Delft University of Technology

Bachelor Thesis

Testing a Nuclear Pebble-Bed Reactor Model in OpenFOAM

Author:
Tom Bakx

Supervisors:
Dr. Danny Lathouwers
Gert Jan Auwerda

August 22, 2011

Abstract

This research investigates whether the CFD code OpenFOAM can be used to simulate the mass and heat transfer through a nuclear pebble-bed reactor. OpenFOAM is a finite volume, open-source CFD program, which has the advantage that the code can be changed to suit the user. This research focusses on the Pebble-Bed Modular Reactor (PBMR-400), which is a nuclear reactor design based on the pebble-bed type, of the generation IV initiative, with as major advantages passive safety and online refuelling. In this reactor the fuel is contained in pebbles, which form a randomly packed bed through which helium coolant flows. Many other CFD codes have simulated the mass and heat flow of the PBMR, and serve as a computational benchmark for this reactor.

To verify the OpenFOAM code, four test cases simulating mass and heat flow have been investigated, as well as a simulation of the mass and heat flow through the PBMR. The one-dimensional pressure drop case showed that discrete steps in porosity cause local fluctuations in the pressure and velocity field, which can lead to errors on coarse meshes, but should not influence the final results of a pebble-bed reactor, because these reactors are large compared to the local fluctuations. The two-dimensional laminar flow case showed the code is capable of solving Poiseuille flow. The Achenbach pressure drop case showed that the KTA relation would describe Achenbach's measurements better than the code, which uses Ergun's relation, and predicted a pressure difference of 70 % for Reynolds numbers of a PBMR, around $4 * 10^4$. The analytical heat transfer case was solved correctly.

The PBMR computation showed a good agreement with the axial coolant and pebble temperature profiles, but differed by 20 K, because the benchmark did not account for absorption of energy due to expansion in the analysis of conservation of energy. The calculations showed that this term should not be ignored in conservation of energy. The boundary conditions for heat flux through the walls could not be computed with the code, so artificial boundary conditions were used. The computation also used a different Nusselt relation for the heat transfer from the pebbles to the fluid, and used a simplified pebble-pebble heat transfer relation. This caused an error in the radial temperature profiles of the coolant and pebbles. The computed pressure drop over the pebble-bed was 75 % higher than the benchmark suggested, which was explained by the use of different pressure drop relations,

as predicted by the Achenbach test case. The OpenFOAM code is thus capable of solving the mass and heat flow through a nuclear pebble-bed reactor, especially if Ergun's relation is replaced by the KTA relation.

Contents

1 Introduction **5**
- 1.1 The Very High Temperature Reactor . . . 5
- 1.2 Previous Research . . . 6
- 1.3 Aim of this Thesis . . . 7
- 1.4 Thesis Outline . . . 7

2 Governing Equations **9**
- 2.1 Conservation of Momentum and Mass . . . 9
- 2.2 Conservation of Energy . . . 10
- 2.3 Pebble Temperature Equation . . . 11

3 Solving the Equations with OpenFOAM **13**
- 3.1 Introduction to CFD . . . 13
  - 3.1.1 Finite Difference Method . . . 14
  - 3.1.2 Finite Volume Method . . . 14
  - 3.1.3 Finite Element Method . . . 14
- 3.2 A Quick Tour in OpenFOAM . . . 15
- 3.3 Discretization Schemes . . . 15
  - 3.3.1 Interpolation . . . 15
  - 3.3.2 Gradient . . . 16
  - 3.3.3 Divergence . . . 16
  - 3.3.4 Laplacian . . . 16
  - 3.3.5 Source Term . . . 17
  - 3.3.6 Under-Relaxation . . . 17
- 3.4 The Solver: my_pebbleBedFoam . . . 17
  - 3.4.1 UEqn.H . . . 18
  - 3.4.2 hEqn.H . . . 18
  - 3.4.3 pEqn.H . . . 19
  - 3.4.4 TpebEqn.H . . . 20

4 Test Cases **21**
4.1 One-Dimensional Pressure Drop 21
4.1.1 Set up, Measurement & Expectations 21
4.1.2 Results 22
4.2 Two-Dimensional Laminar Flow Between Parallel Plates . . . 24
4.2.1 Set Up, Measurements & Expectations 24
4.2.2 Results 24
4.3 Achenbach's Experiment 26
4.3.1 Set Up, Measurements & Expectations 26
4.3.2 Results 27
4.4 Analytical Heat Transfer 29
4.4.1 Set Up, Measurements & Expectations 29
4.4.2 Results 30

5 Pebble-Bed Modular Reactor **31**
5.1 Set Up, Measurement & Expectations 31
5.2 Results 32
5.2.1 Helium Temperature 32
5.2.2 Pebble Temperature 35
5.2.3 Pressure Drop 36

6 Conclusions **37**
6.1 Experimental Conclusions 37
6.2 Future Research 38

Chapter 1

Introduction

After the Chernobyl disaster, the world's point of view on nuclear energy changed radically. Nuclear energy was not an option any more and since then almost no nuclear power plants have been built. However, with the greenhouse effect becoming ever more visible and the heavy oil dependence of the economy, alternatives need to be sought for fossil fuel. Sustainable energy is one of the alternatives, but is still not a practical energy source, and nuclear energy has become the most viable option for the world's energy supply. New nuclear reactors have been developed, under the Generation IV initiative, that have found ways to circumvent former drawbacks. One of these new nuclear reactor types is the pebble-bed reactor, discussed in this thesis.

1.1 The Very High Temperature Reactor

Just after the second world war the first ideas of a pebble-bed reactor arose, and several years later the idea of a nuclear powered pebble bed took shape. This idea was conceptually different from normal nuclear reactors, because of the combination of fuel, containment, structure and neutron moderator. The pebbles used in this concept are also highly temperature resistant, resisting up to 1600 Kelvin. This temperature is much higher than the highest registered temperature in nuclear reactors during normal operation and even during reactor accident scenarios. Also, this makes the pebble-bed reactor inherently safe, because during an accident, active cooling is never necessary. This type of reactor allows for online refuelling, which means the reactor does not require a shut-down any more. The pebbles pass through the pebble bed and at the exit the status of the pebbles is checked. The pebbles can pass the reactor many times, which ensures a better usage of fuel, since frequently

passing through the reactor takes care of the space-dependance of uranium usage [12].

In 1966 the Arbeitsgemeinschaft Versuchreaktor (AVR) was built, being the first pebble-bed research reactor. It had a capacity of 15 MW and a power density thirty times smaller than a light water reactor, for safety considerations. After 21 years of successful research the AVR was decommissioned in 1988, due to the recent disaster at Chernobyl and some operational problems. After inspection of the bed and removal of most of the pebbles, the bottom reflector appeared to be broken, which made the AVR the most heavily beta-contaminated nuclear installation worldwide [16].

The Thorium High-Temperature nuclear Reactor (THTR-300) began operation in 1983 and was designed to breed usable uranium from thorium, which is plentiful in the earth's surface, during service. The reactor was put out of commission rather quickly due to poor design, that caused accidents releasing small radioactive quantities into the environment, and financial problems [22].

The High Temperature Reactor (HTR-10) began operation in the year 2003. It was designed as a prototype for a commercial scale 250 MW modular reactor. Current research is focussed on two pebble-bed reactors. One type is situated in China, the High Temperature Reactor-Pebblebed Modules (HTR-PM), designed to produce 250 MW of thermal energy per module. Construction started in 2009 and completion is expected in 2013. The other type is located in South-Africa, the Pebble-bed Modular Reactor (PBMR-400), of which a schematic view is given in figure 1.1. The design started in 1999, but due to financial problems, the design process has slowed down. The PBMR-400 is only in its design phase, but a lot of computational work has been done on this reactor type, because it has been made a computational benchmark [7]. Steady-state and transient situations regarding the thermo-hydrolics and neutronics have been computed and serve as a benchmark for CFD programs focussed on pebble-beds.

1.2 Previous Research

Computational research on pebble-bed reactors is necessary because physical experiments with pebble-bed reactors cannot provide a full description of the pebble-bed, without altering the set-up. Without a full description of a pebble-bed reactor, it cannot be used safely.

Determining the flow around all of the pebbles inside the reactor core, in the order of 100.000 to several million pebbles, is a task far out of the current computational reach. Some CFD calculations focus on small parts of

the reactor, determining the flow around several pebbles and using this information to extrapolate behavior in the total pebble-bed [15]. This method gives precise information of local heat gradients, but fails to give a good macroscopic view of the reactor. Other methods depend on using a two-phase porous medium approach, describing the porous medium only by the porosity [12]. This makes it possible to get a good macroscopic view of the behavior inside the reactor, such as wall channeling and non-uniform power density, but microscopic effects cannot be calculated [13].

