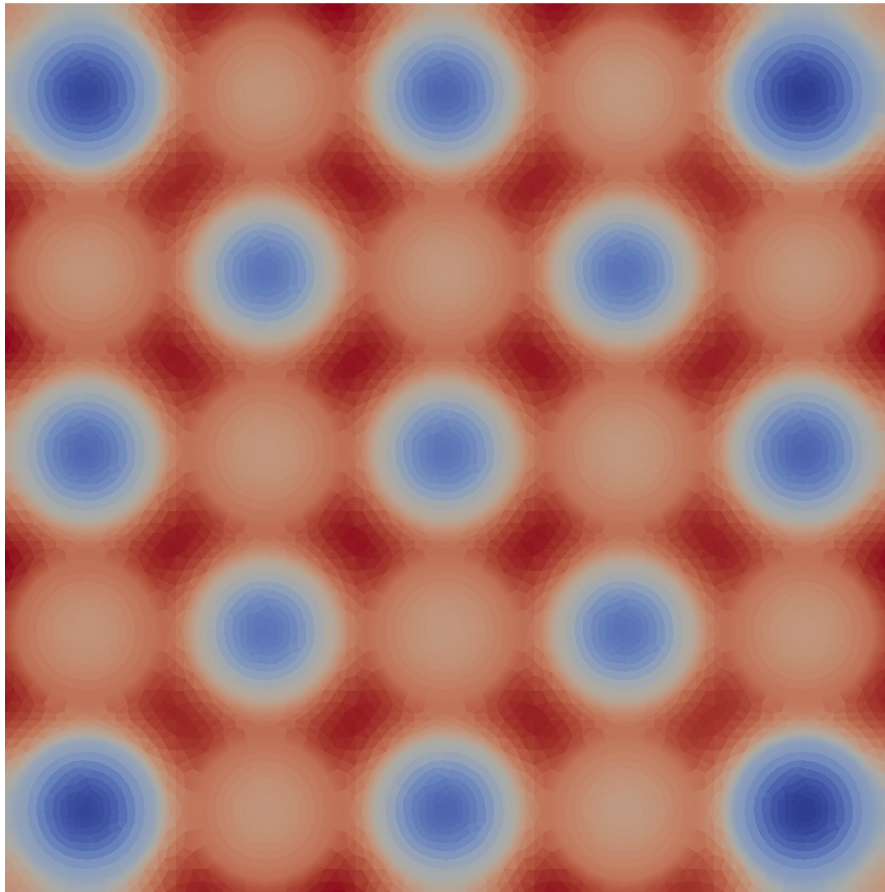


# CHALMERS



Development of an integrated deterministic neutronic/thermal-hydraulic model using a CFD solver

Masters's Degree Thesis

*Master's thesis in Master Program of Nuclear Engineering*

KLAS JARETEG

Department of Applied Physics

*Division of Nuclear Engineering*

CHALMERS UNIVERSITY OF TECHNOLOGY

Gothenburg, Sweden 2012

CTH-NT-259

ISSN 1653-4662



CTH-NT-259

Development of an integrated deterministic  
neutronic/thermal-hydraulic model using a CFD solver

Masters's Degree Thesis  
Master's Programme in Nuclear Engineering  
CHALMERS UNIVERSITY OF TECHNOLOGY

KLAS JARETEG

Department of Applied Physics  
*Division of Nuclear Engineering*  
CHALMERS UNIVERSITY OF TECHNOLOGY  
Gothenburg, Sweden 2012

Development of an integrated deterministic neutronic/thermal-hydraulic model using a CFD solver  
Masters's Degree Thesis  
KLAS JARETEG

© KLAS JARETEG, 2012

Master's thesis  
CTH-NT-259  
ISSN 1653-4662  
Department of Applied Physics  
Division of Nuclear Engineering  
Chalmers University of Technology  
SE-412 96 Gothenburg  
Sweden  
Telephone: +46 (0)31-772 1000

Cover:  
Thermal neutron density in a 5x5 simplified PWR fuel bundle (for details see Fig. 4.27)

Chalmers Reproservice  
Gothenburg, Sweden 2012



Development of an integrated deterministic neutronic/thermal-hydraulic model using a CFD solver

Master's thesis in Master Program of Nuclear Engineering

KLAS JARETEG

Department of Applied Physics

Division of Nuclear Engineering

Chalmers University of Technology

## ABSTRACT

In LWR cores, multiple fields of physics will need to be tackled. The power level is determined by the neutron density field, and the generated heat will determine the fuel conditions, which in turn affect the coolant/moderator flow and thermo-physical conditions. As the neutron cross-sections depend on both the fuel and moderator conditions, an interdependence between the neutronics and thermal-hydraulics is found. To perform accurate, high-resolution nuclear reactor core calculations, it is therefore necessary to couple strategies and methodologies from different areas of physics.

In this work a finite volume approach is used to perform steady state, coupled calculations for thermal-hydraulics and neutronics on a shared high-resolution mesh. The thermal-hydraulics is solved by a standard CFD approach, using the SIMPLE algorithm for the pressure-velocity coupling together with an energy equation expressed in terms of temperature. The neutronic cross-sections are pre-calculated using a nuclear Monte Carlo code. The neutronics is added in to the CFD solver, avoiding any external software coupling, and is discretized according to standard finite volume approach. The criticality eigenvalue problem is solved by the power iteration method, which has been added to the CFD solver. To couple the thermal-hydraulics and neutronics an iterative scheme is proposed. The finite volume solver has been implemented in OpenFOAM.

The developed direct coupled methodology is applied to a simplified PWR fuel sub-assembly. The physical dependencies between neutronics and thermal-hydraulics are correctly recovered. The achieved results suggest that this newly-developed coupling procedure provides an unprecedented level of details and accuracy as compared with standard non-direct coupled calculations.

Keywords: high-fidelity, neutronics, thermal-hydraulics, CFD, multi-physics, coupled deterministic modeling, LWR



# Contents

<b>Abstract</b>	<b>i</b>
<b>Nomenclature</b>	<b>1</b>
<b>1 Introduction</b>	<b>5</b>
1.1 Nuclear Power . . . . .	5
1.1.1 Light Water Reactors . . . . .	5
1.1.2 LWR fuel structure . . . . .	6
1.2 Basic physics of a Light Water Reactor core . . . . .	7
1.3 Review of current methodologies for reactor calculations and motivation of the work . . . . .	7
1.4 Description of this work . . . . .	8
1.5 Structure of the thesis . . . . .	9
<b>2 Theory</b>	<b>11</b>
2.1 Neutronics . . . . .	11
2.1.1 Neutron transport equation . . . . .	11
2.1.2 Diffusion approximation . . . . .	12
2.1.3 Discrete energy equation . . . . .	13
2.1.4 Space homogenization . . . . .	14
2.1.5 Concerning the choice of diffusion approximation . . . . .	14
2.1.6 Cross-sections . . . . .	14
2.2 Thermal-hydraulics . . . . .	16
2.2.1 Governing equations . . . . .	16
2.2.2 Time averaging . . . . .	16
2.2.3 Turbulence modeling . . . . .	18
2.2.4 Fluid energy equation . . . . .	19
2.2.5 Solid temperature equation . . . . .	20
2.3 Coupling between neutronics and thermal-hydraulics . . . . .	21
<b>3 Methodology</b>	<b>23</b>
3.1 Discretization of the problem . . . . .	23
3.1.1 Computational grid . . . . .	23
3.1.2 Discretization of the equations . . . . .	24
3.1.3 Boundary conditions . . . . .	26
3.2 Group-wise macroscopic cross-sections and nuclear parameters . . . . .	26
3.3 Solution of the coupled neutronic/thermal-hydraulic problem . . . . .	27
3.3.1 Thermal-hydraulic CFD solver . . . . .	27
3.3.2 Implementation of the multi-group neutron diffusion equation . . . . .	28
3.3.3 Algorithm for the coupled problem . . . . .	30
<b>4 Application</b>	<b>31</b>
4.1 Description of the fuel assembly . . . . .	31
4.2 Matrix solvers . . . . .	34

