WPBB-PR(17) 18302

V Pasler et al.

Development and verification of a component-level hydrogen transport model for a DEMO-like HCPB breeder unit with OpenFOAM

Preprint of Paper to be submitted for publication in
Fusion Engineering and Design

This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 under grant agreement No 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

This document is intended for publication in the open literature. It is made available on the clear understanding that it may not be further circulated and extracts or references may not be published prior to publication of the original when applicable, or without the consent of the Publications Officer, EUROfusion Programme Management Unit, Culham Science Centre, Abingdon, Oxon, OX14 3DB, UK or e-mail Publications.Officer@euro-fusion.org

Enquiries about Copyright and reproduction should be addressed to the Publications Officer, EUROfusion Programme Management Unit, Culham Science Centre, Abingdon, Oxon, OX14 3DB, UK or e-mail Publications.Officer@euro-fusion.org

The contents of this preprint and all other EUROfusion Preprints, Reports and Conference Papers are available to view online free at http://www.euro-fusionscipub.org. This site has full search facilities and e-mail alert options. In the JET specific papers the diagrams contained within the PDFs on this site are hyperlinked

Simulation of tritium transport inside a DEMO-like HCPB breeder unit with OpenFOAM

Volker Pasler*, Frederik Arbeiter, Christine Klein, Dmitry Klimenko, Georg Schlindwein, Axel von der Weth

Karlsruhe Institute of Technology, P.O.Box 3640, D-76021 Karlsruhe, Germany

This work describes the development of a numerical model to simulate transient tritium transport on the BU level for the EU HCPB concept for DEMO. The key output quantities of the model are the tritium concentration in the purge gas and in the coolant and the tritium inventory inside the BU structure. The model capabilities should cover normal operation as well as accident conditions.

The Open Source Field Operation And Manipulation framework OpenFOAM serves as the basis for the model. Equations and boundary conditions required for hydrogen isotopes transport are implemented. Realistic properties data as diffusion constants and Sieverts constants are required, too. A key model issue is solid-fluid interface mass transfer. Here two correlations that (1) approaches Sieverts equilibrium in the diffusion limit and (2) a rate dependent correlation that includes the diffusion limit for very high ad-/desorption rate constants are introduced. A two species interface mass transfer correlation based on the single species rate dependent correlation is developed, too. First verification calculations are compared to analytic solutions and TMAP calculations.

Keywords: tritium transport, safety, OpenFOAM, HCPB.

1. Introduction

Tritium as one of the two necessary fuels for the currently technically pursued D-T fusion process will have to be produced inside the fusion plant blanket itself e.g. starting from lithium making use of the fusion neutrons. For the so called breeding of tritium, one European concept for DEMO called HCPB (helium cooled pebble bed) uses pebble beds of lithium orthosilicate (OSI) and beryllium. Beryllium is foreseen as a neutron multiplier to gain the required breeding neutrons at suitable energies to breed tritium in the neighboring OSI bed. High pressure helium serves as coolant; a separated stream of low pressure helium, usually called the purge gas, is used to transport the bred tritium out of the breeder to the tritium extraction system (TES). Usually a small amount of hydrogen is added to the purge gas to support tritium extraction. A breeder blanket typically is build-on of several identical or at least very similar breeder units (BUs). These BUs represent a practicable level for Computational Fluid Dynamics (CFD) modeling.

The attenuation of nuclear interaction processes with increasing distance from the first wall (FW) and the cooling configuration will result in considerable temperature spreads, tritium generation profiles and different grades of radiation damage to the structure material (Eurofer-97) as well as to the breeder ceramics (OSI) and neutron multiplier material (Be) inside a BU. Transport parameters and tritium retention properties are known to depend considerably on these quantities.

The design of a BU has to take into account these parameters with regard to the breeding efficiency and the self-sufficiency of the reactor with fuel. The tritium inventory in the components plays a role with regard to safety and decommissioning requirements.

The above boundary conditions (BCs) obviously ask for multiple physics capabilities when analyses are to be done. Thermomechanical analyses of a BU by the designers with commercial CFD and FEM codes are standard proceeding. However simulation options for tritium release and transport are still incomplete. The available tritium transport modeling codes typically operate at the system level or at a microscopic level. Suitable tools that describe the tritium release and transport behavior inside a BU on the component level are still missing. A publication [1] indicates the awareness of the EU fusion community responsible of the need for a component level model closing the gap between the well-known TMAP code [2] and the diverse system level codes. Notably a component level model is expected to improve the accuracy of the answers on questions related to safety (such as tritium inventory, tritium retention and contamination of the coolant flows) and to the TES design. First efforts with ANSYS [3] have not been further developed. More recent work had been done in the US at UCLA based on commercial codes like SC/TETRA [4] and COMSOL [5].

This work introduces a tritium transport model on the BU level based on the open source CFD framework OpenFOAM. The complete work concept also includes the identification of suitable properties data what is published elsewhere [6].

2. Physical correlations and properties data

2.1 Gas phase processes

The two relevant gas phases are (1) the purge gas flow through the pebble beds (where a porous media approach is applicable) and (2) the coolant channels with higher helium pressure (~80bar) [7]. In both cases

* *Corresponding author*
E-mail address: volker.pasler@kit.edu

hydrogen isotopes are assumed as passive scalars that do not to contribute to the flow field equations.

It is foreseen to support tritium release from the breeder zone by isotope exchange processes. This is done by adding typically 0.1% hydrogen to the purge gas. Assuming dry purge gas, this means in practice that the hydrogen molecules Q_2 (H_2, HT, ...) in the BU will be mainly H_2. In an equilibrium composition of isotopes, most tritium will be present as HT while the T_2 fraction will be close to negligible for most aspects.

2.1.1 Hydrogen isotopes transport in pebble beds

A first approach of a transient Q_2 transport equation includes (from left to right) diffusion, convection and a source term:

$$\frac{\partial c}{\partial t} - \Delta\left(D_{Q_2} c\right) + \nabla\cdot\left(vc\right) + q_{Q_2} = 0 \quad (2.1)$$ with

c = Q_2 molecules concentration [m^{-3}],

D_{Q_2} = diffusivity of Q_2 in purge gas [m^2/s],

v = purge flow speed [m/s],

q_{Q2} = Q_2 volumetric source term [$1/m^3/s$].

The helium purge gas flow is taken as a background field with flow velocities in the order of typically about 1cm/s. Diffusion is modeled as binary ideal gas diffusion between Q_2 and helium. Interaction between the different species molecules is neglected with regard to diffusion.

The Chapman-Enskog formula [8] provides the basic correlation for binary gas diffusion transport coefficient:

$$\mathrm{D_{AB}(T)} = 1.8583\, x 10^{-7} \frac{\sqrt{\mathrm{T^3\left(1/M_A + 1/M_B\right)}}}{\mathrm{p}\,\sigma_{AB}\,\Omega_{AB}} \left[\mathrm{m^2/s}\right] \quad (2.2)$$

In a porous media (pebble bed), the impact of porosity ε (volume ratio void/solid), tortuosity $\tau = \left(\frac{l_{eff}}{l_{straight}}\right)^2$ or $\left(\frac{1}{\cos\Theta}\right)^2$ (Θ is the mean angle between pore direction and transport direction) and constrictivity $\delta = \frac{d_{particle}}{d_{pore}} < 1$ is taken into account by an effective diffusion parameter. The effective gas diffusion coefficient for hydrogen molecules Q_2 in the helium purge gas will be:

$D_{eff} = \frac{\varepsilon\,\delta}{\tau}\mathrm{D_{AB}}$ with A=Q_2 and B=He. The required data for σ_{Q_2He} and Ω_{Q_2He} are listed in table 2 of [8].

2.1.2 Hydrogen isotopes transport in coolant channels

The transport equation (2.1) is used in the coolant channels, too. Helium has very small van-der-Waals constants. According to [9], the deviation of Helium between real gas law and ideal gas law at 60bar and 550K is about 1.5%. This helps to justify the usage of the Chapman-Enskog diffusion equation (2.2) for the helium coolant gas flow at about 80bars, too. Moreover a possible inaccuracy in diffusion is expected to play a minor role because the coolant flow velocities are in the order of tens of m/s. Consequently wall turbulence will significantly contribute to perpendicular transport of hydrogen away from and to the coolant channel wall. For completeness it should be noticed anyway that an extension of the diffusion model to "moderate dense gases of rigid spheres" would be available in the frame of the Thorne-Enskog theory [10].