However, computations on pebble-bed reactors have yielded different results than the actual testreactors for low temperature pebble-bed reactors [16]. A better understanding of the CFD of pebble-bed reactors is therefore needed.

1.3 Aim of this Thesis

This thesis focusses on an open-source CFD program called OpenFOAM. A special solver has been developed within OpenFOAM, to solve equations governing the mass and heat transfer inside pebble-beds, describing the pebble-bed as a two-phase steady-state porous medium.

The objective of this research is to determine if the solver is capable of solving heat and mass transfer inside a pebble-bed reactor correctly. This is done with several testcases, three analytical cases and one physical experiment, based on the experiment of Achenbach. Finally the total code is tested on the PBMR computational benchmark, to examine if the code calculates what the benchmark suggests.

1.4 Thesis Outline

This thesis will first cover the equations determining the behavior of the pebble-bed. Then the implementation of these equations in the solver is discussed, along with information of CFD in general and specifically about OpenFOAM. Three analytical test cases are examined, regarding the one-dimensional pressure drop, Poiseuille flow and one-dimensional heat transfer. An experiment, done by Achenbach [1], is computed and the PBMR is simulated [7]. Afterwards a conclusion and recommendation for future research will be given.

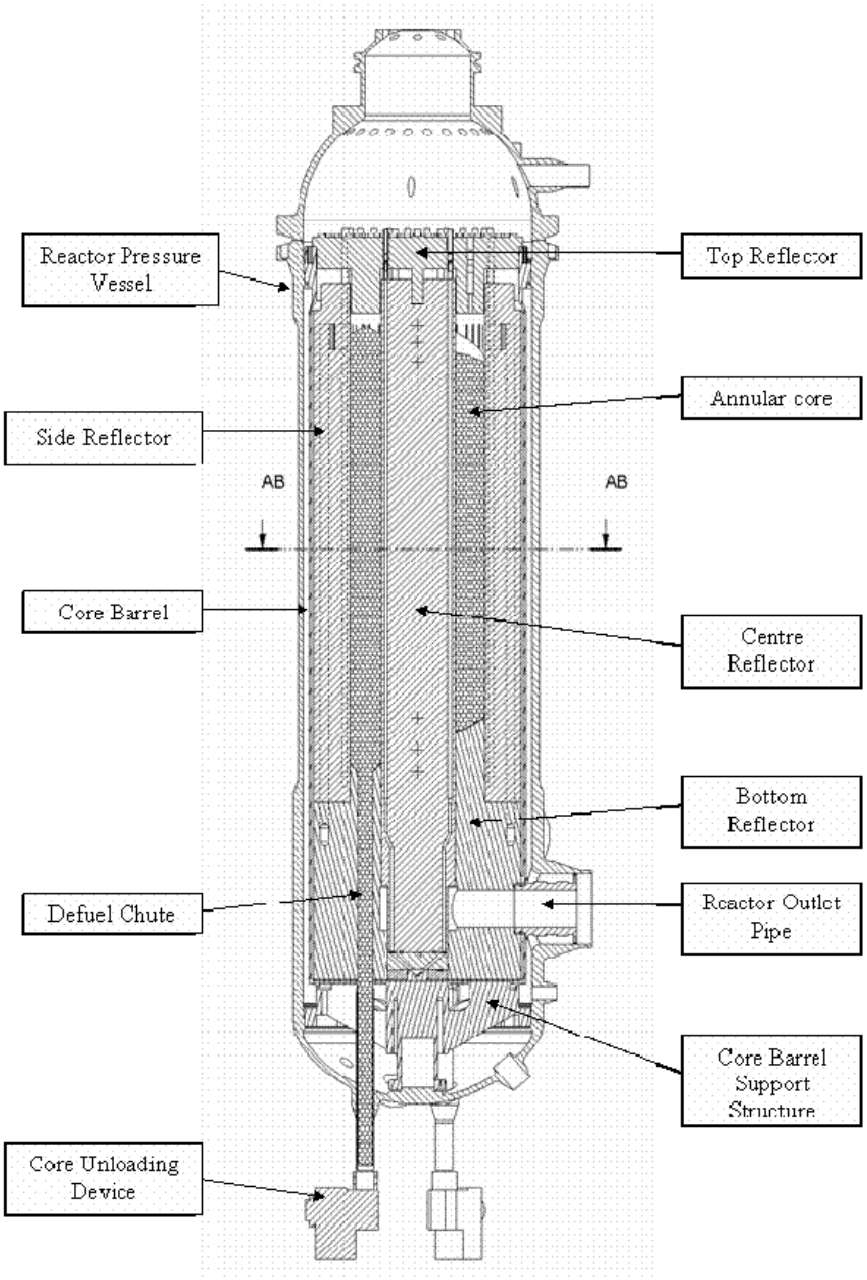

Figure 1.1: Representation of the PBMR-400.[7]

Chapter 2

Governing Equations

The pebble-bed solver is based on a model consisting of a set of equations, which are detailed in this chapter. The equations are based on conservation of momentum, energy and mass.

2.1 Conservation of Momentum and Mass

The velocity field is governed by conservation of momentum, in its compressible form [6].

$$\underbrace{\phi(\nabla \mathbf{u})}_{\text{c.a.}} + \underbrace{F_{por}\mathbf{u}}_{\text{p.d.}} + \underbrace{\epsilon\mathbf{T}}_{\text{s.s.t.}} = \underbrace{-\epsilon\nabla p}_{\text{p.g.}} \tag{2.1}$$

The first part of this equation is the convective acceleration term, using the mass flux $\phi = \epsilon\rho|\mathbf{u}|$, where ϵ is the porosity, ρ the density and $\mathbf{u}$ the velocity (c.a.). The porous drag is a function of F_{por}, which covers the resistance due to porous media (p.d.). The shear stress term is a function of the deviatoric stress tensor, $\mathbf{T}$ (s.s.t.). Finally the pressure gradient depends on the fluid pressure, p (p.g.).

The porous drag term depends on the porous drag force, F_{por}. The relation of the F_{por} is a combination of two parts, the first part is the viscous flow (Carman and Kozeny) and the last part is the turbulent flow (Burke and Plummer), and was combined by Ergun in 1952, see equation 2.2 [2] [11].

$$F_{por} = \frac{150\mu(1-\epsilon)^2}{\epsilon^2 d_{peb}^2} + \frac{1.75(1-\epsilon)\rho|\mathbf{u}|}{\epsilon^2 d_{peb}} \tag{2.2}$$

Investigation by Achenbach and the German nuclear service, KTA, resulted

in another relation, see equation 2.3 [1].

$$F_{por} = \frac{160\mu(1-\epsilon)^3}{\epsilon d_{peb}^2} + \frac{3\mu^{0.1}(1-\epsilon)^{2.1}\rho^{0.9}|\mathbf{u}^{0.9}|}{\epsilon d_{peb}^{1.1}} \tag{2.3}$$

Both equations for the porous drag force show, respectively, a viscous and a turbulent flow part. The code used in this thesis uses Ergun's equation, eq. 2.2.

Conservation of mass is used to correct the pressure and velocity fields, eq. 2.4, which will be explained in section 3.4.3.

$$\nabla \cdot \phi = \nabla \cdot (\epsilon\rho\mathbf{u}) = 0 \tag{2.4}$$

2.2 Conservation of Energy

The temperature of the coolant is considered by conservation of energy, by applying this on the enthalpy field of the coolant and using $h = c_p T$, where c_p is the specific heat.

$$\underbrace{\phi \cdot (\nabla h)}_{\text{c.e.}} - \underbrace{\nabla \cdot (\epsilon\alpha_{eff}\nabla h)}_{\text{d.e.}} = \underbrace{htc(T_{peb} - T_{He})}_{\text{h.t.}} + \underbrace{\frac{\phi}{\rho} \cdot \nabla p}_{\text{p.g.}} \tag{2.5}$$

The first term considers the convection of enthalpy, using mass flux ϕ and enthalpy h (c.e.), followed by the diffusion of enthalpy, using the porosity ϵ, the thermal diffusivity α and the enthalpy, h (d.e.). The heat transfer term uses the heat transfer coefficient, htc, and the difference between the pebble temperature T_{peb} and coolant temperature T_{He} (h.t.). The pressure gradient uses mass flux, ϕ, density ρ and pressure gradient, p (p.g.) [6].