4.3	Boundary conditions . . . . .	35
4.4	Two-dimensional case . . . . .	35
4.4.1	Computational mesh . . . . .	35
4.4.2	Two-dimensional neutron macroscopic cross-sections . . . . .	38
4.4.3	Convergence . . . . .	38
4.4.4	Results . . . . .	40
4.4.5	Power dependence . . . . .	42
4.4.6	Influence of the computational mesh . . . . .	43
4.4.7	Modeling of the heat transfer in the gap . . . . .	45
4.4.8	Parallelization . . . . .	46
4.5	Three-dimensional case . . . . .	48
4.5.1	Neutron macroscopic cross-sections for the three-dimensional case . . . . .	48
4.5.2	Computational mesh . . . . .	49
4.5.3	Results . . . . .	50
4.5.4	Moderator homogenization . . . . .	53
<b>5</b>	<b>Conclusions</b>	<b>55</b>
5.1	Summary of the methodology . . . . .	55
5.2	Summary of the results . . . . .	55
5.3	Future work . . . . .	56
	<b>References</b>	<b>59</b>
	<b>Acknowledgements</b>	<b>63</b>
<b>A</b>	<b>OpenFOAM overview and usage</b>	<b>65</b>
A.1	OpenFOAM overview . . . . .	65
A.2	Implementation of the new solver . . . . .	66
<b>B</b>	<b>Materials</b>	<b>68</b>
B.1	Serpent material compositions . . . . .	68
<b>C</b>	<b>Code</b>	<b>70</b>
C.1	Serpent . . . . .	70

# Nomenclature

## Symbols:

### Neutronics

- $\Sigma_a$  Absorption cross-section [ $\text{m}^{-1}$ ]  
 $\Psi$  Angular neutron flux [ $\text{m}^{-2}\text{s}^{-1}$ ]  
 $N_m$  Atomic density of a general material m [ $\text{m}^{-3}$ ]  
 $\mu$  Average scattering angle [1]  
 $H^d$  Downscattering source term [ $\text{m}^{-3}\text{s}^{-1}$ ]  
 $\gamma$  Energy per fission [J]  
 $\Sigma_f$  Fission cross-section [ $\text{m}^{-1}$ ]  
 $\chi$  Fission neutron spectrum [1]  
 $F$  Fission source term [ $\text{m}^{-3}\text{s}^{-1}$ ]  
 $\Sigma_{s0}$  Isotropic scattering cross-section [ $\text{m}^{-1}$ ]  
 $\sigma_x$  Microscopic cross-section for reaction  $x$  [ $\text{m}^2$ ]  
 $k_{\text{eff}}$  Multiplication factor [1]  
 $\nu$  Neutron fission yield [1]  
 $\phi$  Scalar neutron flux [ $\text{m}^{-2}\text{s}^{-1}$ ]  
 $P_N$  Power density [ $\text{Wm}^{-3}$ ]  
 $\Sigma_T$  Total macroscopic cross-section [ $\text{m}^{-1}$ ]  
 $H^u$  Upscattering source term [ $\text{m}^{-3}\text{s}^{-1}$ ]

### Thermal-hydraulics

- $\Psi$  Compressibility [ $\text{Pa}^{-1}$ ]  
 $\rho$  Density [ $\text{kgm}^{-3}$ ]  
 $\epsilon$  Dissipation of turbulent kinetic energy [ $\text{m}^2\text{s}^{-3}$ ]  
 $\nu$  Frequency [Hz]  
 $g$  Gravitational acceleration [ $-9.81\text{ms}^{-2}\hat{z}$ ]

$\varphi$  Mass flux [ $\text{kgm}^{-2}\text{s}^{-1}$ ] (generally  $\rho U$ )  
 $p$  Instantaneous pressure [Pa]  
 $u$  Instantaneous velocity [ $\text{ms}^{-1}$ ]  
 $h$  Specific enthalpy [J/kg]  
 $c_p$  Specific heat capacity at constant pressure [ $\text{JK}^{-1}\text{kg}^{-1}$ ]  
 $\tau$  Stress tensor [ $\text{Nm}^{-1}$ ]  
 $q''$  Surface heat flux [ $\text{Jm}^{-2}$ ]  
 $q'''$  Volumetric heat source [ $\text{Jm}^{-3}$ ]  
 $T$  Temperature [K]  
 $K$  Thermal conductivity [ $\text{WK}^{-1}\text{m}^{-1}$ ]  
 $\alpha$  Thermal diffusivity [ $\text{m}^2\text{s}^{-1}$ ]  
 $\beta$  Thermal expansion coefficient [ $\text{K}^{-1}$ ]  
 $k$  Turbulent kinetic energy [ $\text{m}^2\text{s}^{-2}$ ]  
 $P_r$  Turbulent Prandtl number  
 $P$  Time averaged pressure [Pa]  
 $U$  Time averaged velocity [ $\text{ms}^{-1}$ ]

## Thermal radiation

$I_b$  Black body emission intensity [ $\text{Wm}^{-2}$ ]  
 $G$  Incident radiative heat flux [ $\text{Wm}^{-2}$ ]  
 $\phi$  Inscattering function [1]  
 $\epsilon$  Radiation emissivity [1]  
 $I$  Radiation intensity [ $\text{Wm}^{-2}$ ]  
 $K_a$  Radiation absorption coefficient [ $\text{m}^{-1}$ ]  
 $K_s$  Radiation scattering coefficient [ $\text{m}^{-1}$ ]

## General

$\Omega$  Angular direction  
 $\mathbf{s}$  Direction vector, with length  $s = |\mathbf{s}|$   
 $X$  Example quantity, no physical meaning  
 $m$  General material notation  
 $\chi$  General reaction notation  
 $r$  General space coordinate  
 $t$  General time coordinate

$\delta_{ij}$  Kroenecker delta

$\varrho$  Under-relaxation factor

$V$  Volume of mesh cell [m<sup>3</sup>]

## Subscripts

**overbar** Time averaging

$X_i$  i:th component of the tensor  $X$

$X_{i,j}$  derivative of i:th component of the tensor with respect to the j:th direction of  $X$

# Abbreviations

BWR	Boiling Water Reactors
CFD	Computational Fluid Dynamics
HPC	High Performance Computing
LWR	Light Water Reactor
MOX	Mixed oxide
PDE	Partial Differential Equation
PWR	Pressurized Water Reactors
UOX	Uranium oxide



# Chapter 1

## Introduction

This first chapter gives an introduction to nuclear energy, the basic concepts of light water reactors, and the methodology applied to reactor coupled calculations. Moreover the work carried out in this thesis is outlined, including motivations, objectives and framework.

