_guess)./ff;
34            ff_guess=ff;
35         end
36
37        A= 0.5*m_flow_tot/rho/A_flow;
38        B= sqrt(ff(2)/ff(1)*L(2)/L(1));
39        C= sqrt(ff(3)/ff(2)*L(3)/L(2));
40
41        % Imposing the two conditions for parallel flow we get 3 eq. for 3 unkowns
42        % 1) delta_p_S = delta_p_M = delta_p_L
43        % 2) m_flow tot= 3*m_flow_S + 3*m_flow_M + 2*m_flow_L
44        U(3)= A/(1+3/2*B*C+3/2*C); %long
45        U(2)= C*U(3);                    %medium
46        U(1)= B*U(2);                    %short
47
48         err_U= [abs(U(1)-U_guess(1))/U(1) abs(U(2)-U_guess(2))/U(2) abs(U(3)-U_guess
    (3))/U(3)];
49         U_guess= [U(1) U(2) U(3)];
50
51 end
52
53 m_flow= rho.*U_guess.*A_flow;
54 m_flow_8= 3*m_flow(1)+3*m_flow(2)+2*m_flow(3);
```

56

## A.4   Numerical error bar

| Input parameter | Mean value | Error | min/max | Effect on $\Phi$ | $\Phi_{PAV,OF,min}$ | $\Phi_{PAV,OF,max}$ |
|---|---|---|---|---|---|---|
| p [Pa] | 359 | 0.24% | 358.14 359.86 | ↑ | $p_{min}$ | $p_{max}$ |
| T [°C] | 451 | 0.49% | 448.78 453.22 | ↑ | $T_{min}$ | $T_{max}$ |
| $\dot{m}$ [kg/s] | 1.20 | 8.3% | 1.10 1.30 | ↑ | $\dot{m}_{min}$ | $\dot{m}_{max}$ |
| $K_{s,PbLi}$ [mol/m$^3$/Pa$^{0.5}$] | 7.87e-5 | 15% | 6.66e-5 9.08e-5 | ↑ | $K_{s,PbLi,min}$ | $K_{s,PbLi,max}$ |
| $D_{PbLi}$ [m$^2$/s] | 4.91e-09 | 10% | 4.33e-09 5.47e-9 | ↑ | $D_{min}(T_{min})$ | $D_{max}(T_{max})$ |
| $\rho_{PbLi}$ [kg/m$^3$] | 9657 | 0.30% | 9627 9690 | ↓ | $\rho_{max}(T_{min})$ | $\rho_{min}(T_{max})$ |

Table A.1: Test 3.1

| Input parameter | Mean value | Error | min/max | Effect on $\Phi$ | $\Phi_{PAV,OF,min}$ | $\Phi_{PAV,OF,max}$ |
|---|---|---|---|---|---|---|
| p [Pa] | 114 | 0.29% | 113.67 114.33 | ↑ | $p_{min}$ | $p_{max}$ |
| T [°C] | 356 | 0.52% | 354.16 357.84 | ↑ | $T_{min}$ | $T_{max}$ |
| $\dot{m}$ [kg/s] | 1.28 | 7.8% | 1.18 1.38 | ↑ | $\dot{m}_{min}$ | $\dot{m}_{max}$ |
| $K_{s,PbLi}$ [mol/m$^3$/Pa$^{0.5}$] | 2.10e-5 | 21% | 1.65e-5 2.55e-5 | ↑ | $K_{s,PbLi,min}$ | $K_{s,PbLi,max}$ |
| $D_{PbLi}$ [m$^2$/s] | 2.48e-09 | 10% | 2.19e-09 2.76e-09 | ↑ | $D_{min}(T_{min})$ | $D_{max}(T_{max})$ |
| $\rho_{PbLi}$ [kg/m$^3$] | 9771 | 0.30% | 9740 9803 | ↓ | $\rho_{max}(T_{min})$ | $\rho_{min}(T_{max})$ |

Table A.2: Test 4.1