Empirical formulas are used to calculate the fluid properties in the code, see equations 2.6 to 2.10

$$htc = \frac{\alpha c_p Nu(1-\epsilon)}{6d_{peb}^2} \tag{2.6}$$

$$\alpha_{eff} = \frac{0.11\rho d_{peb}(\epsilon - 1)|\mathbf{u}|}{(0.39 - 1)} \tag{2.7}$$

$$Nu = (1 + 1.5(1-\epsilon)) \left(2 + \sqrt{(0.664\sqrt{Re}Pr^{1/3})^2 + \frac{(0.037Re^{0.9}Pr)^2}{Re^{0.1} + 2.443(Pr^{2/3} - 1)}}\right) \tag{2.8}$$

$$Re = \frac{d_{peb}\rho|\mathbf{u}|}{\mu} \tag{2.9}$$

$$Pr = \frac{\mu c_p}{k} \tag{2.10}$$

The heat transfer coefficient, htc, determines the heat exchange between the pebbles and the fluid and is modeled according to the relation given by Gnielinski [17]. α_{eff} gives the effective diffusion of enthalpy and uses the relation from Yagi and Kunii [18]. The Nusselt number Nu refers to the ratio between convective and conductive heat transfer, which is necessary for the heat transfer coefficient, where d_{peb} is the pebble diameter, μ is the kinematic viscosity, and k is the thermal conductivity and is calculated with the equation from Gnielinski [17]. Pr and Re are the Prandtl and Reynolds numbers, respectively.

2.3 Pebble Temperature Equation

To calculate the temperature of the pebbles the following equation is used.

$$\underbrace{-\nabla \cdot (k_{peb} \nabla T_{peb})}_{\text{d.t.}} + \underbrace{htc(T_{peb} - T_{He})}_{\text{h.t.}} = \underbrace{Q}_{\text{source}} \tag{2.11}$$

The left side contains the diffusion of heat between the pebbles and uses the pebble-bed conductivity k_{peb}, which includes the heat transfer due to radiation between the pebbles, and the internal heat transfer of the pebbles (d.t). The pebble-pebble heat transfer through contact points is not included as it is considered negligible compared to the high radiative heat transfer at high pebble temperatures in the pebble-bed. The next term calculates the heat transfer from the pebbles to the coolant (h.t.). The right side of the equation contains the source term, Q, which is the power generated by fission inside the pebbles in $\mathrm{W/m^3}$. The porosity may approach 1 in near-wall situations, which makes the thermal conductivity approach infinity. For these situations special near-wall formulas are used. The thermal conductivity of graphite inside the pebble-bed is given by Zehner and Schluender [20], and the thermal conductivity of graphite near the walls is given by Tsotsas and Martin [21].

$$k_{graphite}^{within} = 4\sigma T^3 d_{peb} \left(\left(1 - \sqrt{(1-\epsilon)}\right) + \frac{\sqrt{(1-\epsilon)}}{\frac{2}{\epsilon_r} - 1} \left(\frac{B_z + 1}{B_z}\right) \left[1 + \frac{1}{\left(\frac{2}{\epsilon_r} - 1\right)\Lambda}\right]^{-1} \right) \tag{2.12}$$

$$k_{graphite}^{near-wall} = \frac{4\sigma T^3 d_{peb}}{\left(\frac{2}{\epsilon_r} - 1\right)} \left(1 - \sqrt{(1-\epsilon)}\right) + \sqrt{(1-\epsilon)} \left[\frac{1}{\frac{4\sigma T^3 d_{peb}}{\left(\frac{2}{\epsilon_r} - 1\right)}} + \frac{1}{p_g}\right]^{-1} \tag{2.13}$$

Where σ is the Stefan-Boltzmann constant, ϵ_r is the emissivity of the pebbles, which is 0.8 and B_z, Λ, and p_g are given in equations 2.14, 2.15, and 2.16.

$$B_z = 1.25\left(\frac{1-\epsilon}{\epsilon}\right)^{10/9} \tag{2.14}$$

$$\Lambda = \frac{p_g}{4\sigma T^3 d_{peb}} \tag{2.15}$$

$$\begin{aligned} p_g = {} & 108.901 - 0.188285T + 2.79606 \cdot 10^{-4}T \\ & -2.18888 \cdot 10^{-7}T^3 + 6.6 \cdot 10^{-11}T^4 \end{aligned} \tag{2.16}$$

Chapter 3

Solving the Equations with OpenFOAM

The equations governing the physics of pebble-beds discussed in chapter 2 do not have an analytical solution - except in extremely simplified cases - and need to be solved computationally. In our case this has been done with the aid of the open source software package OpenFOAM. The implementation of the equations in OpenFOAM and the methods used to solve them numerically are discribed below.

First an introduction is given on the various methods generally used to solve computational fluid dynamics (CFD) problems. The next section describes the OpenFOAM software package, followed by an explanation of the schemes used to discretize the various equations of chapter 2. The last section details the implementation of the equations in the my_pebbleBedFoam solver and how this system of equations is solved.

3.1 Introduction to CFD

The equations defining the behavior of the pebble-bed, and generally the behavior of fluid mechanics inside the given geometry, often consist of differential equations, which demand special ways of approaching. For approximating these problems and geometries, a certain discretization method needs to be chosen. Over the years, several discretization methods have been invented, all having their strong and weak points, of which the most important are:

- Finite Difference Method
- Finite Volume Method

- Finite Element Method

These discretization methods rely on a mesh to be able to solve. A mesh is a collection of points, at a given distance from each other, covering the entire object of interest. This mesh does not need to be uniformly distributed or orthogonal, but non-uniformity and non-orthogonality need to be taken into consideration at the discretization schemes. For greater accuracy and detail the grid can be made finer at places of interest, see [4].

3.1.1 Finite Difference Method

This is the oldest method of solving partial differential equations, discovered by Euler in the 18th century, and is based on approximations of the partial derivatives between nodal points within the mesh. This method starts off with the conservation laws in a differential form. The mesh with this method is mainly based in structured grids, which are also used as coordinate lines. The downside of this method is that the conservation laws are not enforced, unless special care is taken.

3.1.2 Finite Volume Method

This method devides the mesh into small volumes, with in its center a node. The conservation laws are stated in an integral form using single volume cells and use *Gauss' Theorem* to switch from volume to surface integrals. This method requires interpolation to determine the values at the edges of the control volume. The advantage of this method is that the surface integrals at both sides of the boundary need to be the same, which makes it easy to enforce the conservation laws. The disadvantage is the fact that this code relies on three calculation methods, namely integration, interpolation and differentiation, making the code more difficult. OpenFOAM, the CFD program used in this thesis, is based on the FV method.

3.1.3 Finite Element Method

This method is similar to the Finite Volume Method, because the domain is spatially discretisized and considered with the use of volume integrals, but depends on a piecewise function between all nodal points that satisfies the differential equation. This gives a function defined everywhere in the domain, and not only on the nodes. The advantage of this method is the fact that it can be used on any grid, regardless of geometry. A drawback is that for simpler grids it is more difficult to find efficient solution methods.

3.2 A Quick Tour in OpenFOAM

OpenFOAM is a *C++* based CFD code, which relies on the Finite Volume Method. One of the greatest advantages of OpenFOAM is that it is open-sourced. This means everybody can change the code to the way he/she prefers. Much configuration has to be done by the user, but this gives a lot of freedom compared to commercial packages [5].

The Basic Format

OpenFOAM is run on a UNIX cluster. Two directories are of importance, namely the case directory and the solver directory. The case directory contains the situation dependant information, such as the boundary and initial conditions, the mesh and discretization methods. The solver directory contains the files in which the equations governing the case and their solution steps are defined. The files in the solver are converted into an executable which can be run in the case directory. The output can either be viewed with a text editor or with a visualization program.

3.3 Discretization Schemes

In CFD the problems need to be discretized, because computational methods rely on solving discrete quantities instead of continuous functions. Space, time, and the equations need to be adapted to discrete parts. Time discretization will not be considered in this thesis, because only steady-state situations are evaluated. Space discretization will be done with the use of a mesh. This splits the domain into a set of cells that fill and bound it. Before the equations can be solved on the mesh, they need to be discretized. The operators are also discretized. This section will explain the operators used in the code in OpenFOAM [4] [13] [23].

3.3.1 Interpolation

Interpolation is necessary for FV solvers because values are stored at the centers of the volumes, but are sometimes needed on the faces of the volume. Two different interpolation schemes are implemented in OpenFOAM:

- Central Differencing: a weighted mean between two nodes
- Upwind Differencing: uses the value of the node upstream.

Upwind differencing is a faster method and has inherent stability, yet has a smaller accuracy than central differencing. Central differencing is best used on course meshes, but can give instability in non-orthogonal meshes.

3.3.2 Gradient

Just as interpolation, the gradient, $\nabla\phi$, can be calculated with two different methods. The first method considers *Gauss' theorem* and examines the gradient with the use of volume and surface integration.

$$\int_V \nabla\phi dV = \int_S \phi d\mathbf{S} \approx \sum_f \mathbf{S}_f\phi_f \tag{3.1}$$

The V and S boundaries of integration are the volume and surface domains of the cell, f is the index of the cell faces. The sum over f is the sum over all the cell faces, with ϕ_f the variable of interest on the cell face, times the surface face vector, $\mathbf{S}_f$.