### 1.1 Nuclear Power

Electricity production from nuclear power plants is today standing for approximately 14% of the world's ever increasing demand for electricity[1]. The addressing of the environmental issues related to the release of greenhouse gases has led to a renewed interest in nuclear energy[2]. The nuclear renaissance has at late been slowed or put on hold by many countries following the Fukushima accident[3].

#### 1.1.1 Light Water Reactors

Most of the existing 436 nuclear reactors[4] are Light Water Reactors (LWRs). A LWR is based on the boiling of natural water through the use of fission of primarily uranium. The steam drives a turbine, connected to an electric generator, producing electricity to the grid. The variations of commercial LWR:s is primarily seen in the method used to boil the water.

In the Pressurized Water Reactor (PWR) the thermal-hydraulic state in the core is such that boiling is not occurring. The boiling is instead happening in a second loop (the secondary side), in the steam generators. The hot and highly pressurized flow of water in the primary loop, heated by the nuclear fission, is driven primarily by pumps. An outline can be seen in Fig. 1.1.

In the Boiling Water Reactor (BWR) the water is let boil immediately in the core of the reactor. Consequently, this is a direct cycle system. The steam produced in the core is let pass a steam separator and steam dryer to increase the quality of the steam (i.e. reducing the amount of liquid in the steam), to improve turbine performance and avoid possible issues (e.g. erosion damage). An outline can be seen in Fig. 1.2. The boiling occurs inhomogeneously, and the amount of vapor in the steam will increase with increasing axial elevation.

In these kind of nuclear power plants, water is not only the coolant, but it works also as moderator of the neutrons. When uranium-235 undergoes fission, a number of fast neutrons (i.e. with relatively high energies) are released. These neutrons can be used to maintain a nuclear chain reaction, so that power can be produced stationarily. However, since the uranium-235 enrichment in LWR fuel is relatively low, and the uranium-235 fission cross-section with fast neutrons is much smaller than for thermal neutrons, a moderator is needed to accomplish an efficient fission rate in the core.

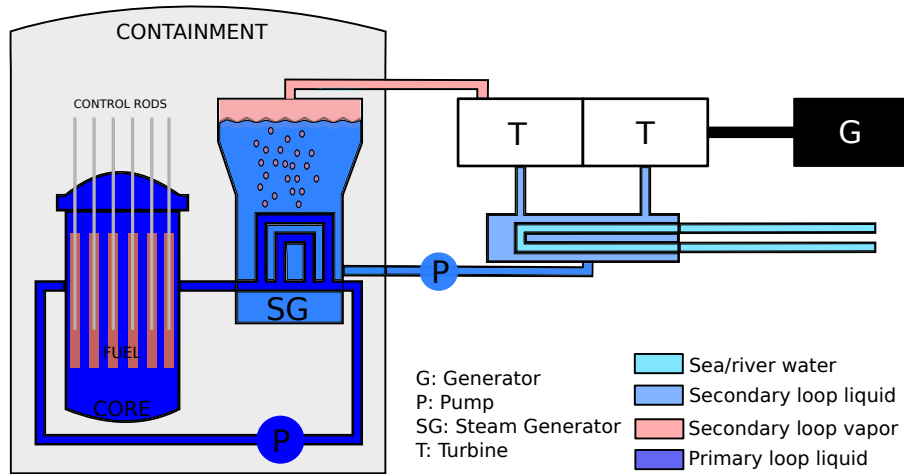


Figure 1.1: Outline of a PWR. Partly recreated from [5].

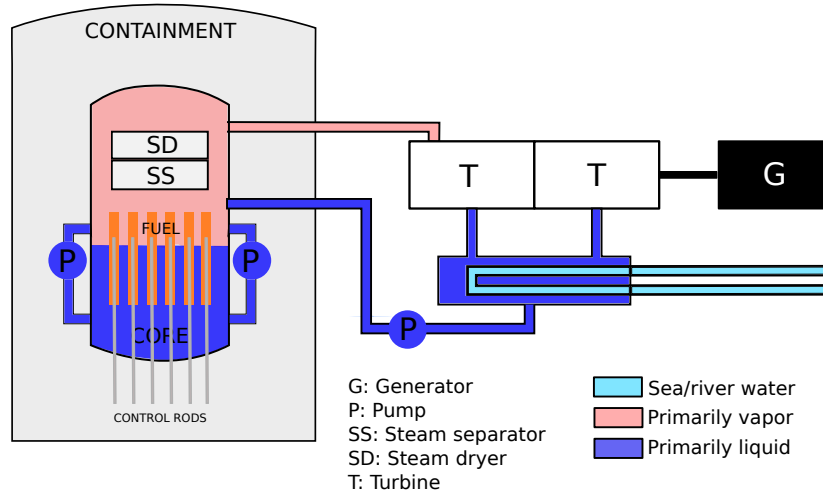


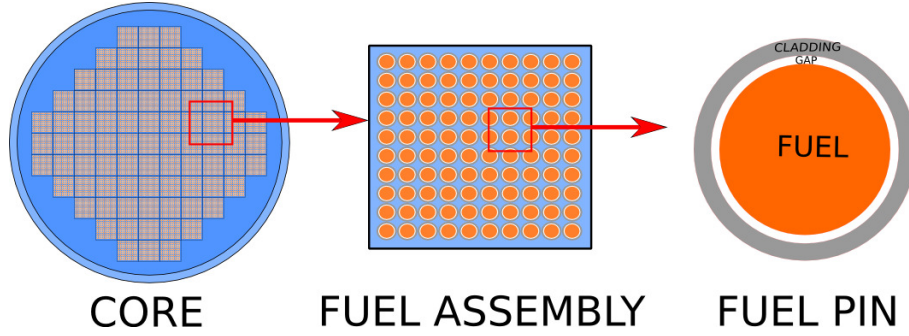
Figure 1.2: Outline of a BWR. Partly recreated from [5, 6].

### 1.1.2 LWR fuel structure

PWR and BWR nuclear fuel share common characteristics. A rectangular lattice of fuel assemblies is placed in the core of the reactor. The fuel assembly in turn consists of a rectangular lattice of fuel pins. The pins consists of piled pellets of nuclear fuel. An outline of the radial structure of a typical LWR core can be seen in Fig. 1.3.

The radial length scale of the fuel pin is on the order of 1 cm, whereas the radial length scale of the fuel bundle is on the order of 10-20 cm (varying among models and BWR and PWR fuel). The height of the stacked fuel pellets is on the order of 3-4 m. The fuel is centered in the fuel pin, with a thin surrounding gap consisting of helium when fuel is fresh, and when burned also gaseous fission products. The gap allows for thermal expansion of the fuel and a free volume for the gaseous fission products.

The cladding is the first safety barrier of the nuclear fuel, acting as a container for the fuel. The cladding composition may vary, but usually consists of some zirconium based alloy.