| Input parameter | Mean value | Error | min/max | Effect on $\Phi$ | $\Phi_{PAV,OF,min}$ | $\Phi_{PAV,OF,max}$ |
|---|---|---|---|---|---|---|
| p [Pa] | 129 | 0.31% | 128.60 129.40 | ↑ | $p_{min}$ | $p_{max}$ |
| T [°C] | 355 | 0.52% | 353.16 356.84 | ↑ | $T_{min}$ | $T_{max}$ |
| $\dot{m}$ [kg/s] | 1.22 | 8.2% | 1.12 1.32 | ↑ | $\dot{m}_{min}$ | $\dot{m}_{max}$ |
| $K_{s,PbLi}$ [mol/m$^3$/Pa$^{0.5}$] | 1.22e-5 | 35% | 7.92e-6 1.65e-5 | ↑ | $K_{s,PbLi,min}$ | $K_{s,PbLi,max}$ |
| $D_{PbLi}$ [m$^2$/s] | 2.46e-09 | 10% | 2.19e-09 2.76e-09 | ↑ | $D_{min}(T_{min})$ | $D_{max}(T_{max})$ |
| $\rho_{PbLi}$ [kg/m$^3$] | 9773 | 0.30% | 9740 9803 | ↓ | $\rho_{max}(T_{min})$ | $\rho_{min}(T_{max})$ |

Table A.3: Test 5.1

| Input parameter | Mean value | Error | min/max | Effect on $\Phi$ | $\Phi_{PAV,OF,min}$ | $\Phi_{PAV,OF,max}$ |
|---|---|---|---|---|---|---|
| p [Pa] | 184 | 0.25% | 183.53 184.47 | ↑ | $p_{min}$ | $p_{max}$ |
| T [°C] | 355 | 0.52% | 353.14 356.86 | ↑ | $T_{min}$ | $T_{max}$ |
| $\dot{m}$ [kg/s] | 1.22 | 8.2% | 1.12 1.32 | ↑ | $\dot{m}_{min}$ | $\dot{m}_{max}$ |
| $K_{s,PbLi}$ [mol/m$^3$/Pa$^{0.5}$] | 1.36e-5 | 19% | 1.10e-5 1.62e-5 | ↑ | $K_{s,PbLi,min}$ | $K_{s,PbLi,max}$ |
| $D_{PbLi}$ [m$^2$/s] | 2.46e-09 | 10% | 2.17e-09 2.74e-09 | ↑ | $D_{min}(T_{min})$ | $D_{max}(T_{max})$ |
| $\rho_{PbLi}$ [kg/m$^3$] | 9773 | 0.30% | 9742 9804 | ↓ | $\rho_{max}(T_{min})$ | $\rho_{min}(T_{max})$ |

Table A.4: Test 6.1

| Input parameter | Mean value | Error | min/max | Effect on $\Phi$ | $\Phi_{PAV,OF,min}$ | $\Phi_{PAV,OF,max}$ |
|---|---|---|---|---|---|---|
| p [Pa] | 225 | 0.21% | 224.53 225.47 | $\uparrow$ | $p_{min}$ | $p_{max}$ |
| T [°C] | 355 | 0.53% | 353.13 356.87 | $\uparrow$ | $T_{min}$ | $T_{max}$ |
| $\dot{m}$ [kg/s] | 1.22 | 8.2% | 1.12 1.32 | $\uparrow$ | $\dot{m}_{min}$ | $\dot{m}_{max}$ |
| $K_{s,PbLi}$ [mol/m$^3$/Pa$^{0.5}$] | 1.02e-5 | 15% | 8.66e-6 1.17e-5 | $\uparrow$ | $K_{s,PbLi,min}$ | $K_{s,PbLi,max}$ |
| $D_{PbLi}$ [m$^2$/s] | 2.46e-09 | 10% | 2.17e-09 2.74e-09 | $\uparrow$ | $D_{min}(T_{min})$ | $D_{max}(T_{max})$ |
| $\rho_{PbLi}$ [kg/m$^3$] | 9773 | 0.30% | 9741 9804 | $\downarrow$ | $\rho_{max}(T_{min})$ | $\rho_{min}(T_{max})$ |

Table A.5: Test 7.2

# Bibliography

[1] IEA. *CO$_2$ Emissions in 2022*. Tech. rep. https://www.iea.org/reports/co2-emissions-in-2022, 2023.