The second method divides the difference between two nodes, by the distance between the two nodes.

$$\frac{\phi_N - \phi_P}{|\mathbf{d}|} = \nabla_f^{\perp}\phi \tag{3.2}$$

The difference between the cell of interest, P, and the neighboring cell, N, divided by the cell size, $|\mathbf{d}|$, gives the divergence perpendicular to the face, thus in the direction of the surface face vector. In case the mesh is non-orthogonal a correction term needs to be considered.

3.3.3 Divergence

The divergence is considered over a volume-cell with *Gauss' theorem.*

$$\int_V \nabla\cdot\phi dV \approx \sum_f \mathbf{S}_f\cdot\phi_f \tag{3.3}$$

3.3.4 Laplacian

The laplacian is considered over a volume cell, which, with *Gauss' theorem*, is converted into an inner product between the gradient and surface vector.

$$\int_V \nabla\cdot\nabla\phi \approx \sum_f \mathbf{S}_f\cdot\nabla\phi_f \tag{3.4}$$

3.3.5 Source Term

The source term, indicating production or destruction of a variable, can be a general function of ϕ. Going from the general linearization to the, for FV schemes necessary, integral form.

$$S_\phi = \phi S_I + S_E \rightarrow \int_V S_\phi(\phi) dV = S_I V_P \phi_P + S_E V_P \tag{3.5}$$

S_ϕ is the linearised source term, governed by S_I and S_E, respectively the implicit and explicit source terms, and V_P is the volume of the cell. The implicit term, S_I, uses the the current iteration's variables. The explicit term, S_E, uses previous iteration's variables. Both variables may be a function of the previous iteration's ϕ.

3.3.6 Under-Relaxation

Under-relaxation is a trick used to prevent large fluctuations between iterations. It is based on adding less than the total correction of an iteration to a variable, by a factor of α, the relaxation factor. This may cause convergence to occur slower, but guarantees stability for sufficiently low α.

$$\phi_{new} = \phi_{old} + \alpha \phi_{correction} \tag{3.6}$$

Note that $0 < \alpha \leq 1$

3.4 The Solver: my_pebbleBedFoam

my_pebbleBedFoam is the name of the pebble-bed solver that was tested in this thesis. The solver runs the subsolvers in this specific order:

- calculate UEqn.H, governing the momentum transfer.
- calculate hEqn.H, governing the heat transfer of the coolant.
- calculate pEqn.H, governing the pressure-momentum coupling.
- calculate TpebEqn.H, governing the pebble temperature.

These sequence will be iterated, until the solution is converged. The following subsections will explain the equations that are solved by the code.

3.4.1 UEqn.H

Regarding the velocity field, conservation of momentum, eq. 2.1, is rewritten to facilitate solving the equation for **u**.

$$\nabla \cdot (\phi \mathbf{u}) - (\nabla \cdot \phi)\,\mathbf{u} + F_{por}\mathbf{u} + \epsilon \mathbf{T} = -\epsilon \nabla p \tag{3.7}$$

This equation is solved with the explicit p value. This **u** field does not obey the continuity equation, eq. 2.4, which will be corrected for in the pEqn routine. The equation is solved with a relaxation factor of 0.8 and the $\nabla \cdot (\phi \mathbf{u})$ term uses the Gauss scheme with upwind differencing, or simply Gauss upwind.

3.4.2 hEqn.H

First of all this sub-solver updates the values of α and c_p and recalculates the values of Pr, Re, Nu, htc and α_{eff}, with equations 2.6 to 2.10. The coolant energy conservation equation, eq. 2.5, has been rewritten to make it more suitable for computational calculation.

$$\begin{aligned} \nabla \cdot (\phi h) - (\nabla \cdot \phi)h - \nabla \cdot (\epsilon \alpha_{eff} \nabla h) + \frac{htc}{c_p} h \;&=\; \nabla \cdot \left(\frac{\phi}{\rho} p \right) \\ &\quad -P \nabla \cdot \left(\frac{\phi}{\rho} \right) + htc T_{peb} \end{aligned} \tag{3.8}$$

After solving this equation for h, the temperature effects on physical parameters are updated, and the sub-solver is done. This equation uses a relaxation factor of 0.9. The $\nabla \cdot (\phi h)$ term is approximated with Gauss upwind, the $\nabla \cdot (\epsilon \alpha_{eff} \nabla h)$ term is approximated with Gauss linear, using central differencing and ρ is interpolated with the use of central differencing.

3.4.3 pEqn.H

The pEqn calculates the pressure field and adjusts the velocity field so it obeys the continuity equation, eq. 2.4. This subroutine uses the SIMPLE-algorithm to compute the pressure and velocity fields [4] [10].

The SIMPLE Algorithm

The momentum equation, eq. 2.1, is written down in a matrix notation, eq. 3.9.

$$A_P^{u_i} u_{i,P}^{n+1} + \sum_l A_l^{u_i} u_{i,l}^{n+1} = Q_{u_i}^n - \left(\frac{\delta p^n}{\delta x_i}\right)_P \tag{3.9}$$

In this equation A is the matrix representation of equation 2.1, u indicates the velocity field, Q indicates a source term, p is the pressure field, and x is the cell width of the gradient. The P index indicates the node in question and the l index indicates the neighboring nodes. The i index denotes the direction of u and x.

In the code, current values for the pressure and source terms are not known, so the values of previous computations are used. Rewriting equation 3.9, with the use of $m = n+1$, and using the fact that the source and pressure terms use previously computed values, results in equations 3.10 and 3.11.

$$u_{i,P}^{m*} = \frac{Q_{u_i}^{m-1} - \sum_l A_l^{u_i} u_{i,l}^{m*}}{A_P^{u_i}} - \frac{1}{A_P^{u_i}} \left(\frac{\delta p^{m-1}}{\delta x_i}\right)_P \tag{3.10}$$

$$u_{i,P}^{m*} = \tilde{u}_{i,P}^{m*} - \frac{1}{A_P^{u_i}} \left(\frac{\delta p^{m-1}}{\delta x_i}\right)_P \tag{3.11}$$

The use of the asterisk, $*$, indicates that this value is not yet the final value of u, because it does not obey continuity yet. The first term on the right hand side of equation 3.10 is replaced by $\tilde{u}$ for convenience.

The continuity equation, eq. 2.4, can be rewritten to equation 3.12

$$\frac{\delta \rho u_i^m}{\delta x_i} = 0 \tag{3.12}$$

In this equation u_i^m is the new velocity field, which obeys continuity. This can be rewritten as equation 3.13, where p^m is the new pressure field. Inserting this in equation 3.12 results in equation 3.14.

$$u_{i,P}^m = \tilde{u}_{i,P}^{m*} - \frac{1}{A_P^{u_i}} \left(\frac{\delta p^m}{\delta x_i}\right)_P \tag{3.13}$$

$$\frac{\delta}{\delta x_i}\left[\frac{\rho}{A_P^{u_i}}\left(\frac{\delta p^m}{\delta x_i}\right)\right]_P = \left[\frac{\delta \rho \tilde{u}_i^{m*}}{\delta x_i}\right]_P \tag{3.14}$$

This equation can be solved for the pressure field p, and with this corrected pressure, a correction for the velocity can be computed, using the updated version of equation 3.11, equation 3.13. The new velocity field obeys continuity, but not necessarily conservation of momentum 2.1. With the new pressure field, a new u_i^{m*} can be calculated, and again a new pressure field. This process is iterated until the pressure and velocity fields are converged.

This solver uses a lot of unstable mathematics, so a low relaxation coefficient of 0.22 is used. The mass flux ϕ is interpolated with central differencing and the pressure laplacian, $\nabla^2 \left(\frac{\epsilon^2 \rho p}{A^{u_i}} \right)$ is evaluated with the Gauss central differencing scheme.

3.4.4 TpebEqn.H

This subroutine calculates λ_g and k_{peb}, eq. 2.13, followed by equation 3.15.

$$htc(T_{peb} - T_{He}) = \nabla^2(k_{peb} T_{peb}) + Q \tag{3.15}$$

A relaxation factor of 0.9 is used on the pebble temperature and the laplacian uses the Gauss gradient method, with linear interpolation.

Chapter 4

Test Cases

The code is verified with the use of several test cases. Each test case focuses on certain parts of the code. The momentum equation is tested with the use of a one-dimensional pressure drop case and a two dimensional case, concerning laminar flow between parallel plates. To conclude the momentum part of the code, an experiment done by Achenbach is simulated. The thermal part of the code is tested by a one-dimensional heat transfer case.

4.1 One-Dimensional Pressure Drop

This calculation concerns the pressure drop in a one dimensional case, which has the benefit it has a simple analytical solution. Besides verification of the pressure drop, the dependence on mesh size and measuring position are examined.