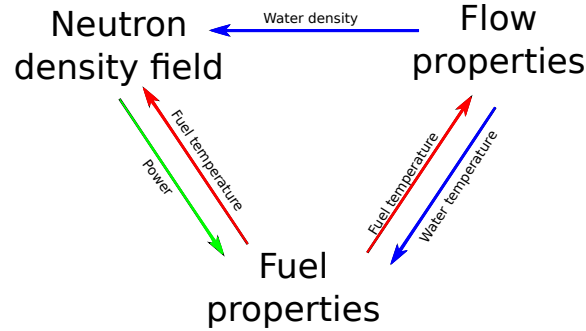


**Figure 1.3:** Outline of the radial structure of simplified LWR rectangular fuel.

## 1.2 Basic physics of a Light Water Reactor core

In a LWR, neutrons are thermalized in the moderator and induce fissions in the nuclear fuel. The energy released by fission is transferred from the fuel, through the cladding, to the coolant. The flow of the coolant removes the heat from the core. Therefore, to study the behavior of the core, knowledge of neutron kinetics, heat transfer mechanisms, fluid dynamics, among other phenomena, is essential.

Furthermore, since the neutron density (and, thus, the power) is affected by the fluid conditions and the fluid conditions depend on the power (i.e. the neutron density) it is important to take in account the interdependence between the different fields of physics. This is summarized in the schematics in Fig. 1.4.



**Figure 1.4:** Outline of the interdependence of the different fields of physics in the nuclear reactor core of a LWR reactor.

## 1.3 Review of current methodologies for reactor calculations and motivation of the work

In order to provide realistic simulation of the behavior of a LWR core, the coupling between neutronics and thermal-hydraulics must be modeled. For transient calculations the usual procedure is to couple a 3D neutronic core simulator (e.g. PARCS[7]) and a 1D thermal-hydraulic system code (e.g. RELAP5[8]). The coupled transient core and thermal-hydraulic codes methodology has been applied extensively and successfully, although it can suffer some issues including computational inefficiency and convergence problems[9]. For steady state core simulations, a simplified thermal-hydraulic model is often included in the core simulator (e.g. SIMULATE-3[10]). More details can be found in [11, 12, 13].

Considering the above description it is seen that in none of the outlined computational methodologies are the different fields of physics directly treated in an explicit, three dimensional

manner. Reasons for this include:

- computational burden: explicit calculations of full dimensionality of all fields will be more computationally heavy than performing divided calculations, considering only a part of the problem and using lower dimensional simplified calculations or correlations for the other fields.
- historical reasons: the methodologies discussed above were proposed at a stage where the computational power was much smaller. The problems had to be divided to give feasible running times. The development during the 1990s was more to the use of workstations and less to use high performance computers, and this limited the computational power[14].
- methodological reasons: present high performance computing (HPC) is based on massively parallel computing (see e.g. [15]). The nuclear reactor codes, with their developed methodologies, are often aimed to be run on a single CPU, making the extensions to modern HPC machines more difficult.
- complexity reason: as different fields of physics require different skills the use of multi-physics has for sure often been limited by few scientists being experts in many fields. The thermal-hydraulics calculations have developed through the computational fluid dynamics (CFD), today with little influence from the reactor physics code developers. This naturally leads to separate methodologies, and difficulty in combining the tools.

Late developments have shown more effort put into the use of super computers for nuclear engineering calculations, allowing for higher accuracy of the now performed calculations, new possibilities of better models and new combinations of physics.

An example of an application in which a 1D thermal-hydraulic approximation can be non-satisfactory is reactor physics calculations for modern BWR fuel assemblies. Previous works showed that such an approach can give inaccurate results because the radial void fraction distribution of the moderator can play a non-negligible role[16, 17]. Therefore, the use of CFD capabilities, allowing a better description of the moderator properties, together with neutronics looks of particular interest.

In this light, late work has shown examples of external coupling methodologies using commercial CFD-software combined with existing neutronic solvers[18, 19, 20, 21]. External coupling softwares, exchanging the coupled fields, were in all cases used, limiting the performance and applicable multi-physics methodologies in the codes.

## 1.4 Description of this work

The purpose of this work is to analyze the feasibility of implementing a steady-state neutron diffusion and thermal-hydraulics coupled methodology into an existing CFD tool. Many aspects of this work are connected to the development of the code itself, aiming to answer:

- What methodology of physics coupling should be applied?
- What is the computational effort?
- What background data and parameters are needed?

Successfully managing to implement such a CFD based code, other questions related to the developed methodology could be addressed:

- How high-resolution is needed to correctly model the interdependence of the different fields of physics?
- How strong is the coupling between the different fields of physics?

- In what kind of applications is such a code beneficial or even necessary?

The methodology developed is based on a computational tool that solves the steady-state multi-group neutron diffusion equation coupled with a CFD solver for the thermal-hydraulics that is built from solvers available in the open source CFD software OpenFOAM[22, 23]. The coupling is internal, so no external software is needed to transfer data from the neutronic field to the thermal-hydraulic field, and vice versa. The numerical algorithm relies on features (e.g. computational mesh generation, operator discretization, matrix solvers, etc.) that are also available in OpenFOAM. Neutron cross-sections are pre-calculated by following a procedure that use the nuclear Monte Carlo Code SERPENT[24, 25].

To test the developed tool, the case of a simplified fuel assembly under PWR conditions was studied. Partial fuel assemblies with multiple different pin compositions were tested, with the size of the system primarily limited by the CPU-usage.

## 1.5 Structure of the thesis

After this introductory chapter, a detailed account of the theory will follow in Chapter 2. The methodology developed in this Master of Science thesis project is described in Chapter 3, and the demonstration of its applicability is illustrated in Chapter 4. Finally, conclusions are drawn and future work is discussed in Chapter 5.



# Chapter 2

## Theory

The methodology studied in this work has at its core a newly developed computational tool for simulating the coupling between the neutronics and the thermal-hydraulics of a single-phase flow in a nuclear fuel assembly. This chapter gives the theoretical background on which such a computational tool is based. First the neutron multi-group neutron diffusion equation is derived (section 2.1). Then the thermal-hydraulic model is discussed (section 2.2). Finally the coupled problem is illustrated (section 2.3).

### 2.1 Neutronics

The aim of this section is to formulate the multi-group neutron diffusion equation and to illustrate the related assumptions and approximations. Furthermore the cross-sections and the neutron parameters needed in such an equation will be defined and discussed.