2.2 Hydrogen isotopes transport in solid structures

Diffusion and trapping are the most relevant processes in the Eurofer-97 walls of the BU. Most Q-diffusion is expected to be H (protium) from the purge gas additions while tritium will be only a small accompanying load. As the purge gas will be present long before tritium release starts, traps in the metal structure are expected to be already preferentially occupied by protium and play no significant role for tritium any more. However different trap potentials of the isotopes imply the possibility to change this state by thermal cycling during longer operation. A first approach of a Q diffusion equation is:

$$\frac{\partial c}{\partial t} - \Delta\left(D_{QL}\, c\right) + q_Q = 0$$

D_{QL} = lattice diffusion coefficient [m^2/s] depending on temperature taken from material database.

Equations for diffusion including trapping as proposed by McNabb and Foster may be found e.g. at [11]:

$$\frac{\partial}{\partial t}\left(\mathrm{c} + \sum_{i=1}^{m} \mathrm{N_i}\Theta_i\right) = D_{QL}\frac{\partial^2 c}{\partial x^2} \quad (2.3)$$ and

$$\frac{\partial \Theta_i}{\partial t} = k\mathrm{c} - \mathrm{p}\,\Theta_i \quad (2.4)$$ with

N_i = density of high energy "i" type sites [mol/m^3], m different types are assumed,

Θ_i = fraction of occupied "i" traps,

k = trapping kinetic constant [$m^3/mol/s$],

p = detrapping kinetic constant [1/s].

Under the assumption of local equilibrium (= Oriani model), an equilibrium constant may be defined as:

$$K_i = \frac{k_i}{p_i} = \mathrm{K_{i,0}}\exp\left(-\frac{\Delta G_i}{\mathrm{RT}}\right)$$

With the lattice/normal diffusion sites occupation $\Theta_L << 1$

$$K_{\mathrm{i}} = \frac{\Theta_i}{\mathrm{c}(1-\Theta_i)}$$

In this case the above McNabb Foster coupled equations 2.3 and 2.4 reduce to the single equation:

$$\frac{\partial}{\partial t}\left(\mathrm{c} + \sum_{\mathrm{i}=1}^{\mathrm{m}} \mathrm{N}_{\mathrm{i}} \frac{K_i c}{1 + K_i c}\right) = D_{\mathrm{L}} \frac{\partial^2 c}{\partial x^2}$$

It is however unclear if $\Theta_{\mathrm{L}} << 1$ is a valid assumption.

2.3 Diffusion parameters and Sieverts coefficient numerical data

The relevant literature data for RAFM steel was reviewed and condensed to selected best estimate data for Eurofer-97 structures [6]. Criteria were amongst others steel admixtures that effect hydrogen transport properties and heat treatments during manufacturing.

According to this work, the best estimate seems the data for Optifer published in [12]. The corresponding correlation for the diffusion coefficient of protium is:

$$\mathrm{D}_{\mathrm{eff}} = \frac{\mathrm{D}}{\left(1 + 4.23\ 10^{-6} \exp\left(\frac{52.2 kJ/mol}{RT}\right)\right)} \quad \text{with}$$

$$\mathrm{D} = 5.49\ 10^{-8} \exp\left(\frac{-10.6 kJ/mol}{RT}\right) \left[\frac{\mathrm{m}^2}{s}\right]$$

$\mathrm{D}_{\mathrm{eff}}$ includes trapping effects. The impact of trapping grows with decreasing temperature.

The best estimate for the Sieverts coefficient would be again the data of Optifer published in [12]. The corresponding correlation for the Sieverts coefficient is:

$$\mathrm{K}_{\mathrm{S,eff}} = \mathrm{K}_{\mathrm{S}}\left(1 + 4.23\ 10^{-6} \exp\left(\frac{52.2 kJ/mol}{RT}\right)\right) \quad \text{with}$$

$$\mathrm{K}_{\mathrm{S}} = 0.328 \exp\left(\frac{-29 kJ/mol}{RT}\right) \left[\frac{\mathrm{mol}}{m^3\sqrt{Pa}}\right]$$

$\mathrm{K}_{\mathrm{S,eff}}$ includes trapping effects.

3 General concept of mass transport modeling with OpenFOAM

In OpenFOAM fluid and solid regions are usually solved sequentially region by region in each time step. The interaction of quantities between neighboring regions is realized by BCs that are specified at the corresponding region interfaces. E.g. the so called "turbulentTemperatureCoupledBaffleMixed" thermal BC can operate together with the conjugate heat transfer solver "chtMultiRegionFoam" to allow for a full CFD thermal analysis of a system with solid and fluid components in OpenFOAM. As heat and mass transfer are formally similar, mass transfer capabilities can be implemented to OpenFOAM quite straightforward by the extension and modification of the above parts of the original code and -what makes things much easier- only of them.

In OpenFOAM solver and BC are separate issues in the view of formal programming aspects. The solver contains the C++ program "main". This means a new solver executable has to be compiled if equations are changed or added. The below work includes additional equations to the original "chtMultiRegionFoam" solver while the full original conjugate heat transfer capabilities are kept. This maintains the option of doing also a thermal analysis together with the hydrogen transport analysis inside a BU. However the import of elsewhere calculated temperature data (with dedicated meshes) seems the most efficient approach.

A minimum of changes to the existing OpenFOAM installation helps to avoid errors and to maintain the portability between versions. For these reasons the BCs are compiled separately and embedded as dynamic link libraries (DLLs) at runtime.

Below some terminologies of the OpenFOAM C++ code are used that are not common to a typical reader or OpenFOAM user. They are not required for a qualitative understanding of the model and can just be taken as names for variables or functions that do what is described in the text below. Anyway they had been included to clarify for an interested reader how the issues had been actually implemented to the code. The meaning of these code specific terminologies is explained in the OpenFOAM online documentation [13].

For readers with advanced interest in OpenFOAM it may be interesting to know that this work bases on the branch that may be found at www.openfoam.org. This work started with OpenFOAM version 2.2.2, our present OpenFOAM version is 4.1. Portability of our work between these and intermediate OpenFOAM versions had been possible with minor changes. There is another OpenFOAM branch at www.openfoam.com which basically provides the same original functionality and may be run with almost identical input files. However decisive C++ classes and members inside this source code are different. Unfortunately our source code cannot be adapted and compiled that easily with these versions although it should be possible with some knowledge on the classes of the openfoam.com branch. The same is true for the extend-project branch (www.extend-project.de). From our coarse comparison, there the classes in question seem to be very similar like in the www.openfoam.com branch.

4 OpenFOAM mass transport models

4.1 Simple Sieverts law mass transport BC for fluid-solid interfaces

4.1.1 Physics modeled by the BC

This BC assumes that transport to and from the surface is governed by diffusion processes only. The mass flow from the fluid to the solid is calculated by assuming fluid and solid diffusion in series. At the

surface the Sieverts correlation is used to "convert" a fluid concentration (or partial pressure) $C_{g,0}$ to a solid concentration $C_{s,0}$ which corresponds to the assumption of equilibrium conditions.

In contrast to the original temperature BC, the BC boundary value is not equal for both sides. The Sieverts correlation leads to a quadratic equation that has to be solved to gain the corresponding fluid/solid concentrations $C_{g,0}$ and $C_{s,0}$ at the boundary that result in the same mass flow on both sides. Figure 4.1 illustrates the situation at the boundary.

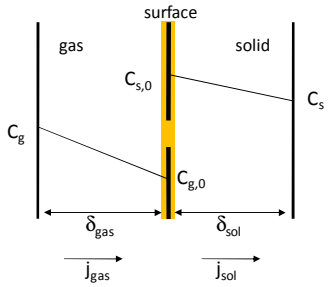

Figure 4.1: Illustration of relevant concentrations and parameters for the surface mass transfer model.