[2] `https://euro-fusion.org/eurofusion/roadmap/`.

[3] Pijush Bhattacharjee. "Modified Atomic Theory Identifies Formation of Superconductor". In: 2 (Feb. 2010), pp. 65–71. DOI: `10.7763/IJET.2010.V2.102`.

[4] Peter Dunn. "On the right path to fusion energy -National Academies study recommends a pilot fusion energy program that aligns with MIT's fusion approach and SPARC project". In: *MIT News* (2018).

[5] Joseph Antony. "Geological timeline of significant events on Earth". In: *Water Worlds in the Solar System* (2023), pp. 55–114. DOI: `10.1016/b978-0-323-95717-5.00020-7`.

[6] M. Kovari et al. "Tritium resources available for fusion reactors". In: *Nuclear Fusion* 58.2 (2017). DOI: `10.1088/1741-4326/aa9d25`.

[7] W. Biel. "Tritium Breeding and blanket technology". In: *Bad Honnef* (2014).

[8] Alessandro Tassone et al. "Recent Progress in the WCLL Breeding Blanket Design for the DEMO Fusion Reactor". In: *IEEE Transactions on Plasma Science* 46 (2018), pp. 1446–1457. URL: `https://api.semanticscholar.org/CorpusID:19123275`.

[9] *OpenFOAMwiki*. `https://openfoamwiki.net/index.php/ChtMultiRegionFoam`. chtMultiRegionFoam.

[10] W. Malalasekera H.K. Versteeg. *An Introduction to Computational Fluid Dynamics*. Second Edition, 2007.

[11] F. Papa et al. "Engineering design of a Permeator Against Vacuum mock-up with niobium membrane". In: *Fusion Engineering and Design* 166 (2021), p. 112313. ISSN: 0920-3796. DOI: `https://doi.org/10.1016/j.fusengdes.2021.112313`. URL: `https://www.sciencedirect.com/science/article/pii/S0920379621000892`.

[12] Paul W. Humrickhouse and Brad J. Merrill. "Vacuum Permeator Analysis for Extraction of Tritium from DCLL Blankets". In: *Fusion Science and Technology* 68.2 (2015), pp. 295–302. DOI: `10.13182/FST14-941`. eprint: `https://doi.org/10.13182/FST14-941`. URL: `https://doi.org/10.13182/FST14-941`.

[13] H. Weller C. Greenshields. `https://doc.cfd.direct/notes/cfd-general-principles/axisymmetric-wedge-condition`. Notes on CFD. 2022.

[14] Alex Roqué Velasco. *Boundary conditions for convective heat transfer in OpenFOAM*.

[15] Reiter F. "Solubility and Diffusivity of Hydrogen Isotopes in Liquid Pb-17Li". In: *Fusion Engineering and Design* 14 (1991), pp. 207–211.

[16] Kenji Komiya et al. "Alloying Effects on Hydrogen Permeability of Niobium". In: *PRICM-5*. Vol. 475. Materials Science Forum. Trans Tech Publications Ltd, Jan. 2005, pp. 2497–2500. DOI: `10.4028/www.scientific.net/MSF.475-479.2497`.

[17] Takayuki Terai et al. "Diffusion coefficient of tritium in molten lithium-lead alloy (Li17Pb83) under neutron irradiation at elevated temperatures". In: *Journal of Nuclear Materials* 187.3 (1992), pp. 247–253. ISSN: 0022-3115. DOI: `https://doi.org/10.1016/0022-3115(92)90504-E`. URL: `https://www.sciencedirect.com/science/article/pii/002231159290504E`.