#### 2.1.1 Neutron transport equation

By considerations of the processes involving neutrons in a nuclear reactor core the following integro-differential transport balance equation can be written[6]:

$$\begin{aligned} \frac{\partial}{\partial t} n(r, \Omega, E, t) + \Omega \cdot \nabla \Psi(r, \Omega, E, t) + \Sigma_T(r, E, t) \Psi(r, \Omega, E, t) = \\ \int_{(4\pi)} \int_0^\infty \Sigma_S(r, \Omega' \rightarrow \Omega, E' \rightarrow E, t) \Psi(r, \Omega', E', t) d\omega' dE' + \frac{\chi(E)}{4\pi} \int_0^\infty \nu \Sigma_f(r, E', t) \Phi(r, E', t) dE' \end{aligned} \quad (2.1)$$

where the scalar and angular flux are related by:

$$\Phi(r, E, t) = \int_{(4\pi)} \Psi(r, \Omega', E, t) d\omega' \quad (2.2)$$

Equation (2.1) is not generally possible to solve analytically. For the steady state limitation of this thesis work, the time dependence is excluded. The system is instead rewritten using the effective multiplication factor,  $k_{\text{eff}}$  (i.e. the ratio of the number of neutrons in one generation to the number of neutrons in the previous one). Defining the neutron current as:

$$J(r, E, t) = \int_{(4\pi)} \Omega \cdot \Psi(r, \Omega, E, t) d\omega, \quad (2.3)$$

and integrating over the angular space, eq. (2.1) will then become:

$$\begin{aligned} \nabla \cdot J(r, E) + \Sigma_T(r, E)\Phi(r, E) = \\ \int_{(4\pi)} \int_{(4\pi)} \int_0^\infty \Sigma_S(r, \Omega' \rightarrow \Omega, E' \rightarrow E)\Psi(r, \Omega', E')d\omega d\omega' dE' + \\ \frac{\chi(E)}{k_{\text{eff}}} \int_0^\infty \nu \Sigma_f(r, E')\Phi(r, E')dE' \end{aligned} \quad (2.4)$$

The effective multiplication factor is to be seen as a renormalizing factor. For  $k_{\text{eff}} = 1$  the system is said to be critical. The number of neutrons born by fission and absorbed and leaking are then at balance. The renormalization of the fission term allows determining the general behavior of the nuclear reactor without solving the time-dependent problem, eq. (2.1).

The first approximation introduced is to assume that the neutrons are isotropically scattered, removing the angular dependence such that:

$$\begin{aligned} \nabla \cdot J(r, E) + \Sigma_T(r, E)\Phi(r, E) = \\ \int_0^\infty \Sigma_{s0}(r, E' \rightarrow E)\Phi(r, E')dE' + \frac{\chi(E)}{k_{\text{eff}}} \int_0^\infty \nu \Sigma_f(r, E')\Phi(r, E')dE' \end{aligned} \quad (2.5)$$

where the approximation of isotropic scattering enters as:

$$\Sigma_S(r, \Omega' \rightarrow \Omega, E' \rightarrow E) = \frac{\Sigma_{s0}(r, E' \rightarrow E)}{4\pi} \quad (2.6)$$

Do note that other approaches to solve eq. (2.1) could be utilized, partially or completely, preserving the angular dependence.

### 2.1.2 Diffusion approximation

The solution of eq. (2.5) can be obtained by introducing a relation between the scalar neutron flux ( $\Phi$ ) and the neutron current ( $J$ ). One possible approximative way to relate the neutron current and neutron scalar flux is through the use of Fick's law[6]:

$$J(r, E) \approx -D(r, E)\nabla\Phi(r, E) \quad (2.7)$$

This gives the neutron diffusion equation, i.e.:

$$\begin{aligned} -\nabla(D(r, E)\nabla\Phi(r, E)) + \Sigma_T(r, E)\Phi(r, E) = \\ \int_0^\infty \Sigma_{s0}(r, E' \rightarrow E)\Phi(r, E')dE' + \frac{\chi(E)}{k_{\text{eff}}} \int_0^\infty \nu \Sigma_f(r, E')\Phi(r, E')dE' \end{aligned} \quad (2.8)$$

The diffusion coefficient,  $D$ , is estimated through transport theory, retaining some degree of angular dependence, as[6]:

$$D(r, E) = \frac{1}{3(\Sigma_T(r, E) - \mu(r)\Sigma_{s0}(r))} \quad (2.9)$$

where:

$$\Sigma_T = \Sigma_a + \Sigma_{s0} \quad (2.10)$$

Equation (2.8) is an eigenvalue problem in which the effective multiplication factor will correspond to an eigenvalue of the system, and the neutron flux,  $\Phi(r, E)$  an eigenstate of the system.

In order to solve (2.8) in a computational manner, the energy and space dependence must be cast to discrete relations.



### 2.1.3 Discrete energy equation

The energy-dependence is split into a discrete set of energy intervals such that [6]:

$$[E_{\min}, E_{\max}] = \prod_{g=1}^G [E_g, E_{g-1}] \quad (2.11)$$

with the lowest neutron energy group labeled G and the highest neutron energy group labeled 1. Eq. (2.8) will, in its discrete energy dependence, be:

$$-\nabla (D_g(r) \nabla \Phi_g(r)) + \Sigma_{T,g}(r) \Phi_g(r) = \sum_{g'=1}^G \Sigma_{s0,g' \rightarrow g}(r) \Phi_{g'}(r) + \frac{\chi_g^p}{k_{\text{eff}}} \sum_{g'=1}^G \nu \Sigma_{f,g'}(r) \Phi_{g'}(r) \quad (2.12)$$

where, in order to be consistent with eq. (2.8), the multi-group quantities are defined as:

$$\Phi_g(r) = \int_{E_g}^{E_{g-1}} \Phi(r, E) dE \quad (2.13)$$

$$\Sigma_{T,g}(r) = \frac{\int_{E_g}^{E_{g-1}} \Sigma_T(r, E) \Phi(r, E) dE}{\int_{E_g}^{E_{g-1}} \Phi(r, E) dE} \quad (2.14)$$

$$[\nu \Sigma_{f,g}](r) = \frac{\int_{E_g}^{E_{g-1}} [\nu \Sigma_f](r, E) \Phi(r, E) dE}{\int_{E_g}^{E_{g-1}} \Phi(r, E) dE} \quad (2.15)$$

$$\Sigma_{s0,g' \rightarrow g}(r) = \frac{\int_{g'}^{g'-1} \int_{E_g}^{E_{g-1}} \Sigma_{s0}(r, E', E) \Phi(r, E) dE dE'}{\int_{E_g}^{E_{g-1}} \Phi(r, E) dE} \quad (2.16)$$

$$\chi_g = \int_{E_g}^{E_{g-1}} \chi(E) dE \quad (2.17)$$

$$D_g = \frac{\int_{E_g}^{E_{g-1}} D(r, E) \|\nabla (\Phi(r, E))\| dE}{\int_{E_g}^{E_{g-1}} \|\nabla (\Phi(r, E))\| dE} \quad (2.18)$$

The last equation can be approximated using eq. (2.9):

$$D_g(r) = \frac{1}{3 (\Sigma_{T,g}(r) - \mu_g(r) \Sigma_{s0,g}(r))} \quad (2.19)$$

with:

$$\Sigma_{s0,g}(r) = \sum_{g'=1}^G \Sigma_{s0,g \rightarrow g'}(r) \quad (2.20)$$

To compute the multi-group neutron flux  $\Phi_g(r)$  the quantities (2.14)-(2.18) need to be pre-determined, either through calculations or measurements.

### 2.1.4 Space homogenization

The discrete energy diffusion equation, eq. (2.12), must be discretized in space to give a finite number of equations to be solved. This can be done in many ways, including finite elements, finite differences and finite volumes. The last method is the one used in this thesis.