The BC assumes that the surface should not store mass, i.e. at any time mass (atom) flows $j_{gas} = j_{sol}$ are equal with:

$$j_{gas} = -2D_{gas}\frac{C_g - C_{g,0}}{\delta_{gas}} \quad \text{and} \quad j_{sol} = -D_{sol}\frac{C_{s,0} - C_s}{\delta_{sol}}$$

D_{gas}, D_{sol} are the diffusion coefficients, C_g, C_s are the mesh center concentrations, $C_{g,0}, C_{s,0}$ are the surface concentrations, $\delta_{gas}, \delta_{sol}$ are the distances from the mesh center to the surface in the gas/solid phase, respectively. The factor of 2 takes into account that the gas concentration represents H_2 molecules and not the solid concentration of H atoms. Defining $\Delta_g = 2D_{gas}/\delta_{gas}$ and $\Delta_s = D_{sol}/\delta_{sol}$ the equation $j_{gas} = j_{sol}$ can be written as:

$$\Delta_g\left(C_g - C_{g,0}\right) = \Delta_s\left(C_{s,0} - C_s\right) \tag{4.1}$$

Sieverts law correlates the solid surface concentration to the partial pressure of the assumed ideal gas by:

$$C_{s,0} = K_s(T_0)\sqrt{p_0} = K_s(T_0)\sqrt{C_{g,0}kT_0} \tag{4.2}$$

Eq. 4.2 is squared and inserted into eq. 4.1 to get

$$\Delta_g\left(C_g - \frac{C_{s,0}^2}{K_S^2\, kT_0}\right) = \Delta_s\left(C_{s,0} - C_s\right) \tag{4.3}$$

This is transformed to a quadratic equation for $C_{s,0}$ with the solution for the BC value:

$$C_{s,0} = -\frac{K_S^2\, kT_0\, \Delta_s}{2\Delta_g} + K_S\sqrt{kT_0}\sqrt{\frac{K_S^2\, kT_0}{4}\left(\frac{\Delta_s}{\Delta_g}\right)^2 + C_g + \left(\frac{\Delta_s}{\Delta_g}\right)C_s} \tag{4.4}$$

Now from eq. 4.2 also the gas side BC value can be calculated as:

$$C_{g,0} = \frac{C_{s,0}^2}{K_S^2\, kT_0} \tag{4.5}$$

4.1.2 Realization of the BC in OpenFOAM

The mass transfer BC basically works like the original thermal BC with the additional step of the evaluation of the Sieverts correlation. The BC values (*stored to "this->refValue()" of the corresponding OpenFOAM boundary patch fields*) are just set to the values of $C_{g,0}$ and $C_{s,0}$.

One still open issue is that the evaluation of the BC's in OpenFOAM is done separately for each region. This means the change of fluid temperature or concentration (the fluid regions loop is evaluated before the solid regions) would slightly affect the heat or mass flow in the second calculation. One can overcome this by dropping the original (formally beautifully programmed) solution of a symmetric call of the BC. Instead the $C_{s,0}$ calculated in the first (fluid) call is stored and used for the second (solid) call in the solid regions, too.

Unfortunately this step in the right direction results in a still poor accuracy of the interface mass balance. This is a principle problem caused by the change of the fixed value BC during the iteration cycles. The error caused by this effect should be only zero if $D_{gas}/\delta_{gas} = D_{sol}/\delta_{sol}$. As however hydrogen diffusion coefficients in HCPB relevant solids and fluids typically differ by several orders of magnitude the error is not expected to be small. In extreme test cases up to 40% of the transferred mass got lost or was added. Similarly the heat balance of the original thermal BC suffers from the same issue as soon as the OpenFOAM quantity "*kappa*deltaCoeffs()*" becomes significantly different on the solid and the fluid side of the boundary.

In the next section this mass balance issue is fixed by replacing the original BC functionality by the introduction of suitable source term fields.

4.2 Mass conservative Sieverts interface BC

4.2.1 Formal concept of the mass conservative version

The below described alternative BC-like formalism separates inter region mass exchange from the prescription of BC values. Instead interface mass flows are calculated only once in each time step and set as additional source terms in both sides boundary face cells. The BC correlations of section 4.1.1 can still be used. The mass conservative version will operate as follows:

(1) calculate the interface mass flows according to Sieverts law only once in first call of the fluid side loop of the BC: $j_{gas} = -2D_{gas}\frac{C_g - C_{g,0}}{\delta_{gas}}$, $j_{sol} = -j_{gas}$ This

calculation must not be repeated during the inner iterations as they already include mass transfer.

(2) record calculated mass sources to an additional source term field at the surface/interface mesh locations (like *fluidSourceTermField.patchInternalField()*).

(3) prescribe zero gradient BCs for the concentration variables C_g and C_s at the interfaces, i.e. mass transfer by the BCs of the concentration equations now will be zero (mass transfer calculated from eq. 4.1 would necessarily slightly change during inner iterations).

(4) transfer the negative source term to the solid region side (like s*olidSourceTermField.patchInternal Field()*). Here a factor of 2 plays a role because the concentration equations are for atoms in solids but for molecules in fluids.

4.2.2 Equations for the mass conservative version

The introduction of the new source term fields to the solid and fluid mass transport equations is taken as an opportunity to show how the mass transport equations are represented in the OpenFOAM C++ notation. FcH2 is the fluid hydrogen (molecules) concentration, FH2Src is a usual source term e.g. from pebble bed volumetric hydrogen release, the new source field FH2BdySrc is exclusively used by the BC. The fluid side equation becomes:

```
// H2 transport equation in fluid region
fvScalarMatrix TpH2Eqn(
// d/dt
fvm::ddt(FcH2)
// diffusion term
-fvm::laplacian(DFH2,FcH2, „laplacian(DH,cH)“)
// convection term, basically div( phi/rho * FcH2)
+fvm::div(fvc::absolute(phi/fvc::interpolate(rho), U),
                FcH2,
                „div(phi,FcH2)“
                )
// "usual" source term in pebble bed
- FH2Src/mesh.V()
// exclusive source term for wall transfer BC
- FH2BdySrc/mesh.V()
);
```

The solid side transport equation has identical terms except it misses the convection term. On the solid side the concentration variable is ScH, the new BC specific source term field is SHBdySrc.

4.2.3 Setting the BC specific source term fields

In OpenFOAM usually the boundary fields for a variable are evaluated by the solver just before the equation matrix is solved. The BCs for the concentration variables FcH2 and ScH are still processed by the code following this guideline. As a side effect the BC's of both fields are no longer occupied by the surface mass transfer model. This turned out to be advantageous when recalculating 1D scenarios modeled by the TMAP code (section 5) and might be useful if further surface effects are to be added to the model in a later stage. Anyway always reasonable BCs are required and in typical cases and for now just the standard OpenFOAM BC "zeroGradient" is specified at the solid-fluid interface from both sides.

In OpenFOAM the new fields FH2BdySrc and SHBdySrc formally also have boundary fields by default. But there are no equations for these variables so BC's are never evaluated. Now it is useful to know that also a direct call of the BC evaluation routines by the user is possible. E.g.

```
FH2BdySrc.boundaryField().updateCoeffs();
```

may be used to calculate the interface mass flows using the BC evaluation routines. The call has to be done inside the fluid regions loop but outside the inner iterations loop for the equation for the variable FcH2. This way repetitive calls (see (1) in section 4.2.1) in inner iterations are avoided safely.

This proceeding requires a formal BC entry for the source term fields FH2BdySrc in the input like:

```
FH2BdySrc
{
 boundaryField
  {
    „fluidBed_to_solidWall“
    {
    type        hwtSievertsSingle;
    Tnbr        SHBdySrc;
    value       uniform 1.00E+24;
    }
  }
}
```

where "fluidBed" and "solidWall" are example names of a fluid and a solid region, respectively. On the solid side a very similar entry is required for SHBdySrc.

Note that the BC call is done for the boundary field (i.e. the surface patches) and consequently results will become available on that level (as member of the BC C++ class "hwtSievertsSingleFvPatchScalarField"). On the other hand, the mass transfer calculation results are not required as BC's any more but now they will have to find their way directly into the FH2BdySrc field of the OpenFOAM matrix equation in section 4.2.2. The solution is to record the surface mass transfer into the first surface layer of the internal field of FH2BdySrc. Here a difficulty appears because the mass flows are calculated in the BC patch field classes environment but due to the source term field concept they are now also required in the volume field classes environment used by the matrix equation formalism. Because the programmers of OpenFOAM attached great importance to declare C++ classes and member as **const**(ant) to avoid unintended data modification, a reasonable write access between members of different classes typically asks for making non-constant references using **const_cast**.

A look into the OpenFOAM Doxygen documentation showed that most of the functionalities that are useful for our purposes already exist in the OpenFOAM code repository. So a minimum of own code had to be added. A new member was defined for the missing key functionality to record the mass flows into the source term (internal) fields mesh elements close to the surface.