4.1.1 Set up, Measurement & Expectations

The geometry consists of a one-dimensional column of 1.4 m high. From 0.2 m to 1.2 m the porosity is 0.39 and the first and last 0.2 m the porosity is 1, so the flow field can stabilize. The pebble diameter is 0.06 m. The fluid properties are kept constant, with $\mu = 10^{-5}\ kgm^{-1}s^{-1}$ and $\rho = 1\ kgm^{-3}$. The velocity is varied between 0.02 and 10 ms^{-1}, resulting in a Reynolds number distribution from 120 to 60000. To investigate the effect of cell size, the mesh size is varied between 140, 280, 700 and 1400 cells. Additionally, for the 1400 mesh the pressure drop is calculated over three different intervals for the 1400 cell mesh:

- total field: The pressure difference between the in- and outlet of the domain.

- porosity field: The pressure difference between the begin and end part of the porous field.
- derivative: The pressure gradient in the middle of the porous field.

The pressure drop should follow the analytical solution, given by Ergun's equation, eq. 2.2. The dependence on the mesh size is unknown, as is the dependence on the measuring position.

4.1.2 Results

The results of the measuring position computations are displayed as a relative difference between the analytical solution and the calculated value, see figure 4.1. This figure shows that the derivative in the pebble-bed is in excellent agreement with Ergun's equation. The pressure drop over the whole domain and the pressure drop over the porous field show some error, with the biggest relative error in the pressure drop between the begin and end of the pebble-bed. For almost all Reynold numbers the pressure drop over the porous region is larger than over the total field.

This behavior is caused by unstable regions caused by the discrete transition of porosity at the beginning and end of the porous region. This discrete transition causes fluctuations in the pressure and velocity fields, which cause greater errors on the transition position than over the total bed.

The results of the mesh dependance are displayed in figure 4.2. This computation uses the relative difference of the total field with Ergun's analytical solution, eq 2.2. The reciprocal relative difference is plotted against the mesh size. The result is a linear relationship, which indicates a $\frac{1}{nodes}$ relationship. A possible explanation is that the unstable regions, causing the inaccuracies, always have the same length in nodes. When the number of nodes increases, the percentage of nodes that have unstable behavior decreases as $\frac{1}{r}$.

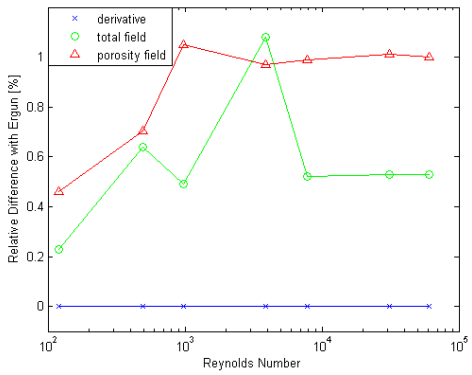

Figure 4.1: The relative difference between the analytical solution and the computed value plotted against the Reynolds number

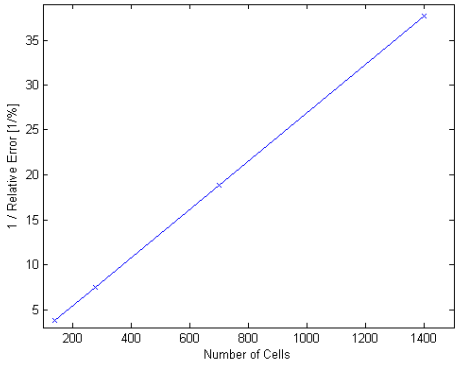

Figure 4.2: The reciprocal relative difference between the analytical solution and the computed value plotted against the mesh size for a Reynolds number of 120

4.2 Two-Dimensional Laminar Flow Between Parallel Plates

This computation concerns the perpendicular velocity of a laminar flow between two parallel plates. For this experiment, the expected solution is Poiseuille flow, see equation 4.1.

4.2.1 Set Up, Measurements & Expectations

The geometry consists of two parallel plates, through which a fluidum flows. The plates are 0.01 m apart and are 1 m long. A no slip boundary condition exists on the plates and the inlet velocity is 1 ms^{-1}, which is distributed uniformly. The fluid properties are kept constant, with $\mu = 10^{-3}$ $kgm^{-1}s^{-1}$, $\rho = 1$ kgm^{-3} and since there is no pebble bed, the porosity is 1. This results in a Reynolds number of 10. The mesh is 200 cells in the flow direction and 50 cells perpendicular to the flow direction.

Poiseuille flow, see equation 4.1, is expected for the perpendicular velocity, since the mesh size should be adequate and the Reynolds number is well within the laminar flow range. The analytical solution for this case is:

$$u_x = \frac{1}{2\mu}\frac{dp}{dx}\left(y^2 - \left(\frac{d}{2}\right)^2\right) \tag{4.1}$$

u_x is the flow in the flow direction, μ is the dynamic viscosity, $\frac{dp}{dx}$ is pressure gradient in the flow direction and the last term describes the distance from the walls, where y is the radial distance, and d is the distance between the two plates. The Poiseuille flow has a quadratic distribution. The computation will be compared to the quadratic solution, to check whether it fits poiseuille flow.

4.2.2 Results

The flow field between the two plates is displayed in figure 4.3. The fluid flow first has to stabilize, because the velocity field is uniform at the inlet.

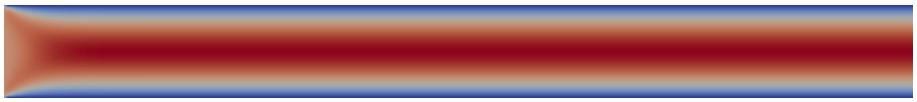

Figure 4.3: The fluid velocity in the flow direction, red indicates high velocities, blue indicates low velocities

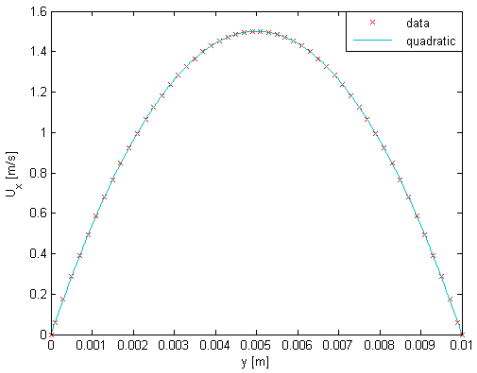

Figure 4.4: The data points of the velocity distribution(data), fitted by a quadratic function(quadratic).

The velocity profile is examined at 0.8 m from the inlet, see figure 4.4. It is also fitted by a quadratic distribution, eq. 4.1. The sum of the absolute errors is 0.06 %, which means this test case is in excellent agreement with the expected analytical solution.

4.3 Achenbach's Experiment

Achenbach measured the pressure drop over a packed bed in a physical experiment for Reynolds ranging from 150 to $3 * 10^5$ and compared his results with the pressure drop relation from the KTA, see equation 2.3 [1]. This test case will compare the code to the experimental data gathered by Achenbach.

4.3.1 Set Up, Measurements & Expectations

The geometry consists of a 1.4 m high cylinder, with a radius of 0.5 m. The first and last 0.2 m have porosity 1, the pebble-bed has a uniform porosity of 0.39 and a pebble diameter of 0.06 m. The fluid is air, at a constant temperature of 293 K. The velocity on the walls has the slip boundary condition and the inlet velocity ranges from 0.1 to 25.6 ms^{-1}. Two outlet pressures are used of 10^5 and 10^6 Pa. Different outlet pressures are used, because if the pressure drop is in the order of the outlet pressure, the pressure drop affects the fluid properties. The mesh consists of 70 nodes in the flow direction, 50 nodes in the radial direction, and uses a wedge symmetry plane to simulate a cylinder.

In the experiment done by Achenbach, [1], a measure for the pressure drop is given by the use of the pressure drop coefficient, ψ, see equation 4.2

$$\psi = \frac{2d_{peb}}{\rho|u|^2}\left(\frac{\epsilon^3}{1-\epsilon}\right)\frac{\Delta p}{\Delta h} \tag{4.2}$$

The paper concludes with a relationship for ψ, see equation 4.3, which is the KTA relation for the pressure drop [1].

$$\psi = \frac{320}{\frac{Re}{1-\epsilon}} + \frac{6}{\left(\frac{Re}{1-\epsilon}\right)^{0.1}} \tag{4.3}$$

The data Achenbach gathered in his experiment and his fit through the data are displayed in figure 4.5. This data will be used for code comparison. The paper of Achenbach notes that the relation, eq. 4.3, does not describe his measurements for Reynolds numbers higher than $8 * 10^4$. The computed results are expected to follow Ergun's equation, eq. 2.2, since this equation is used in the code.