The space is split in  $I$  homogeneous cells (volumes), and the diffusion equation (2.12) is solved for each of these cells, with proper boundary conditions that take in account the flux in the adjacent volumes. Therefore, the diffusion equation for the generic  $i$ -th cell, with  $i = 1, \dots, I$ , can be written:

$$-\nabla(D_{i,g}\nabla(\Phi_{i,g})) + \Sigma_{T,i,g}\Phi_{i,g} = \sum_{g'=1}^G \Sigma_{s0,i,g' \rightarrow g}\Phi_{i,g'} + \frac{\chi_g^p}{k_{\text{eff}}} \sum_{g'=1}^G \nu\Sigma_{f,i,g'}\Phi_{i,g'} \quad (2.21)$$

The interdependence between different cells will be found in the discretization of the streaming operator,  $\nabla$ . The space dependence will be further discussed in the methodology section (see section 3.1.2).

### 2.1.5 Concerning the choice of diffusion approximation

The neutron diffusion methodology is often applied in 3D core calculations, whereas 2D lattice calculations often utilize neutron transport methodologies, e.g. discrete ordinates method [13], retaining higher order angular dependence.

The methodology developed and illustrated in this thesis is addressed to study scales that are comparable with the lattice calculation cases. Nevertheless, the diffusion approximation was used, primarily, because it is easier to implement as a test of the newly proposed coupling methodology.

It must be noted that when spatial material heterogeneities are present, eq. (2.7) can give poor results. It could therefore, in accordance with the small scale of the problem solved, be beneficial to implement some angular dependent method.

### 2.1.6 Cross-sections

In order to solve eq. (2.21), the macroscopic discrete cross-sections and the diffusion coefficients need to be determined. In the continuous space and energy formalism the macroscopic cross-sections are given by:

$$\Sigma_{\chi}(r, E, t) = \sum_{m=1}^M \sigma_{\chi,m}(E)N_m(r) \quad (2.22)$$

As shown in eq. (2.22), the macroscopic cross-sections depend on energy, position and time. These dependencies underlie other dependencies, as:

- on fuel temperature
- on fuel burnout
- on moderator thermal-hydraulic conditions
- on control rod positioning
- on poisoning

Furthermore, as shown in eq. (2.14)-(2.18), for the condensed cross-sections the flux is used as a weight function. This is somewhat a contradiction since the cross-sections are used to compute the flux. It is thus necessary to compute the homogenized and condensed cross-sections with a reasonable approximation of the flux.

The spatial resolution of the macroscopic cross-sections that is needed for the current work is relatively high, since the goal is to implement the neutron diffusion equation in a CFD tool that makes use of a fine spatial mesh for the solution of the thermal-hydraulic equations. The generation of the macroscopic cross-sections can be achieved with deterministic lattices codes or with Monte Carlo based codes.

Deterministic lattice codes estimate the macroscopic cross-sections by solving first the flux in a micro-energy group structure and in a micro-region structure. Then, such a result is condensed and homogenized via a macro-group structure and a macro-region structure to a final few-group (typically two-group) cross-sections at fuel assembly level. Therefore, data at the fuel pin level, that are necessary in this work, are usually not available from calculations performed with existing lattice codes.

On the other hand, Monte Carlo nuclear codes allow to model problem with arbitrary geometries and material composition, and the energy dependency can be treated as a continuum. Thus, energy condensation and spatial homogenization can be derived in a more straightforward manner. For this reason the Monte Carlo nuclear code SERPENT was chosen to create the macroscopic cross-sections for the case under study in this thesis. The code has also the advantage to be based on a collision estimator which makes group constants calculations easier than, for instance, the case of the code MCNP5[26]<sup>1</sup>.

---

<sup>1</sup>In MCNP tally calculations group to group cross-sections are not readily available. An adjusted procedure was applied to overcome this issue, but it was found inefficient.

## 2.2 Thermal-hydraulics

In this section a model for single-phase turbulent flow of a compressible fluid is presented. In addition, a heat transfer model for solids is discussed. This is focused on the specific case of applications for nuclear fuel assemblies.

### 2.2.1 Governing equations

To study the behavior of a fluid flow, conservation equations for mass, momentum and energy are introduced[27]:

$$\frac{\partial \rho}{\partial t}(r, t) + \nabla \cdot (\rho u)(r, t) = 0 \quad (2.23)$$

$$\frac{\partial(\rho u)}{\partial t}(r, t) + \nabla \cdot (\rho u \otimes u)(r, t) = \nabla \cdot \tau(r, t) - \nabla \cdot (p)(r, t) + \rho(r, t)g \quad (2.24)$$

$$\frac{\partial(\rho h)}{\partial t}(r, t) + \nabla \cdot (\rho u h)(r, t) = -\nabla \cdot q''(r, t) + q'''(r, t) + \tau(r, t) \nabla \otimes u(r, t) + \nabla \cdot (u p)(r, t) \quad (2.25)$$

For Newtonian fluids, the stress tensor  $\tau$  can be written as:

$$\tau = \mu \left( u_{i,j} + u_{j,i} - \frac{2}{3} u_{k,k} \delta_{ij} \right) \quad (2.26)$$

where the repeated indices imply summing, and the comma partial derivatives.

### 2.2.2 Time averaging

The above equations model the local instantaneous flow of a fluid, and their solution is very challenging, both in terms of modeling and computational effort. For practical purposes, an approach that is used to simplify the problem is to average such equations over proper time intervals and spatial volumes. However, this procedure introduces a certain degree of approximation since relatively high-frequency and small scales phenomena are filtered out. The space averaging is discussed in the methodology section (see section 3.1.2).

The time-average of the conservation equations (2.23)-(2.25) is based on Reynolds decomposition. Accordingly, a general quantity  $x(r, t)$  describing one of the properties of the fluid flow can be considered as composed of a mean value and a fluctuating component, i.e.:

$$x(t) = X(t) + x'(t) \quad (2.27)$$

where the averaged part is defined as:

$$X = \int_t^{t+\delta t} x(t') dt' \quad (2.28)$$

and the fluctuating part  $x'(t)$  has the property:

$$\int_t^{t+\delta t} x'(t') dt' = 0 \quad (2.29)$$

where  $\delta t$  is a time interval suitable to filter out small scale fluctuations, but retaining slower variations.

Thus the velocity, pressure and enthalpy of a fluid can be expressed as:

$$u(r, t) = U(r, t) + u'(r, t) \quad (2.30)$$

$$h(r, t) = H(r, t) + h'(r, t) \quad (2.31)$$

$$p(r, t) = P(r, t) + p'(r, t) \quad (2.32)$$

where  $u'(r, t)$ ,  $h'(r, t)$  and  $p'(r, t)$  are the fluctuating parts.

Substituting eqs. (2.30)-(2.32) into (2.23)-(2.25) and averaging the resulting equations on the same time interval  $[t, t + \delta t]$ , a time averaged form of the conservation equations is obtained:

$$\frac{\partial \rho}{\partial t}(r, t) + \nabla (\rho U)(r, t) = 0 \quad (2.33)$$

$$\frac{\partial(\rho U)}{\partial t}(r, t) + \nabla (\varphi \otimes U)(r, t) = \nabla \cdot \tau(r, t) - \nabla \cdot \overline{\rho u' \otimes u'}(r, t) - \nabla (P)(r) + \rho(r, t)g \quad (2.34)$$

$$\begin{aligned} \frac{\partial(\rho H)}{\partial t}(r, t) + \nabla (\varphi H)(r, t) = \\ -\nabla \cdot \overline{q''}(r, t) + \overline{q'''}(r, t) + \nabla \cdot (UP)(r, t) + \overline{\nabla \cdot (u'p')}(r, t) + \\ \tau(r, t)\nabla \otimes U(r, t) - \overline{\nabla \cdot (\rho u h')} \end{aligned} \quad (2.35)$$

where:

$$\varphi(r, t) = (\rho U)(r, t) \quad (2.36)$$

Do note that only the fluctuating contributions from velocity, enthalpy and pressure have been included, assuming the fluctuations in density to be negligible. This can be shown to be valid for speeds much lower than the speed of sound of the medium[28], which is true for the present calculations.