Thanks to the existence of the member “faceCells()” which returns the cell numbers of the mesh elements just at the surface next to the current interface patch, the corresponding member "addPatchValueToInternalField" can be defined quite briefly as:

```
template<class Type>
void hwtSievertsSingleFvPatchScalarField::addPa
tchValueToInternalField
(const UList<Type>& f, Field<Type>& pif) const

{
 const labelUList& faceCells =
                   this->patch().faceCells();
    forAll(faceCells, facei)
    {
       pif[ faceCells[facei] ]
     = pif[ faceCells[facei] ] + f[facei];
    }
}
```

As regions might have multiple boundaries - e.g. a fluid mesh element in a corner may have several interfaces to different solid regions - an additive treatment ("+="-functionality) seemed appropriate. This however implies that one must not call the BC during inner iterations but only once at the begin of a time step for each region. One also has to reset the source term (internal) fields to zero before each new time step.

As already stated for the BC version in section 4.1.2, a calculation is only done for the fluid side of each interface. On the solid side the call

```
SHBdySrc.boundaryField().updateCoeffs();
```

will not perform any physical calculations but just write the negative fluid result into the proper opposite surface meshes of the internal field of SHBdySrc by using "addPatchValueToInternalField" again. This proceeding grants mass conservation when hydrogen is exchanged between different regions.

In summary the effective appearance of the boundary mass exchange shifted from the BCs of the equations to source term internal fields in the equations themselves. The formal environment with regard to source code compilation and embedding and the formal entry in input files still remained the same as for an OpenFOAM BC. For this reason - although this is not correct in all consequence any more - the mass transfer correlation is still named and implemented into OpenFOAM as a new BC of type named "hwtSievertsSingle".

A comparison of the first version of the Sieverts interface BC (yet a true BC) to the mass conservative version was done in a simple calculation example. The transient fluid and solid concentration results were hardly distinguishable at a first glance. However the error in the mass balance dropped by two orders of magnitude. Moreover the error did not grow any more with time or number of iterations but showed a statistical noise behavior around a constant level.

4.3 Rate dependent interface mass transfer BC for single species

The "hwtRatesDiffSingle" BC correlation provides a simple hydrogen wall transfer making use of given dissociation and recombination rate constants. If again the surface should not store mass, the mass transfer from/to the surface to the bulk is equal to the solid diffusion flux.

$$\mathrm{J} = \left(C_\mathrm{g} kT\right) K_\mathrm{d} - C_s^2 K_\mathrm{r} = \Delta_\mathrm{s}\left(C_{\mathrm{s},0} - C_\mathrm{s}\right) \qquad (4.6)$$

with:

$\mathrm{K_d}$ dissociation rate [molecules/m^2/Pa/s],

$\mathrm{K_r}$ recombination rate [m^4/s/molecules].

The presence of the diffusion term for the solid surface layer in eq. 4.6 effectively avoids unphysical accumulation of mass in the first layer of the solid mesh at high mass transfer rates.

Similarly like in section 4.1.1, a quadratic equation for $\mathrm{C_{s,0}}$ can be obtained from eq. 4.6. The solution is:

$$\mathrm{C_{s,0}} = -\frac{\Delta_\mathrm{s}}{2\mathrm{K_r}} + \sqrt{\left(\frac{\Delta_\mathrm{s}}{2\mathrm{K_r}}\right)^2 + \mathrm{C_s}\frac{\Delta_\mathrm{s}}{\mathrm{K_r}} + \mathrm{C_g\,k\,T_0}\frac{\mathrm{K_d}}{\mathrm{K_r}}} \qquad (4.7)$$

or with $\mathrm{K_s^2} K_r = K_d$:

$$\mathrm{C_{s,0}} = -\frac{\mathrm{K_S^2}\,\Delta_\mathrm{s}}{2\mathrm{K_d}} + \mathrm{K_S}\sqrt{\left(\frac{\mathrm{K_s}\Delta_\mathrm{s}}{2\mathrm{K_d}}\right)^2 + \frac{\Delta_\mathrm{s}}{\mathrm{K_d}}\mathrm{C_s} + \mathrm{k\,T_0\,C_g}} \qquad (4.8)$$

Note that eq. 4.8 looks very similar to eq. 4.4. With $\mathrm{C_{s,0}}$ the actual diffusion mass flow rate to/from the bulk can be calculated. This mass flow rate is set to equivalent source term fields on both sides of the interface like for the "hwtSievertsSingle" BC in section 4.2 above.

For $\mathrm{K_s^2} K_r = K_d$ and typical dissociation rate constants from kinetic theory $K_d = \frac{1}{\sqrt{2\Pi\,\mathrm{MkT}}}$, the results of the rate dependent BC "hwtRatesDiffSingle" approach the results of the "hwtSievertsSingle" equilibrium BC because solid diffusion is the common limiting case of both models. In the case of "hwtRatesDiffSingle" solid diffusion represents the effective kinetic limit if rates become high while for "hwtSievertsSingle" gas diffusion typically is also several orders of magnitudes faster than solid diffusion.

One may also interpret $\mathrm{K_d}$ as an adsorption rate constant and $\mathrm{K_r}$ as a desorption rate constant that may be "significantly lower" than the kinetic dissociation and recombination rates. E.g. the required dissociation rate in an example calculation to get a significant difference between both models was more than ten orders of magnitude below the kinetic theory value for by $K_d = \frac{1}{\sqrt{2\Pi\,\mathrm{MkT}}}$. This significant reduction of the rate constants will describe a slower approach however to the same "$\mathrm{C_s/C_g}$ target ratio" (implied by the Sieverts correlation eq. (4.5)) compared to the diffusion limit. This understanding of a "rate limited process" may be similarly found described on page 214 in [14] where an "adsorption controlled regime (i.e. negligible diffusion mass transfer resistance, very large diffusion and/or

solubility coefficients, small metal thickness)" is opposed the diffusion limited regime.

In summary the hwtRatesDiffSingle correlation factually includes the diffusion limit and rate limit regimes and provides a smooth transition regime, too. In practice the prescription of a set of dissociation and recombination rate constants close to kinetic theory values would have the same effect as the prescription of a different Sieverts constant that would lead to a different "equilibrium" or maybe better " C_s/C_g target ratio" between solid concentration and fluid side (square root or different power of the) partial pressure. This suggests an alternative point of view where surface effects are taken into account by an effective (measured) Sieverts constant.

4.4 Rate dependent interface mass transfer BC for two species

The "hwtRatesDiffDuo" correlation is an extension of "hwtRatesDiffSingle". It describes the wall transfer of two hydrogen species A and B including the presence of AB molecules on the gas side. Possible AB molecules are HD, HT and DT. The species specific properties data are selected from evaluating fluid and solid field names that are provided by the BC call input. Formally from this stage on the solver and BC were made internally operate with A and B only while the user has to specify which species is A and -if present- B. The species specific in-/output field names have been kept for user convenience and clarity.

A new fluid transport equation for the mixed species AB molecule is required at the solver level that formally looks like the single species transport equations shown in section 4.2.2 above. For a HD system, the required new fields are the mixed molecule concentration FcHD, the gas diffusion coefficient DFHD, a source term FHDSrc and the surface transfer specific source term FHDBdySrc. The AB combination is fixed by the call, e.g. FHDBdySrc.updateCoeffs() will call the hwtRatesDiffDuo surface mass transfer correlation for a HD system.

Like in "hwtRatesDiffSingle", again the mass transfer from/to the surface to the bulk is set equal to the solid diffusion flux J. For both species the above equation (4.6) now gets an additional term for the AB molecule. This term appears identical in both equations.

$$J_A = \Delta_{sA}\left(C_{s,0\_A} - C_{s\_A}\right) = \left(C_{g\_A}kT\right)K_{d\_A} - C_{s,0\_A}^2 K_{r\_A} + \frac{1}{2}\left(C_{g\_AB}kTK_{d\_AB} - C_{s,0\_A}C_{s,0\_B}K_{r\_AB}\right) \quad (4.9)$$

$$J_B = \Delta_{sB}\left(C_{s,0\_B} - C_{s\_B}\right) = \left(C_{g\_B}kT\right)K_{d\_B} - C_{s,0\_B}^2 K_{r\_B} + \frac{1}{2}\left(C_{g\_AB}kTK_{d\_AB} - C_{s,0\_A}C_{s,0\_B}K_{r\_AB}\right) \quad (4.10)$$

Eqs. 4.9 and 4.10 may be regarded as a system of nonlinear equations that have to be solved for $C_{s,0\_A}$ and $C_{s,0\_B}$. Alternatively $C_{s,0\_B}$ may be expressed in terms of $C_{s,0\_A}$ so that eqs. 4.9 and 4.10 may be transformed to a lengthy single fourth order polynomial:

$$C_{s,0\_A}^4 + C_{s,0\_A}^3\frac{(2a-be-abf)}{(1-bf)} + C_{s,0\_A}^2\frac{(a^2-abe+2d-bfd+b^2g)}{(1-bf)} + C_{s,0\_A}\frac{(2ad-bde)}{(1-bf)} + \frac{d^2}{(1-bf)} = 0 \quad (4.11)$$

A quartic equation of this form may be solved e.g. by a C++ solver that can be found at [15].