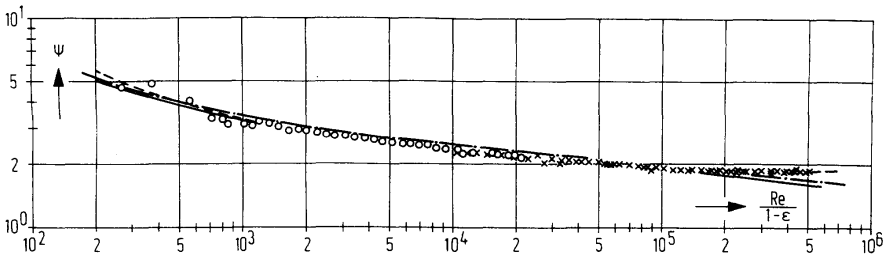

Figure 4.5: The data Achenbach gathered in his experiment and his fit of the data. The o indicate measurements at atmospheric pressure, the x indicate measurements at higher pressure.

4.3.2 Results

The results of the computation are plotted in figure 4.6 and in table 4.1. For the low pressure situation, the computational result and the analytical result start to diverge for Reynolds numbers of $2 * 10^4$. For the high pressure situation, the computational and analytical result start to diverge for Reynolds numbers of $2 * 10^5$. The divergence for higher Reynolds numbers is because the pressure drop over the domain is high compared to the outlet pressure, resulting in significant change in the density over the domain, see table 4.1.

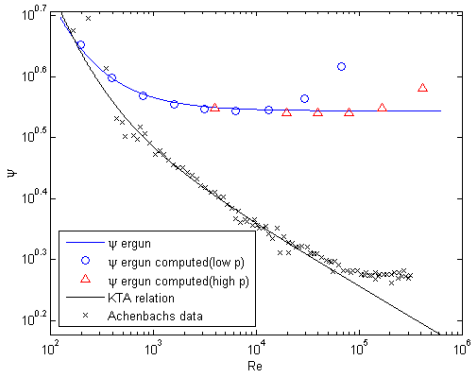

Figure 4.6: The pressure drop coefficient according to the analytical Ergun equation (ψ ergun), the computed data (ψ ergun computed), the KTA relation, and the data gathered by Achenbach.

Table 4.1: The Reynolds number, ψ, pressure drop, inlet density and outlet pressure(low (10^5)) or high (10^6)) of the computation of Achenbach's experiment

Re [−]	ψ [−]	Δp $[Pa]$	ρ $[kgm^{-3}]$	p_{out} $[Pa]$
196	4.49	1	1.1863	10^5
392	3.96	4	1.1863	10^5
785	3.71	15	1.1865	10^5
1570	3.58	58	1.1869	10^5
3147	3.52	229	1.1890	10^5
6335	3.50	918	1.1970	10^5
13035	3.52	3800	1.2314	10^5
29648	3.67	18049	1.4004	10^5
67655	4.13	72402	2.0452	10^5
3920	3.54	36	11.863	10^6
19620	3.48	883	11.865	10^6
39270	3.48	3538	11.870	10^6
78800	3.51	14340	12.032	10^6
159640	3.68	60944	12.580	10^6
340590	4.68	330164	15.780	10^6

Ergun's equation and the KTA relation, and thus the measurements, start to diverge for a Reynolds number of around 400. The PBMR case expects Reynolds numbers of around $4 * 10^4$. When the measured data and Ergun's relation are compared at this Reynolds number, ψ differs by a factor of $\frac{10^{0.31}}{10^{0.54}} = 1.70$. An error of a factor 1.70 can be expected at the PBMR test case [7].

4.4 Analytical Heat Transfer

This calculation concerns the one-dimensional heat transfer of a stacked bed. The heat transfer occurs from the pebbles to the fluid.

4.4.1 Set Up, Measurements & Expectations

The geometry is a one-dimensional column of 1 m high. The porosity is 0.4 and the pebble diameter is 0.06 m. Fluid properties are kept constant in the pebble bed, with $\rho = 1\ kgm^{-3}$ and $c_p = 4 * 10^3\ m^2s^{-2}K^{-1}$. The htc is fixed at 10000 $kgs^3m^{-1}K^{-1}$. The inlet velocity is set at 2.5 ms^{-1}. This results in a mass flux, ϕ, of 1 $kgs^{-1}m^{-2}$ and an α_{eff} of 0.01582 $kgm^{-1}s^{-1}$, see eq. 2.7. The inflow temperature of the coolant is set at 500 K, a zero gradient boundary condition for the outflow temperature of the coolant is enforced and the pebbles are kept at a constant temperature of 1000 K. Two meshes are used, to check the effect of cell size on the error, of 200 and 2000 nodes in the flow direction. This test case has an analytical solution, which can be calculated by rewriting equation 2.5 into equation 4.4

$$\nabla(\underbrace{c_p\phi}_{\text{A}}T) - \nabla(\underbrace{\epsilon c_p \alpha_{eff}}_{\text{B}}\nabla T) + \underbrace{htc}_{\text{C}}T = \underbrace{htcT_{peb}}_{\text{D}} \tag{4.4}$$

Here T is the coolant's temperature and T_{peb} is the pebble temperature. For one dimension, equation 4.4 has the standard solution of equation 4.5

$$T(x) = c_1 e^{\left(\frac{A-\sqrt{A^2+4BC}}{2B}\right)x} + c_2 e^{\left(\frac{A+\sqrt{A^2+4BC}}{2B}\right)x} + \frac{D}{C} \tag{4.5}$$

Using the boundary conditions in equations 4.7 and 4.6, c_1 and c_2 can be calculated.

$$\left.\frac{dT_{coolant}}{dx}\right|_{x=0} = 0 \tag{4.6}$$

$$T_{coolant}|_{x=L} = 500K \tag{4.7}$$

In these boundary conditions, x is the position along the column and L is the total length of the column, 1 m. Boundary condition eq. 4.6 is not a physical boundary, but is enforced because the code requires a boundary condition on the outlet. This results in equation 4.8 for c_1 and equation 4.9 for c_2.

$$c_1 = \frac{500 - \frac{D}{C}}{e^{\left(\frac{A-\sqrt{A^2+4BC}}{2B}\right)} + \frac{A-\sqrt{A^2+4BC}}{A+\sqrt{A^2+4BC}} e^{\left(\frac{A+\sqrt{A^2+4BC}}{2B}\right)x}} \tag{4.8}$$

$$c_2 = \frac{500 - \frac{D}{C}}{e^{\left(\frac{A+\sqrt{A^2+4BC}}{2B}\right)} + \frac{A+\sqrt{A^2+4BC}}{A-\sqrt{A^2+4BC}} e^{\left(\frac{A-\sqrt{A^2+4BC}}{2B}\right)x}} \tag{4.9}$$

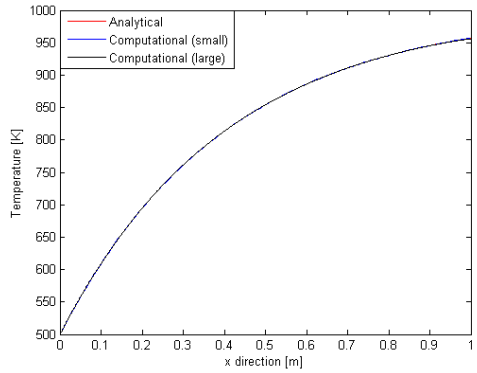

Figure 4.7: The temperature profile of the computation and of the analytical solution

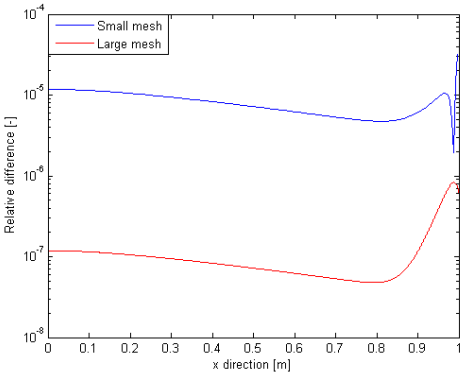

Figure 4.8: The relative error between the computational result and the analytical solution

4.4.2 Results

The results are plotted in figures 4.7 and 4.8. Figure 4.7 shows the computational temperature profile and is compared to the analytical solution, eq. 4.5. Figure 4.8 shows the difference between the computational result and the analytical solution.

The computational result and the analytical solution are in good comparison, with a maximum difference of 0.04%. The relative difference shows an increase in error near the outlet. This is because the code forces zero gradient between the node and the wall. The analytical solution has zero gradient on the wall, so there is a difference in position. For the large mesh, the error is less than for a the small mesh, as expected.

This test case shows that the code is capable of calculating the heat transfer through helium correctly.