[18] S. A. Steward. "Review of Hydrogen Isotope Permeability Through Materials". In: 1.1 (Aug. 1983). DOI: `10.2172/5277693`. URL: `https://www.osti.gov/biblio/5277693`.

[19] American Society of Mechanical Engineers. *Standard for Verification and Validation in Computational Fluid Dynamics and Heat Transfer*. ASME V&V 20, 2019.

[20] Francesca Papa et al. "Tritium Extraction from Lithiumn-Lead Eutectic Alloy: Experimental Characterization of a Permeator against Vacuum Mock-Up at 450 deg;C". In: *Energies* 16.7 (2023). ISSN: 1996-1073. DOI: `10.3390/en16073022`. URL: `https://www.mdpi.com/1996-1073/16/7/3022`.

[21] Yuki Edao, Hidetaka Noguchi, and Satoshi Fukada. "Experiments of hydrogen isotope permeation, diffusion and dissolution in Li–Pb". In: *Journal of Nuclear Materials - J NUCL MATER* 417 (Oct. 2011), pp. 723–726.

[22] D. Martelli, A. Venturini, and M. Utili. "Literature review of lead-lithium thermophysical properties". In: *Fusion Engineering and Design* 138 (2019), pp. 183–195. ISSN: 0920-3796. DOI: `https://doi.org/10.1016/j.fusengdes.2018.11.028`. URL: `https://www.sciencedirect.com/science/article/pii/S0920379618307361`.

[23] Y.C. Chan and E. Veleckis. "A thermodynamic investigation of dilute solutions of hydrogen in liquid Li-Pb alloys". In: *Journal of Nuclear Materials* 123.1 (1984), pp. 935–940. ISSN: 0022-3115. DOI: `https://doi.org/10.1016/0022-3115(84)90198-3`. URL: `https://www.sciencedirect.com/science/article/pii/0022311584901983`.

[24] Richard Schumacher and Alarich Weiss. "Hydrogen Solubility in the Liquid Alloys Lithium-Indium, Lithium-Lead, and Lithium-Tin". In: *Berichte der Bunsengesellschaft für physikalische Chemie* 94.6 (1990), pp. 684–691. DOI: `https://doi.org/10.1002/bbpc.19900940612`. eprint: `https://onlinelibrary.wiley.com/doi/pdf/10.1002/bbpc.19900940612`. URL: `https://onlinelibrary.wiley.com/doi/abs/10.1002/bbpc.19900940612`.

[25] E. Mas de les Valls et al. "Lead–lithium eutectic material database for nuclear fusion technology". In: *Journal of Nuclear Materials* 376.3 (2008), pp. 353–357.

[26] A. Aiello, A. Ciampichetti, and G. Benamati. "Determination of hydrogen solubility in lead lithium using sole device". In: *Fusion Engineering and Design* 81.1 (2006). Proceedings of the Seventh International Symposium on Fusion Nuclear Technology, pp. 639–644. ISSN: 0920-3796. DOI: `https://doi.org/10.1016/j.fusengdes.2005.06.364`. URL: `https://www.sciencedirect.com/science/article/pii/S092037960500565X`.

[27] H. Katsuta, H. Iwamoto, and H. Ohno. "Hydrogen solubility in liquid Li17Pb83". In: *Journal of Nuclear Materials* 133-134 (1985), pp. 167–170. ISSN: 0022-3115. DOI: `https://doi.org/10.1016/0022-3115(85)90127-8`. URL: `https://www.sciencedirect.com/science/article/pii/0022311585901278`.

[28] Silvano Tosti et al. "Hydrogen solubility and electrical resistivity measurements of hydrogenated Pb-Li". In: *2013 IEEE 25th Symposium on Fusion Engineering, SOFE 2013* (May 2013), pp. 1–3. DOI: 10.1109/SOFE.2013.6635456.

[29] Yoshinori Shibuya et al. "Isothermal release of tritium from neutron-irradiated Li17Pb83". In: *Journal of Nuclear Materials* 150.3 (1987), pp. 286–291.