A test with realistic data indicated that typically the equation has four real solutions: two positive and two negative numbers of similar absolute value. The re-insertion of the numeric solutions into the original equation showed that always the lowest positive solution was the physical one. Physically interesting numerical values for $C_{s,0\_A/B}$ are typically in the order of at least 1E+10. Because zero to fourth order of $C_{s,0\_A}$ appear together in the same equation, eq. 4.11 showed a huge sensitivity on numerical accuracy. We found that re-insertion of the solver solution for $C_{s,0\_A}$ in eq. 4.9 may result in a difference of up to 50% between left and right side of eq. 4.9. Nevertheless a relative change of this solver solution for $C_{s,0\_A}$ of only 1E-5 turned out already sufficient to fix this mass balance issue. It is important to be aware that this small relative error in $C_{s,0\_A}$ will only impact the transient. The effect is obviously negligible with regard to typical accuracies of properties data like diffusion constants (where sometimes a correct order of magnitude already means good data). The accuracy of the mass balance will not suffer at all as again the mass transfer is calculated once on the fluid side only and directly transferred to the solid side. To save computation time we skip to improve the solver result e.g. by an iterative re-insert loop but just take it as it is.

Although the single species fluxes are specified, the presence of an AB molecule generates an additional degree of freedom. Therefore a splitting of fluid side source terms to isotopes $A_2/B_2/AB$ is done assuming a species equilibrium constant K=2. For more details see section 4.5 below.

The three species system or oxidized species systems are not considered so far. The above single and dual species BC's may be compiled to the same DLL. The input specifies the choice by the type of the BC for the selected patch boundary. This means single and two species BC's may be used at different fluid-solid interfaces during the same calculation.

4.5 Species Equilibration

So far the model applies species equilibration only for the net interface mass flow. Beyond that, the initial $A_2/B_2/AB$ composition will not change with time. In contrast, in TMAP [2] the presence of a surface catalyzes the equilibration of species according to rate constants from kinetic theory regardless if there is a net interface

mass transfer or not. While in TMAP the surface always interacts with the complete volume/enclosure, in OpenFOAM things get a bit more difficult as only the fluid side surface cells can formally be made to interact with the surface reasonably. On the other hand surface equilibration is a much faster process than gas diffusion. It can be that much faster because there is no net mass/species transport (i.e. the average particle speeds unlike the diffusion parameters are relevant). In principle one would need to solve something like an additional equilibration equation with a high equilibration coefficient. That equation would formally look similar to a diffusion equation. The below considerations propose and justify a practical solution of this modeling problem without the need for further differential equations.

In a HCPB BU basically two fluid regions exist: the purged pebble bed and the coolant channels.

- The pebble bed can be regarded as a porous body. The pebble surfaces represent considerable surfaces for equilibration by isotope exchange so the role of the bed walls for equilibration inside the pebble bed region is limited. Here equilibration may be well considered as a (pseudo) bulk effect. In practice one may add an equilibration step in each time step inside each mesh element so always local equilibrium between the two pure species molecules and their mixed molecule is maintained.

- The coolant flow is highly turbulent due to a coolant flow speed of tens of m/s in few mm channels at 80bars. Turbulence might be intendedly further enhanced for better heat transfer e.g. by surface structures. In this situation there is considerable fluid exchange in the flow cross section which in practice also results in a fast equilibration over the channel cross section.

The above considerations favor a bulk equilibration for all the pebble beds and coolant channels fluid regions cells against doing an equilibration only for the solid-fluid interface cells. This also would avoid a dependence of the equilibration transient from (interface) cell dimensions.

While the conditions of applying a two species equilibrium model require some discussion, the calculation itself is straightforward. For the reaction of a homonuclear, diatomic molecule

$$\frac{1}{2}A_2 + \frac{1}{2}B_2 \Leftrightarrow AB \tag{4.12}$$

the equilibrium constant is

$$K_{\mathrm{eq\_AB}} = \frac{C_{\mathrm{AB}}}{\sqrt{C_{\mathrm{A2}}}\sqrt{C_{\mathrm{B2}}}} \sim 2 \tag{4.13}$$

With $C_{\mathrm{A20}} = C_{\mathrm{A2}} - \frac{1}{2}C_{\mathrm{AB}}$ and $C_{\mathrm{B20}} = C_{\mathrm{B2}} - \frac{1}{2}C_{\mathrm{AB}}$, the equilibrium concentration of the mixed species can be evaluated to:

$$C_{\mathrm{AB}} = \frac{2\,C_{\mathrm{A20}}C_{\mathrm{B20}}}{C_{\mathrm{A20}} + C_{\mathrm{B20}}} \tag{4.14}$$

C_{A20} and C_{A20} are the total amounts of A_2 and B_2 molecules if no AB was present.

In the code species equilibration is done just before the calculation of the mass transfer source terms for this time step. Depending on the input, equilibration can either be done only in the solid-fluid interface fluid side cells or in the complete fluid region or no equilibration is done. After that the (fluid side part of) "hwtRatesDiffDuo" will calculate the interface/wall mass transfer as already described above.

Trying out interface equilibration vs. bulk equilibration showed non-negligible impact on the transient of the gas species composition. So the choice of the equilibration model might require additional attention in practice.

5 Verification calculations

Analytic solutions often exist only in 1D or 2D, while OpenFOAM usually is 3D. For OpenFOAM 1D or 2D may be simulated by using only a single mesh element in the dimension(s) to be neglected and apply "empty" BCs in the unused direction(s). This simplification was often not taken in the examples below because the application for a BU typically requires a 3D mesh, thus possible problems that appear only in a 3D mesh are relevant. Following this concept most 1D cases are modeled still using a cross section of a few cells with typically "zeroGradient" BC's.

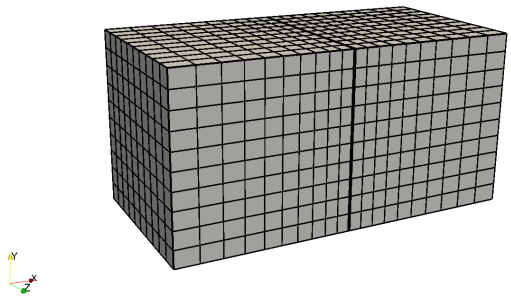

Figure 5.1: OpenFOAM mesh of two volumes linked by a thin membrane.

A common starting geometry of a square membrane that connects two cubic volumes had been selected as the simplest setup that allows testing all the key features described in this paper. Figure 5.1 illustrates the geometry. The membrane had been modeled with 5, 10 or 90 thickness cells. The fluid mesh part may use grading towards the membrane so the step of cell dimensions between solid and fluid regions is reduced somewhat. The dimensions of the example setup were either 200x100x100mm with the cross section modeled by 10x10 cells or 4x2x2mm with the cross section modeled by 2x2 cells. The membrane thickness was always 1.2mm and has to be subtracted from the left volume x-dimensions. Basically this kind of mesh was always used. For the cases when only a single solid or

fluid region investigated, interactions with neighbor regions were just switched off by "zeroGradient" BCs.

5.1 Comparison to analytic solutions

Chapter four of the classic textbook by John Crank "The Mathematics of Diffusion" treats permeation through a plane sheet [16]. As preparative calculations, first for a solid and a fluid example case the 1D diffusion problem of *"the case of diffusion through a plane sheet or membrane of thickness L and diffusion coefficient D, whose surfaces, x = 0, x = L, are maintained at constant concentrations C_1 and C_2 respectively"* (quotation from [16]) is investigated.