Chapter 5

Pebble-Bed Modular Reactor

The Pebble-Bed Modular Reactor (PBMR) is an international benchmark, based on the PBMR-400MW design. It is a code-to-code comparison to test different CFD codes. The tests on the PBMR concern steady-state and transient processes, but this case will only focus on the steady-state processes [7].

5.1 Set Up, Measurement & Expectations

The geometry is an axisymmetric cylinder, with a height of 11.5 m, an inner radius of 1 m and an outer radius of 1.85 m. The first 0.5 m from the inlet has a porosity of 1. The packed bed, from 0.5 to 11.5 m has a porosity of 0.39 and a pebble radius of 0.06 m. The helium inlet mass flux is 192.7 kgs^{-1} and the outlet pressure is 9 MPa. Using the fluid properties of helium at 9 MPa, a velocity of 4.291 ms^{-1} is found. The inlet temperature of the coolant is 773 K and the power distribution is given in figure 5.1, with a total power of 400 MW [7].

The benchmark suggests simulating a 2 m thick graphite wall, with an outer temperature of 293 K, as a boundary condition for the outer wall. This cannot be simulated by the code, so two methods will be used to account for this.

- Zero Gradient: No wall heat flux, by forcing a zero gradient on both walls.

- Fixed Value: Fixed outer wall temperature, supplied by previous CFD results from the benchmark. The inner wall temperature has a zero gradient boundary condition.

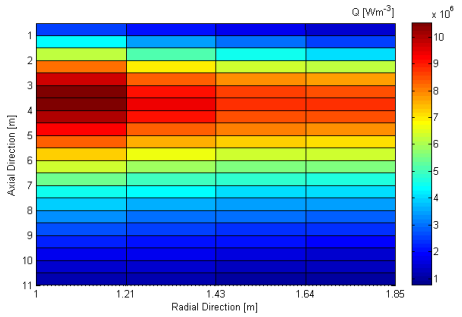

Figure 5.1: The power density profile inside the PBMR reactor. The left side is towards the reactor center [3].

A 230 by 50 cells mesh is used. Most computations of the benchmark suggest an outlet helium temperature of 1172 K, an average helium temperature of 1022 K, an average pebble temperature of 1042 K and a pressure drop of 2.75 bar.

5.2 Results

The results of the CFD code will be discussed in three sections, first discussing the helium temperature, followed by the pebble temperature and finally the pressure drop.

5.2.1 Helium Temperature

The numerical results of the benchmark case are given in table 5.1. This table contains the average helium temperature of the total porous region and from the outlet, the average pebble temperature, and the pressure drop of the benchmark and of the two computations. The helium temperature profiles in the axial and in the radial direction can be seen in figures 5.2 and 5.3.

Table 5.1: The results of the PBMR computation, with the fixed value case and zero gradient case compared with the benchmark results.

	Fixed Value Case	Zero Gradient Case	Benchmark
T Helium Outlet $[K]$	1159.3	1153.7	1172
T Helium Average $[K]$	1011.9	1007.6	1022
T Pebble Average $[K]$	1034.2	1030.0	1042
Pressure Drop $[Pa]$	$4.80*10^5$	$4.77*10^5$	$2.75*10^5$

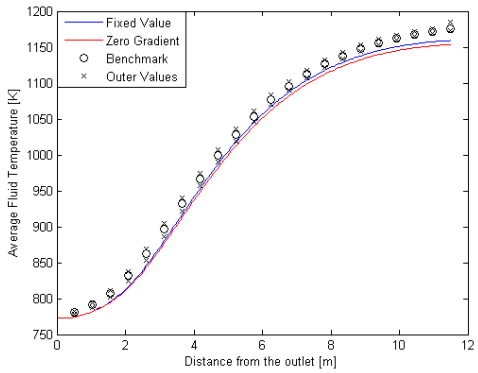

Figure 5.2: The average helium temperature in the axial direction of the fixed value and zero gradient case, compared with the average, minimum and maximum benchmark temperatures.

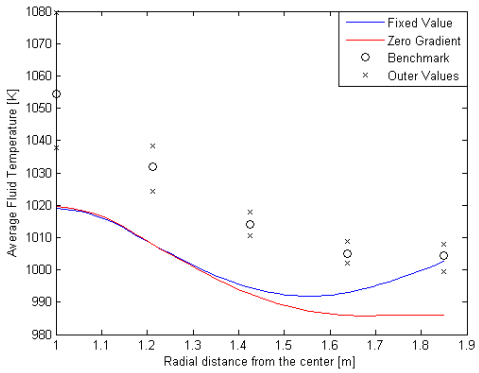

Figure 5.3: The average helium temperature in the radial direction of the fixed value and zero gradient case, compared with the average, minimum and maximum benchmark temperatures.

Table 5.1 shows that the outlet and average fixed value temperatures are higher than the zero gradient temperatures. This means the pebble-bed is heated by the fixed value boundary condition on the outer wall. Although both cases and the benchmark provide similar temperature profiles, the fixed value and the zero gradient cases result in different outlet temperatures than the benchmark suggests, with a maximum temperature difference of around 20 K at the outlet. This is outside of the minimum value of the benchmark's computations.

The temperature difference is caused by the absorption of energy due to expansion. The benchmark's computations do not account for this. The temperature difference of 20 K can also be shown by equation 5.1. In this equation the difference in thermal outlet energy is compared to the energy the fluid absorbs when it expands through the pebble-bed. A specific heat of 5195 $m^2 s^{-2} K^{-1}$ is used, and the density at the in- and outlet are 4.525 kgm^{-3} and 4.295 kgm^{-3} respectively. A constant temperature of 1000 K and a pressure drop of 5 bar are assumed.

$$mc_p\Delta T = Q = \int_{V_1}^{V_2} pdV \approx p\Delta V = (\phi\Delta t)\left(\frac{1}{\rho_1} - \frac{1}{\rho_2}\right)p \tag{5.1}$$

$$\Delta T = \left(\frac{1}{\rho_2} - \frac{1}{\rho_1}\right)\frac{p}{c_p} = \left(\frac{1}{4.295} - \frac{1}{4.525}\right)\frac{9 * 10^6}{5195} = 20.5K$$

Table 5.2: The results of the computation with the pressure gradient term turned off in the enthalpy equation, for two cases and compared with the benchmark.

	Fixed Value Case	Zero Gradient Case	Benchmark
T Helium Outlet $[K]$	1176.0	1174.8	1172
T Helium Average $[K]$	1019.3	1016.5	1022
T Pebble Average $[K]$	1041.5	1038.8	1042
Pressure Drop $[Pa]$	$4.80*10^5$	$4.77*10^5$	$2.75*10^5$

To verify that the pressure gradient term in equation 2.5 is indeed the cause for the low average and output temperature of the helium, this term was turned off and the computations were repeated. This resulted in the expected behavior, see table 5.2. The temperature differences between the cases and the benchmark are similar to the differences between the different computations of the benchmark.

The radial profiles of both cases do not match the benchmark, and the code's results are outside the outer values of the benchmark's computations. The fixed value boundary condition on the outer wall matches the expected result, because it's value is enforced. The temperature difference at the outer wall for the zero gradient case is around 20 K, which is caused by the absorption of energy due to expansion, see equation 5.1. The difference at the inner wall is higher, which is because the benchmark uses a different boundary condition on the inner wall.

5.2.2 Pebble Temperature

The pebble temperature profiles can be seen in figures 5.4 and 5.5. The pebble temperature profiles show wrinkles, which arise from the discrete steps in the power density field. The temperature profiles of both cases follow the benchmark's lowest temperature profile, in both the axial and the radial direction.

The average pebble temperatures can be found in table 5.1, and show that both cases result in a lower average temperature than the benchmark suggests. When the pressure gradient is turned off in the code, see table 5.2 the temperature of both cases and the benchmark differ less. Besides the pressure gradient term, a different formula for the Nusselt number, eq. 2.9, is used in the benchmark, and the code uses a slightly simplified relation for the pebble-pebble heat transfer, eq. 2.11, but this does not seem to have a significant effect.

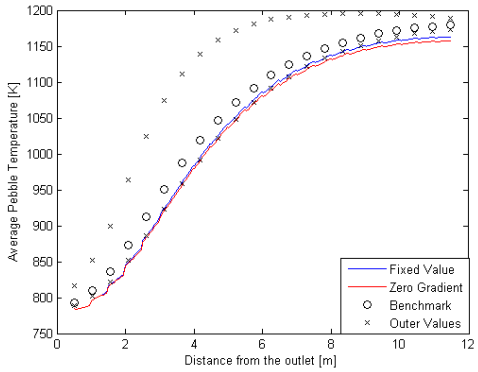

Figure 5.4: The average pebble temperature in the axial direction of the fixed value and zero gradient case, compared with the average, minimum and maximum benchmark temperatures.