In the case of steady state conditions, eqs. (2.33)-(2.35) becomes:

$$\nabla (\rho U)(r) = 0 \quad (2.37)$$

$$\nabla (\varphi \otimes U)(r) = \nabla \cdot \tau(r) - \nabla \cdot \overline{\rho u' \otimes u'}(r) - \nabla (P)(r) + \rho(r)g \quad (2.38)$$

$$\nabla (\varphi H)(r) = -\nabla \cdot \overline{q''}(r) + \overline{q'''}(r) + \nabla \cdot (UP)(r) + \overline{\nabla \cdot (u'p')}(r) + \tau(r, t)\nabla \otimes U(r, t) - \overline{\nabla \cdot (\rho u h')} \quad (2.39)$$

In eqs. (2.38) and (2.39) it can be seen that the fluctuations in velocity, pressure and enthalpy affect the momentum and the energy of the fluid. Such contributions are related to the possible turbulence of the flow.

In order to solve eqs. (2.37)-(2.39) with respect to  $U, P$  and  $H$ , additional relationships for the turbulent components, the heat flux  $q''$  and the volumetric heat source  $q'''$  are needed<sup>2</sup>.

As is described in section 2.2.3 the  $k - \epsilon$ -model can be used for the turbulent term  $\nabla \overline{\rho u' \otimes u'}$  in eq. (2.38), whereas the term  $\nabla \overline{\rho u' h'}$  in eq. (2.39) can be modeled as a further contribution of the turbulent flows to the total thermal conductivity. Equation (2.39) contains a second term due to the fluctuations, i.e.  $\nabla \overline{u'p'}$  that can be assumed to be negligible. Besides, proper models for the heat flux and the volumetric heat source  $q'''$  are discussed in subsection 2.2.4.

<sup>2</sup>The density is determined from thermophysical state function, such that  $\rho = f(P, H)$  (see section 4.1).

### 2.2.3 Turbulence modeling

#### Momentum equation

In order to close eq. (2.38), the so called Reynold stress tensor  $\overline{u' \otimes u'}$ , or in tensor notation  $\overline{u'_i u'_j}$  needs to be determined. Different models for the Reynold stress tensor are available, and they can be based on algebraic, one-equation, or two equation approaches[29]. In the present work, the  $k - \epsilon$  two-equation transport model is used.

Thus, in order to relate the velocity fluctuations to the Reynold averaged flow quantities, the Boussinesq assumption is utilized [29, 28]:

$$\overline{\rho u'_i u'_j} = \mu \left( U_{i,j} + U_{j,i} - \frac{2}{3} U_{k,k} \delta_{i,j} \right) - \frac{2}{3} \rho k \delta_{i,j} \quad (2.40)$$

Boussinesq assumption introduces the new quantity  $k$ , the turbulent kinetic energy, defined by:

$$k = \frac{1}{2} \overline{u'_i u'_i} \quad (2.41)$$

This quantity could be evaluated with the following transport equation:

$$\nabla (\rho k U) = \nabla \left( \frac{\mu_t}{\sigma_k} \nabla k \right) + 2 \mu_t U_{i,j} U_{j,i} - \rho \epsilon \quad (2.42)$$

where  $\epsilon$  is the dissipation of turbulent kinetic energy satisfying a second transport equation:

$$\nabla (\rho \epsilon U) = \nabla \left( \frac{\mu_t}{\sigma_\epsilon} \nabla \epsilon \right) + C_{1\epsilon} \frac{\epsilon}{k} 2 \mu_t U_{i,j} U_{j,i} - C_{2\epsilon} \rho \frac{\epsilon^2}{k} \quad (2.43)$$

with:

$$\mu_t = \rho C_\mu \frac{k^2}{\epsilon} \quad (2.44)$$

and where  $C_x$  are model constants.

#### Enthalpy equation

In analogy to the Boussinesq assumption it is assumed that the fluctuating term  $\overline{\rho u' h'}$  is proportional to the gradient of the enthalpy[28]:

$$-\overline{\nabla \cdot \rho u' h'}(r) = \nabla (\alpha_t \nabla h) \quad (2.45)$$

where  $\alpha_t$  is the turbulent diffusivity. Such turbulent diffusivity is assumed to behave similarly to  $\mu_t$ , the turbulent dynamic viscosity and it can be calculated as:

$$\alpha_t = \frac{\mu_t}{P_r} \quad (2.46)$$

where  $P_r$  is the turbulent Prandtl number. The turbulent diffusivity can be considered an additional contribution to the thermal diffusivity  $\alpha$ , so that an effective diffusivity can be defined as:

$$\alpha_{\text{eff}} = \alpha + \alpha_t \quad (2.47)$$

The thermal diffusivity is related to the thermal conductivity  $K$  as:

$$\alpha = \frac{K}{\rho c_p} \quad (2.48)$$

In a similar manner, one can express the effective thermal conductivity as:

$$K_{\text{eff}} = K + K_t = K + \alpha_t \rho c_p = K + \frac{\mu_t \rho c_p}{P_r} \quad (2.49)$$

The thermal conductivity will thus be enhanced by the turbulence of the flow.

### 2.2.4 Fluid energy equation

As mentioned above, it is necessary to model the surface heat flux  $\overline{q''}$  and the volumetric heat source  $\overline{q'''}$  appearing in eq. (2.39). In this section, it will be seen how it is convenient to arrange this equation in terms of temperature, instead of enthalpy.

To generally model the surface heat flux  $\overline{q''}$ , Fourier's law of conduction is applied, i.e.:

$$\overline{q''}(r) \approx -K(r, T)\nabla T(r) \quad (2.50)$$

For the volumetric heat source in the case of the fluid flows of the nuclear reactor fuel assembly, the only contribution will come from thermal radiation:

$$\overline{q'''}(r) = \overline{q'''}_{\text{rad}}(r). \quad (2.51)$$

The model for the thermal radiation is described below.