5.1.1 Transient diffusion in solid regions

Section 4.3.3 in [16] describes the analytic solution of a scenario where both surface concentrations C_1 and C_2 are constant. The membrane has uniform initial concentration C_0 at t=0. This scenario will end up in a linear concentration gradient along the membrane; however the transient of this scenario is still interesting. The analytic solution of the transient concentration in the membrane is:

$$C(x,t) = C_1 + (C_2 - C_1)\frac{x}{L} + \frac{2}{\pi}\sum_{n=1}^{\infty}\frac{C_2\cos n\pi - C_1}{n}\sin\frac{n\pi x}{L}\exp\left[-D\,t\left(\frac{n\pi}{L}\right)^2\right] + \frac{4C_0}{\pi}\sum_{m=0}^{\infty}\frac{1}{2m+1}\sin\frac{(2m+1)\pi x}{L}\exp\left[-D\,t\left(\frac{(2m+1)\pi}{L}\right)^2\right] \quad (5.1)$$

Calculation parameters were membrane thickness L = 1.2mm, diffusion coefficient D = 1.1190183E-8m^2/s. Initial concentrations were $C_0 = C_2 = 1.E+10/m^3$, $C_1 = 1.E+24/m^3$, i.e. a concentration step by 14 orders of magnitude at the left side surface of the membrane.

Figure 5.2 shows OpenFOAM calculations and analytic results. The expected linear steady state concentration profile is almost developed after 30s. Transient results agree close to perfect with 90 cells along the membrane thickness. Using only 10 cells along the membrane thickness results in visible deviations for the first (1s) curve. Not shown is that an increase of the time step from 0.005s to 0.05s hardly has visible impact. It was verified that results are the same for a y-z mesh cross section of 100x100mm with 10x10 cells vs. 2x2mm with 2x2 cells.

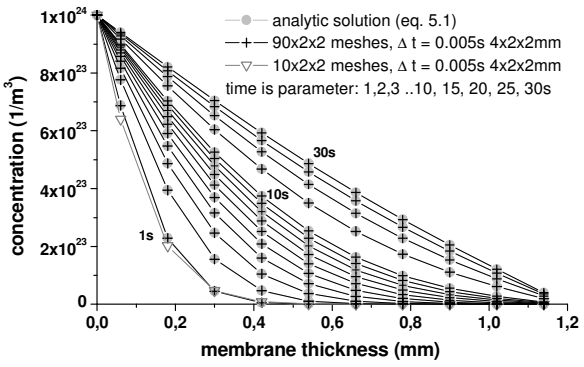

Figure 5.2: H concentration profile over membrane thickness starting from a step profile at 0s. Curve parameter is time.

5.1.2 Transient diffusion in fluid regions

Similarly now a simple scenario for a fluid region is compared to the analytic solution eq. 5.1. Mass transport is governed by Chapman-Enskog gas diffusion (eq. (2.2) in section 2.1.1, D = 5.98E-4m^2/s) while the flow velocity is kept zero. Figure 5.3 shows the concentration profile inside the right side volume "rightV". The profile develops with time from an initial step from 1E+24/m^3 to 1E+10/m^3 towards the expected linear concentration gradient for fixed value BCs. Again the agreement is very good.

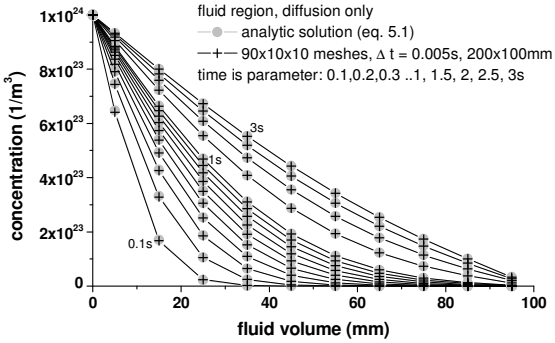

Figure 5.3: H_2 concentration profile over fluid volume "rightV" starting from a step profile at 0s. Curve parameter is time.

5.1.3 Transient permeation

The underlying experiment is two volumes separated by a membrane. The first volume named "leftV" is maintained at a constant partial pressure of 10Pa of hydrogen. "leftV" is brought in contact with the membrane at t=0. The second volume named "rightV" is in contact with the opposite membrane surface. "rightV" is closed and empty at the beginning. Permeability data is gained from the pressure rise in "rightV" over time.

The pressure rise $p_r(t)$ in "rightV" is described according to the below analytic formula which is derived from the original formula 4.24a on page 51 of [16]:

$$p_r(t) = \frac{RT\Phi\sqrt{p_l}}{2V}\frac{A\,d}{D}\left(\frac{D\,t}{d^2} - \frac{1}{6} - \frac{2}{\pi^2}\sum_{n=1}^{\infty}\frac{(-1)^n}{n^2}e^{\frac{-D\,n^2\pi^2\,t}{d^2}}\right) \quad (5.2)$$

with:
R - universal gas constant [8.314 J/mol/K]
T - constant temperature of the complete setup [773K]
V - closed volume (right side of membrane) [1e-3m^3]
Θ - permeability of membrane [2.84E-11 mol/m/s/Pa$^{0.5}$]
p_l - constant D_2 partial pressure in left volume [10 Pa]
A - area of membrane [1.E-2m^2]
t - time [s]
d - thickness of membrane [1.E-3m]
D - diffusivity of D-atoms in membrane [7.95E-9m^2/s]

The numerical data for the example calculation had been taken from [6]. They represent transport data for Deuterium in 9%-Cr RAFM steel. The corresponding OpenFOAM model uses a simplified mesh with a 4x4 cells y-z cross section. The x-direction is modeled with 10 cells for "leftV", 100 cells for the membrane and only

1 cell for "rightV". The gas diffusivity was artificially increased in both volumes by four orders of magnitude against the Chapman-Enskog data to make sure gas diffusion has no impact. Figure 5.4 illustrates the OpenFOAM mesh.

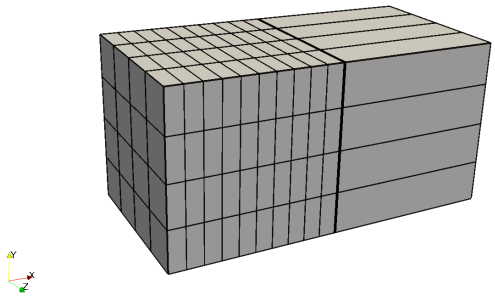

Figure 5.4: OpenFOAM simplified mesh of two volumes linked by a thin membrane for permeation simulation (details see text).

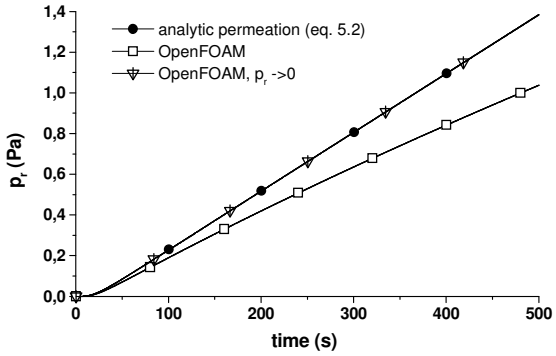

Figure 5.5: Pressure rise in right side volume "rightV". The calculated pressure rise is significantly lower than the analytic formula. If the pressure rise on the right side is neglected for evaluating the mass transfer correlation, the calculation results match the analytic formula.

Figure 5.5 shows the results. The supposed physically correct calculation shows a lower pressure rise with respect to the one predicted by the analytic solution. We suspected that this was caused by the (small) pressure rise in "rightV" which is not taken into account by the analytic solution that assumes $p_l >> p_r = 0$. For a second calculation the OpenFOAM BC between membrane and "rightV" was manipulated that mass transfer was always calculated against $p_r = 0$ disregarding the actual pressure rise in "rightV". This calculation is marked in the graph by "$p_r \rightarrow 0$". After this manipulation the result nearly perfectly fits the analytic solution. It is remarkable that clearly visible deviations already occur in figure 5.6 when the right side pressure is still orders of magnitude below the left side pressure (10Pa). In summary this calculation verifies the proper physical implementation of the new fluid diffusion equation shown in section 4.2.2 as well as the proper functionality of the Sieverts BC introduced in section 4.2.

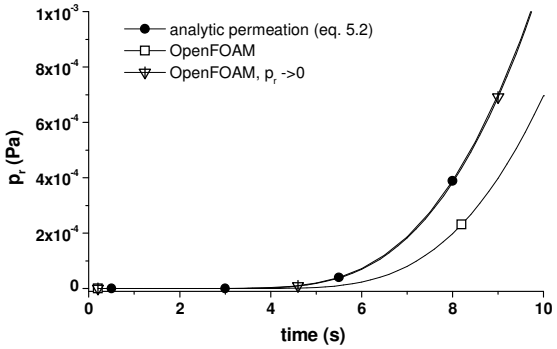

Figure 5.6: (cutout of fig. 5.5): Pressure rise in right side volume. The deviation from the analytic formula already occurs at still very low nonzero right side pressures. Left side pressure is 10Pa.