[30] G.X. Zhang et al. "Alloying effects of Ru and W on hydrogen diffusivity during hydrogen permeation through Nb-based hydrogen permeable membranes". In: *International Journal of Hydrogen Energy* 35.3 (2010), pp. 1245–1249. ISSN: 0360-3199. DOI: https://doi.org/10.1016/j.ijhydene.2009.11.066. URL: https://www.sciencedirect.com/science/article/pii/S0360319909018369.

[31] Marta Malo, Belit Garcinuño, and David Rapisarda. "Experimental refutation of the deuterium permeability in vanadium, niobium and tantalum". In: *Fusion Engineering and Design* 146 (Dec. 2018).

[32] R. Cantelli, F. M. Mazzolai, and M. Nuovo. "Internal Friction due to Long-Range Diffusion of Hydrogen in Niobium (Gorsky Effect)". In: *physica status solidi (b)* 34.2 (1969), pp. 597–600. DOI: https://doi.org/10.1002/pssb.19690340221. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/pssb.19690340221. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/pssb.19690340221.

[33] M.A. Pick and K. Sonnenberg. "A model for atomic hydrogen-metal interactions — application to recycling, recombination and permeation". In: *Journal of Nuclear Materials* 131.2 (1985), pp. 208–220. ISSN: 0022-3115. DOI: https://doi.org/10.1016/0022-3115(85)90459-3. URL: https://www.sciencedirect.com/science/article/pii/0022311585904593.

[34] Luigi Candido, Ciro Alberghi, and M. Utili. "Design of a hydrogen/tritium permeation sensor for Gen-IV sodium fast reactors". In: *Vacuum* (June 2021), p. 110414. DOI: 10.1016/j.vacuum.2021.110414.

[35] Ciro Alberghi et al. "Development of new analytical tools for tritium transport modelling". In: *Fusion Engineering and Design* 177 (2022), p. 113083. ISSN: 0920-3796. DOI: https://doi.org/10.1016/j.fusengdes.2022.113083. URL: https://www.sciencedirect.com/science/article/pii/S0920379622000837.

[36] Peter Harriott and R. M. Hamilton. "Solid-liquid mass transfer in turbulent pipe flow". In: *Chemical Engineering Science* (1965). DOI: 10.1016/0009-2509(65)80110-5.

[37] Frank P. Incropera and David P. DeWitt. *Fundamentals of Heat and Mass Transfer*. 4th Edition. New York City, New York: John Wiley and Sons, Inc., 1996.

[38] F. Lisanti et al. "First comparison of the GETTHEM Tritium Transport and Permeation model Against Experimental Data". In: 2023.

[39] Roberto Bonifetto et al. "Modeling the PbLi flow including tritium transport and permeation with GETTHEM". In: *Nuclear Materials and Energy* 37 (Sept. 2023), p. 101500. DOI: 10.1016/j.nme.2023.101500.

# Ringraziamenti

Vorrei usare questo spazio per ringraziare le persone che mi hanno aiutata in questo percorso.

Un sentito grazie ai miei relatori a Barcellona per la loro disponibilità e accoglienza, specialmente a Ronny per avermi sempre spronata anche nelle difficoltà (quasi sempre OpenFOAM-related).

Un ringraziamento anche ai miei relatori a Torino, di cui ho seguito innumerevoli corsi, per saper trasmettere la passione ed esperienza che hanno per questo argomento.

Un grazie non scontato ai miei genitori, che mi hanno permesso di arrivare fin qui, supportandomi sempre e dandomi fiducia anche quando io non ne avevo in me.

Un grazie alle splendide ragazze con cui ho condiviso questo percorso, Betta, Cecilia, Eleonora e Eleonora. Voi mi ispirate ogni giorno con la vostra passione e intelligenza, e non vedo l'ora di vedere dove saremo tra qualche anno.

Ma, sopra a tutti, grazie a Baussano, senza il quale non sarei qui.

Grazie.