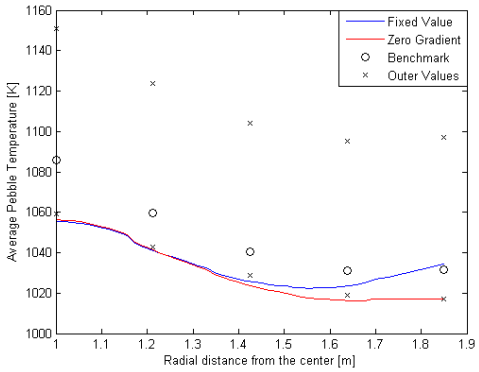

Figure 5.5: The average pebble temperature in the radial direction of the fixed value and zero gradient case, compared with the average, minimum and maximum benchmark temperatures.

5.2.3 Pressure Drop

The pressure drop profile can be seen in figure 5.6 and the effective pressure drops can be seen in table 5.1. The pressure profiles and the pressure drop of the fixed value and zero gradient case differ from the benchmark.

Both pressure drops are around $4.8 * 10^5\ Pa$, compared to a pressure drop of $2.75 * 10^5\ Pa$ for the benchmark. Figure 4.6 of the Achenbach testcase showed a ratio between the Ergun and KTA relation, at a Reynolds number of $4 * 10^4$, of $\frac{\psi_{Ergun}}{\psi_{KTA}} = 1.70$. The ratio between pressure drops is $\frac{\Delta p_{Computation}}{\Delta p_{Benchmark}} = 1.75$, which shows the different pressure drop relations are indeed the cause of the different pressure drops.

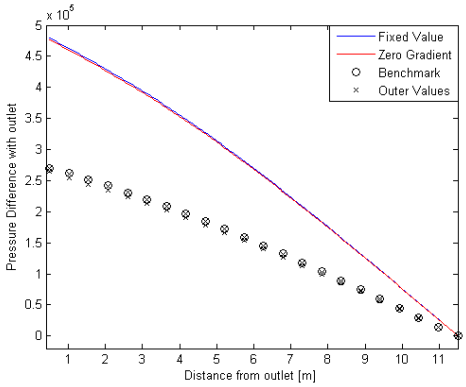

Figure 5.6: The pressure profile of the PBMR case for the fixed value and zero gradient case, compared with the average, minimum and maximum pressure profiles of the benchmark.

Chapter 6

Conclusions

A code implemented in OpenFOAM was investigated using several simple test cases with analytical or experimental solutions, and by calculating the PBMR steady state thermohydraulics benchmark.

6.1 Experimental Conclusions

Comparison of the code for the simple test cases showed the code evaluates the equations implemented in the solver with good accuracy, with one exception. The one-dimensional test case for the pressure drop pointed out that near discrete transitions in porosity, such as at the top of a pebble bed, fluctuations arise in the pressure and velocity fields. As these fluctuations are local and pebble-bed reactors are large, the influence of these fluctuations is small.

The results of the PBMR case differed on several points with those reported by the participants. Firstly, the coolant and pebble temperature fields were on average 20 K lower. This difference was caused by the inclusion of the expansion term in the helium heat transfer equation, causing additional cooling of the coolant. This expansion term was not included in the benchmark, although our calculations show this term should not be ignored.

Also, the results for the pressure drop over the pebble bed showed a 75 % higher pressure drop over the pebble-bed. This difference is because the code uses Ergun's relation and the benchmark uses the KTA relation. As seen in the Achenbach case, the KTA relation closely follows the measurement. Ergun's relation, however predicts a significantly higher pressure drop at higher Reynolds numbers, with a difference of 70 % for Reynolds numbers of $4 * 10^4$, comparable to the Reynolds numbers in the PBMR.

Finally, as the code was not yet capable of calculating heat transfer

through the central and side reflector, artificial boundary conditions were used at the walls of the pebble bed. This, together with different relations for the Nusselt number for the heat transfer from the pebbles to the coolant and a slightly simplified relation for the pebble-pebble heat transfer, resulted in small differences in the radial pebble and helium temperature profiles.

Still, the axial coolant and pebble temperature profiles were in good agreement with the results from the benchmark participants which, together with the good results from the analytical test cases, brings us to the conclusion that the solver implemented in OpenFOAM can be used to compute heat and mass transfer in a pebble-bed reactor, especially if Erguns equation for the pressure drop is replaced by KTA's relation.

6.2 Future Research

This research leaves some questions unanswered, which could be interesting for further investigation.

- Discrete steps in porosity result in fluctuations of the pressure and velocity fields. A different approach to these discrete steps may result in less fluctuation.
- The porosity in the PBMR benchmark is uniform. In actual pebble-bed reactors porosity approaches one near the walls, causing a different flow pattern. Computations with non-uniform porosity can determine what effect this has.
- The PBMR benchmark states different relations than the relations used in the code. Changing the relations in the code to the relations suggested by the benchmark could verify if the code can provide the same results as the benchmark.
- The term for pebble-pebble interactions through contact in the equation for the diffusion of heat between the pebbles has been neglected, because it was thought to have negligible influence. A study to the heat transfer through touching pebbles could determine if this approximation was justified.
- The code was unable to simulate heat transfer through a graphite reflector, and therefore unable to calculate the heat flux through the wall in the manner suggested by the benchmark. Adapting the code to be able to simulate this reflector could check whether the code is capable of calculating the radial temperature profile correctly.

Bibliography

[1] E. Achenbach, *Heat and flow characteristics of packed beds*, Institute of Energy Process Engineering, Julich, Germany, (1994)

[2] S. Ergun, *Fluid flow through placed columns*, Chemical engineering progress, 48(2):90-98, (1952)

[3] *PBMR-400 Data Exercise Sheet 2*, OECD/NEA/NSC, (2006)

[4] J.H. Ferziger and M. Perić. Springer, *Computational Methods for Fluid Dynamics*, 3rd Edition, (2002)

[5] *OpenFOAM User Guide*, OpenCFD, Version 1.7.1, (2010)

[6] Bird, Steward, Lightfoot, *Transport Phenomena*, Chemical Engineering Department University of Winsconsin-Madison, John Wiley and Sons Inc. 2nd Edition, (2002)

[7] *PBMR coupled neutronics/thermal hydraulics transient benchmark the PBMR-400 core design - Benchmark edition*, OECD/NEA/NSC, (2007)

[8] Donald A. Nield, Adrian Bejan, *Convection in Porous Media*, Springer, (2006)

[9] L.P.B.M. Janssen, M.M.C.G. Warmoeskerken, *Transport Phenomena Data Companion*, VSSD, (2006)

[10] T. Craft, *Pressure-Velocity Coupling*, Lecture Notes

[11] *Flow through porous passages*, Fluid Mechanics, Tutorial #4, www.freestudy.co.uk

[12] CG du Toit, *The Flow and Heat Transfer in a Packed Bed High Temperature Gas-Cooled Reactor.*, North-West University, (2010)

[13] C.J. Visser, *Modelling Heat and Mass Flow Through Packed Pebble Beds: A Heterogeneous Volume Averaged Approach*, University of Pretoria, (2007)

[14] S. Ergun and A.A. Orning, *Fluid Flow through Randomly Packed Columns and Fluidized Beds*, Carnegie Institute of Technology, (1949)

[15] T. Frederikse, *CFD Calculations on the helium cooling of the Pebble Bed Reactor Core*, TU Delft, (2010)

[16] *A safety re-evaluation of the AVR pebble bed reactor operation and its consequences for future HTR concepts*, Berichte des Forschungszentrums Jülich, (2008)

[17] V. Gnielinski, *Equations for the calculation of heat and mass transfer during flow, through stationary spherical packings at moderate and high Pclet number*, International Chemical Engineering, Vol. 21, pp. 378-383, (1981)

[18] S. Yagi and D. Kunii, *Studies on effective thermal conductivities in packed beds*, American Institute for Chemical Engineers, Volume 3, Issue 3, pages 373-381, (1957)

[19] S. Yagi and N. Wakao, *Heat and mass transfer from walls to fluid in packed beds*, American Institute for Chemical Engineers, Volume 5, Issue 1, pages 79-85, (1959)

[20] *Heat transport and afterheat removal for gas cooled reactors under accident conditions*, IAEA-TECDOC-1163, section 4.2.2

[21] E. Tsotsas and H. Martin, *Thermal conductivity of packed beds: A Review*, Institute fur Thermische Verfahrenstechnik, Universitat Karlsruhe, (1987)

[22] G.Dietrich, W. Neumann, N. Röhl, *Decommissioning of the Thorium High Temperature Reactor (THTR-300)*, Hochtempeartur-Kernkraftwerk GmbH

[23] H. Rusche, *Computational Fluid Dynamics of Dispersed Two-Phase Flows at High Phase Fractions*, Department of Mechanical Engineering, Imperial College London, (2002)