5.2 Comparison to TMAP results

The scenario descriptions and TMAP inputs of two selected TMAP calculation scenarios had been used for a recalculation exercise with OpenFOAM. These scenarios represent very simplified models of a planned permeation of deuterium through Eurofer-97 experiment setup at KIT [17].

5.2.1 TMAP permeation scenario MEMBR_SCN1

Besides different numeric parameters, this scenario does not differ significantly from the scenario in section 5.1.3. In contrast to the analytic eq. 5.2, TMAP also takes into account the rise of pressure or gas molecules concentration on the right side. In figure 5.5 this would mean that the pressure rise effects from nonzero $p_r(t)$ of the lower curve are going to be included, too.

Description of the TMAP scenario

An enclosure (#1) with constant hydrogen partial pressure is linked via a membrane of Eurofer-97 of area 1E-2m² and thickness 1.2E-3m to another enclosure (#2) with constant volume of 1E-3m^3. The setup is at uniform and constant temperature of 723K. The partial pressure of Q_2 is 1E-3Pa in each enclosure initially. The concentration of Q in the membrane is initialized with the equilibrium value according to Sieverts law. The "lawdep" BC of TMAP7 is applied. Transport parameters for H in Eurofer-97 are taken from [12].

The partial pressure of Q_2 is linearly increased from the initial value to a final value of 1Pa during the first 1s of runtime, mainly for numeric reasons. It is then kept constant for the rest of the 36000s runtime. As results, the surface flux of Q over the right side surface of the membrane and the partial pressure increase of Q_2 in enclosure #2 are observed.

OpenFOAM model

TMAP operates in 1D without a real geometry. So the shape of the membrane is not specified and the actual volume of enclosure #1 is not given as the pressure there is prescribed. In contrast OpenFOAM as a CFD code

needs a geometric mesh. The choice is a square membrane that connects two cubic volumes of 1E-3m^3, hence the mesh basically looks again like in figure 5.1. The membrane had been modeled either with 5 or with 10 thickness cells (the TMAP example used 10). The volume cells show grading towards the membrane so the step of different cell dimensions from the membrane to the volumes is reduced somewhat.

The two volumes called "leftV" and "rightV" are specified as fluid regions. Both fluid regions are in contact with the solid region "membrane" between them. The left side plane of volume "leftV" has a "fixedValue" BC that keeps the Q2 partial pressure constant at 1Pa during the transient. All other outer surfaces have "zeroGradient" BC's. The hwtSievertsSingle (section 4.2) correlation had been used at both fluid-solid interfaces. The Sieverts solubility correlation had been taken from the above TMAP input. For the pressure at 36000s the corresponding solid concentration of Q atoms in the Eurofer-97 membrane is 1.0989E+21 1/m^3. The corresponding data of OpenFOAM is 1.09806E+21 1/m^3, which means that the deviation in the Sieverts equilibrium between both models is lower than 0.08%. This result confirms that the TMAP "lawdep" correlation represents equilibrium conditions.

A characteristic of our model is that hydrogen isotopes are always regarded as impurities in the carrier gas. Although there is no flow, the fluid equations are formally solved for argon at 100000Pa and 723K. The distribution of Q_2 in the volume filled with (zero convection) carrier gas follows two species gas diffusion according to the Chapman-Enskog equation. However gas diffusion has been increased by a factor of 1E+4 to simulate the perfect mixing assumption in TMAP enclosures that contain no argon and are characterized by only one pressure (hence a TMAP enclosure would be equivalent to one cell in CFD). The enhanced gas diffusion also ensures that the Q_2 partial pressure in "leftV" remains very close to the initial value prescribed by the BC when mass transfer by permeation takes place. This represents the closest emulation of the TMAP scenario. The calculation time on a standard desk PC was 3hours.

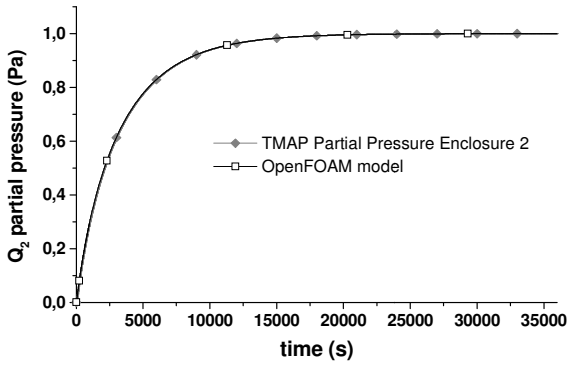

Figure 5.7: TMAP vs. OpenFOAM transient Q_2 partial pressure rise in right side volume.

Results

Figure 5.7 compares the results over the transient 0-36000s. The OpenFOAM results match the TMAP results very precisely. The maximum difference between both calculated partial pressure curves is 1.2E-4Pa.

5.2.2 TMAP permeation scenario PETE-MDL2

This TMAP scenario also simulates a purge gas flow.

Description of the TMAP scenario

Figure 5.8 illustrates the TMAP model. A D_2 partial pressure of 1E-4Pa is ramped to 1.4Pa within 2 seconds in the boundary enclosure BE#1. The left side of diffusion segment DS#1 (the permeation membrane) is linked to BE#1 by Sieverts law. The diffusion segment DS#1 consists of 10 elements, with the effective diffusion coefficient of D_2 at 673K. Diffusion and Sieverts numerical parameters for D in Optifer IVb had been taken from [18]. The right side of DS#1 is linked to the functional enclosure FE#2 again by Sieverts law. FE#2 receives a specified sweep gas volumetric flow rate (1.189e-6 m³/s) from the boundary enclosure BE#3 (where a constant D_2 partial pressure of 1E-4 Pa is set) and outputs the same volumetric flow rate to BE#4.

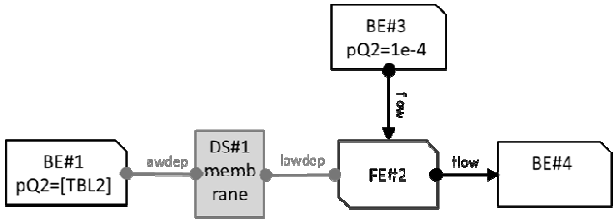

Figure 5.8: Topology of TMAP model PETE-MDL2 taken from [17].

In this case, TMAP calculates the time dependent secondary side D_2 partial pressure from the purge gas flow and the permeate flux of D into FE#2.

OpenFOAM model

The very simple OpenFOAM setup tried to follow the TMAP model as close as possible. Volume "leftV" reduced to a single cell square y-z cross section that is equivalent to the TMAP BE#1. The membrane was modeled with 10 (thickness) cells as in TMAP. Volume "rightV" is a single cell that is equivalent to the TMAP FE#2. As TMAP is 1D, the inlet side for the purge flow has to be the membrane and the constant purge flow velocity through "rightV" is given by $v_{rightV} = \frac{\text{purge flow rate}}{\text{membrane area}}$. At the inlet side a fixed value BC for the D_2 concentration (FcD2 in the above terminology) corresponding to 1E-4 Pa was prescribed. At the outlet side a zero gradient BC was set for the concentration.

Results

Figure 5.9 compares the results for the D_2 partial pressure at the outlet over the transient 0-400s. OpenFOAM calculates a pressure rise up to only about

2.2E-4Pa compared to 0.187Pa for TMAP. A log scale y-axis was selected for a reasonable presentation of both datasets in the same graph. The origin of this strong deviation is that TMAP does not include diffusive transport in volumes. If gas diffusion is taken out of the OpenFOAM transport equation in "rightV", a nearly perfect agreement is achieved (all data differ by less than 1%). This indicates again the high sensitivity of permeation on very low partial pressures p_r which was already discussed in section 5.1.3.

Again the conclusion is that the results are very similar as soon as the OpenFOAM model precisely follows the 1D TMAP formulation.

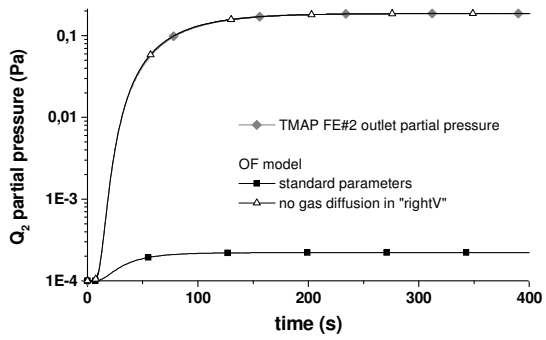

Figure 5.9: TMAP vs. OpenFOAM transient D_2 partial pressure rise in outlet of FE#2/"rightV".

6 Conclusion

The work presented in this paper describes the development steps for a OpenFOAM based numerical simulation model that describes transient tritium transport on the BU level for the EU HCPB concept for DEMO. The focus of this article is on mass transfer at a solid-fluid interface. Mass conservative correlations based on the modified formalism of OpenFOAM BCs had been developed. They allow for solid/fluid interface mass transfer modeling in the diffusion limit case, in the surface limit case as well as in the transition regime between them. These model capabilities are also made available for a two species system and include equilibration options for a mixed species fluid molecule.

For the single species equilibrium correlation verification calculations had been performed against (1D) analytic solutions and TMAP calculation examples. The agreement with the analytic solutions and the TMAP simulations was close to perfect. From these findings we conclude that the implemented physics are basically correct. Further our calculations already highlighted significant sensitivities on some effects that are sometimes neglected in analytic solutions or 1D numerics. Verification calculations in more complex geometries or for mass transfer including mixed species molecules are still outstanding topics for future work.

Another outstanding issue is a model description of trapping. Further an interface to common CFD codes is desirable to make designers calculation results available as background scenario for mass transfer analysis. Although one could also perform a thermal and flow analysis of the BU with OpenFOAM, an import concept promises avoiding duplicate work and e.g. for safety relevant calculations possible quality management questions are better answered by the suppliers of the original fields data. Candidates for import are data fields for flow, temperature and neutronic radiation. From a first view, data converters to OpenFOAM seem available for FLUENT and CGNS but not for CFX. A converter from MCNP data to common CFD formats (FLUENT, CFX, CGNS) is available [19, 20].

The tritium source term will be implemented into the model in three steps. In a first step, local neutronic tritium generation (e.g. imported from MCNP calculations at KIT-INR) is intended to be used directly as the tritium source term field. In a second step, a residence time model for the pebble beds should be implemented. In the third model step that would include predictive capabilities the modeling of tritium production, retention and release to the purge gas would have to be done in detail on the pebble level. Such a pebble level model would require the functionality and the capabilities of some dedicated tool, such as the MISTRAL code [21].

This report provides a condensed overview on our work. Readers with interest in more details and background information that have access to EUROFUSION IDM may refer to [22, 23].

Acknowledgments

The authors thank Sergey Khashin from Ivanovo State University, Ivanovo, Russia, for kindly approving the usage of his quartic equation solver in this project.

This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 under grant agreement No 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

References

[1] Italo Ricapito, P. Calderoni, Yves Poitevin, Luis Sedano, Tritium transport modeling for breeding blanket: State of the art and strategy for future development in the EU fusion program, Fusion Engineering and Design 87 (2012), p. 793–797.

[2] G. R. Longhurst, J. Ambrosek, Verification and validation of the tritium transport code TMAP7, Fusion Sci. Technol. 48, (July 2005), p. 468–471.

[3] R. Meyder, L. V. Boccaccini, N. Bekris, Tritium analysis for the European HCPB TBM in ITER, proceedings of SOFE 2005, Knoxville, Tennessee (CD-ROM proceedings, also unofficially available at http://www.telegrid.enea.it/Conferenze/SOFE05/DATA/05_09.PDF

[4] Hongjie Zhang, Alice Ying, Mohamed A. Abdou,

Integrated simulation of tritium permeation in solid breeder blankets Fusion Engineering and Design 85 (2010), p. 1711–1715, doi:10.1016/j.fusengdes.2010.05.018

[5] Alice Ying, Hongjie Zhang, Brad J. Merrill, Mu-Young Ahn, Advancement in tritium transport simulations for solid breeding blanket, Fusion Engineering and Design 109–111 (2016), p. 1511–1516, http://dx.doi.org/10.1016/j.fusengdes.2015.11.040

[6] Axel von der Weth, Frederik Arbeiter, Dmitry Klimenko, Volker Pasler, Georg Schlindwein, Review of Hydrogen Isotopes Transport Parameter, Fusion Engineering and Design, In Press, 2017, https://doi.org/10.1016/j.fusengdes.2017.04.024

and/or for readers with EUROfusion IDM access:

Axel von der Weth, Frederik Arbeiter, Volker Pasler, Dmitry Klimenko, Georg Schlindwein, Technical note on Hydrogen permeation parameters in RAFM steel (Recommendations for analyses for HCPB), Eurofusion IDM: EFDA_D_2MYS5C, March 2016.

[7] F. Hernandez, P. Pereslavtsev, Q. Kang, P. Norajitra, B. Kiss et al., A new HCPB breeding blanket for the EU DEMO: Evolution, rationale and preliminary performances, Fusion Engineering and Design, Feb 2017, https://doi.org/10.1016/j.fusengdes.2017.02.008

[8] R.B. Bird, W.E. Stewart and E.N. Lightfoot. Transport Phenomena, Wiley, New York, London (1960), p. 511.

[9] Helge Petersen, The Properties of Helium: Density, Specific Heats Viscosity and Thermal Conductivity at Pressures from 1 to 100 bar and from Room Temperature to about 1800K, Danish Atomic Energy Commission, Research Establishment Riso, Riso Report No. 224, (September 1970).

[10] T.N. Bell, R. Shankland and P.J. Dunlop, The pressure dependence of the mutual diffusion coefficients of binary mixtures of helium with ten other gases at 3110 K: tests of Thorne's equation, Chemical Physics Letters, Vol. 45/3, pp. 445-448, (1977).

[11] Patricia Castano-Rivera, Viviana P. Ramunni, Pablo Bruzzoni, Numerical Study of Hydrogen Trapping: Application to an API 5L X60 Steel, International Scholarly Research Network ISRN Materials Science Volume 2012, Article ID 945235, 14 pages doi:10.5402/2012/945235

[12] G.A. Esteban, A. Peña, F. Legarda, R. Lindau, Hydrogen transport and trapping in ODS-EUROFER, Fusion Engineering and Design, 82 (2007), p. 2634–2640.

[13] http://www.openfoam.org/ see there "Resources": USER Guide and C++ Source Guide.

[14] Silvano Tosti, Nicolas Ghirelli, Tritium in Fusion - Production, Uses and Environmental Impact, Nova Science Publishers, March 2013.

[15] http://math.ivanovo.ac.ru/dalgebra/Khashin/poly/index.html.

[16] John Crank, The Mathematics of Diffusion, Brunel University Uxbridge, Oxford University Press 1975.

[17] Dmitry Klimenko, Frederik Arbeiter, Kevin Zinn, Volker Pasler, Georg Schlindwein, Axel von der Weth, Interim Report (2016) on Deliverable Code validation experiments (tritium). EUROFUSION deliverable SAE-2.8.2-T02-D02, EFDA_D_2N6Z48, Jan 2017.

[18] G.A. Esteban, A. Perujo, K. Douglas, L.A. Sedano Tritium diffusive transport parameters and trapping effects in the reduced activating martensitic steel OPTIFER-IVb, Journal of Nuclear Materials 281 (2000) 34-41. Table 3 provides the data used here.

[19] Yuefeng Qiu, Peng Lu, Ulrich Fischer, Pavel Pereslavtsev, Szabolcs Kecskes, A generic data translation scheme for the coupling of high-fidelity fusion neutronics and CFD calculations, Fusion Engineering and Design 89, 1330–1335, 2014.

[20] Yuefeng Qiu, Lei Lu, Ulrich Fischer, Integrated approach for fusion multi-physics coupled analyses based on hybrid CAD and mesh geometries, Fusion Engineering and Design 96–97, 159–164, 2015.

[21] Gianfranco Federici, A.R. Raffray, M.A. Abdou, MISTRAL: a comprehensive model for tritium transport in lithium-base ceramics, Part I: Theory and description of model capabilities, Journal of Nuclear Materials 173(1990) 185-213. Part II: Comparison of model predictions with experimental results, Journal of Nuclear Materials 173 (1990), p. 214-228.

[22] Volker Pasler, Axel von der Weth, Dmitry Klimenko, Frederik Arbeiter, Intermediate Report on Deliverable Stepwise integration of physics models into the framework, Eurofusion deliverable BB-6.2.2-T002-D001, IDM: EFDA_D_2NBXLB, Dec. 2016.

[23] Volker Pasler, Axel von der Weth, Dmitry Klimenko, Frederik Arbeiter, Development of Tritium Transport Models at BU level for HCPB, Eurofusion deliverable BB-6.2.2-T001-D001, IDM: EFDA_D_2MJR2R, Dec. 2015.