#### Thermal radiation

In the same manner as for the neutron transport theory a general transport equation could be written for the the radiation intensity,  $I(\nu, r, \Omega)$  [30]:

$$\frac{dI(\nu, r, \mathbf{s})}{ds} = -(K_a(\nu, r) + K_s(\nu, r))I(\nu, r, \mathbf{s}) + K_a(\nu, r)I_b(\nu, r, T) + \frac{K_s(\nu, r)}{4\pi} \int_{4\pi} I(\nu, r, \mathbf{s}')\phi(\mathbf{s} \cdot \mathbf{s}')d\omega \quad (2.52)$$

The radiative heat flux through a surface with surface normal  $\mathbf{n}$  is given by:

$$q_r(\nu, r) = \int (\mathbf{s} \cdot \mathbf{n})I(\nu, r, \mathbf{s})d\omega \quad (2.53)$$

In the current work, the solution of such a problem is obtained from the  $P_1$  approximation[6]. According to this approach the radiation intensity is assumed angular independent at any point in space and eq. (2.53) can be rewritten as:

$$q_r(\nu, r) = -\frac{1}{3(K_a(r, \nu) - K_s(r, \nu)) - C_A K_s(r, \nu)} \nabla G(\nu, r) \quad (2.54)$$

where  $C_A$  is a model constant and the incident radiation intensity,  $G$ , is defined by:

$$G(\nu, r) = \int I(\nu, r, \mathbf{s})d\omega \quad (2.55)$$

Substituting eqs. (2.54) and (2.55) in to eq. (2.52), assuming isotropic scattering, and integrating over the angular space, the following equation can be derived:

$$-\nabla \left( \frac{1}{3(K_a(\nu, r) - K_s(\nu, r)) - C_A K_s(\nu, r)} \nabla G(\nu, r) \right) = K_a(\nu, r)(4E_b(\nu, r, T) - G(\nu, r)) \quad (2.56)$$

with  $C_A$  being the linear anisotropy coefficient and  $E_b$  defined as:

$$E_b(\nu, r, T) = \pi I_b(\nu, r, T) \quad (2.57)$$

The integration of eq. (2.56) over all frequencies, yields the thermal, frequency-independent model equation:

$$-\nabla \left( \frac{1}{3(K_a(r) - K_s(r)) - C_A K_s(r)} \nabla G(r) \right) = K_a(r)(4E_b(r, T) - G(r)) \quad (2.58)$$

where  $E_b(r, T)$  can be modeled as a black-body frequency independent source,

$$E_b(r, T) = \int E_b(\nu, r, T)d\nu = \sigma_{\text{SB}}n^2(r)T^4(r) \quad (2.59)$$

with  $n$  being the refractive index and  $\sigma_{\text{SB}}$  is the Stefan-Boltzmann constant. Therefore, eq. (2.58) gives a transport of energy that must be included in the volumetric heat source of eq. (2.39), as:

$$\overline{q'''_{rad}}(r, T) = K_a(r)(G(r) - 4E_b(r, T)) \quad (2.60)$$

This transport of energy is such that the radiation energy gained by black body radiation corresponds to energy lost in the fluid, and, on the other hand, the radiative energy lost leads to an increase of fluid enthalpy.

### Energy equation in terms of temperature

Inserting eq. (2.60) into eq. (2.39) one can write an energy equation that contains both temperature and enthalpy, i.e.:

$$\nabla(\varphi h) = -\nabla(\alpha_t \nabla(h)) + \nabla(K \nabla T) + U \cdot g + K_a(G - 4E_b(T)) + \nabla(U p) \quad (2.61)$$

where friction forces and the wall frictions are neglected. From this equation it is then possible to derive an equation in terms of only temperature[6]:

$$(\rho c_p(T))U \cdot \nabla T = \beta(T)U \nabla P + \nabla(K_{\text{eff}}(T) \nabla T) + K_a(G - 4E_b(T)) \quad (2.62)$$

### 2.2.5 Solid temperature equation

The previous equation (2.62), was derived for the conditions of a fluid in the nuclear reactor core. For the solid regions no mass or momentum conservation equations needs to be considered. The energy equation for the solids will, written in terms of temperature, be[6]:

$$\rho(r, T, t)c_p(r, T, t) \frac{\partial T(r, t)}{\partial t} = \nabla(K(r, T, t) \nabla T(r, t)) + q'''(r, t) \quad (2.63)$$

The volumetric source  $q'''$  in the solid is the power released from fission, and is equal to:

$$q'''(r, t) = \gamma \int_0^\infty \Sigma_f(r, t) \Phi(r, t, E) dE \quad (2.64)$$

or according to a multi-group formalism (see section 2.1.3):

$$q'''(r, t) = \gamma \sum_{g=1}^G \Sigma_{f,g}(r, t) \Phi_g(r, t) \quad (2.65)$$

Under steady state conditions eq. (2.63) can be read as:

$$-\nabla(K(r, T) \nabla T(r)) = \gamma \sum_{g=1}^G \Sigma_{f,g}(r) \Phi_g(r) \quad (2.66)$$

In the case of non-fuel solid material, no fission occurs and thus eq. (2.63) for steady state problems can be reduced to the form:

$$-\nabla(K(r, T) \nabla T(r)) = 0 \quad (2.67)$$

Thermal radiation as well as turbulent fluctuations will have negligible effect inside the solid.



## 2.3 Coupling between neutronics and thermal-hydraulics

As already mentioned in section 1.2 the neutronic and the thermal-hydraulic fields are strongly coupled in LWRs, as the same fluid flow is employed to cool the reactor and to moderate neutrons. This interdependence can be clearly pointed out with the equations presented in this chapter.

Starting from the macroscopic cross-sections discussed in section 2.1.6 and used in eq. (2.21), one can see that they depend for instance on the energy of the system. Thus, changes in the density and in the enthalpy of the moderator/coolant fluid (i.e. quantities appearing in the conservation equations (2.37)-(2.39)) and of the fuel temperature (given by eq. (2.66)) can impact the macroscopic cross-sections, and, as a consequence, on the neutron flux. On the other hand, changes in the macroscopic cross-sections and neutron flux are reflected back to the conservation equations (2.37)-(2.39) via eq. (2.66).

The methodology developed in this work needs to take in account the multiple levels of coupling and resolve the interdependence between the different fields of physics, and is outlined in the next chapter.



# Chapter 3

## Methodology

As outlined in section 1.4, the goal of this work is to investigate the feasibility of a methodology that can be applied to high-resolution, coupled neutron/thermal-hydraulic calculations in a nuclear fuel assembly. In view of this, a methodology has been developed and follows these steps:

- Discretization of the problem (section 3.1)
  - Creation of a computational grid suitable to the problem (subsection 3.1.1)
  - Discretization of the equations describing the problem (subsection 3.1.2)
  - Definition of the boundary conditions (subsection 3.1.3)
- Generation of the neutron macroscopic parameters (section 3.2)
- Solution of the coupled neutronic/thermal-hydraulic problem (section 3.3)

The first step is necessary to solve numerically the system of equations, and it makes use of capabilities that are available with and within the open source CFD code OpenFOAM. The neutron macroscopic cross-sections needed in the neutron equations are generated according to a procedure that has been established in the current framework, and that employs the nuclear Monte Carlo code SERPENT. The solution of the multi-physics problem is then achieved with a newly developed computational tool that integrates a solution scheme for the multi-group neutron diffusion equation (2.21) together with a CFD solver for the thermal-hydraulic model derived in section 2.2. The latter is built by combining and modifying properly CFD solvers that are available, again, in OpenFOAM.

A general overview of the usage of OpenFOAM for the purpose of the current work is found in Appendix A.

### 3.1 Discretization of the problem

The equations modeling the physical system and presented in Chapter 2 need to be solved in a numerical manner. Thus, it is necessary to discretize the equations in the space of the physical system under study, which is done by splitting the space in computational cells, i.e. forming a mesh (or grid). Moreover, conditions at the boundary of the system are required.

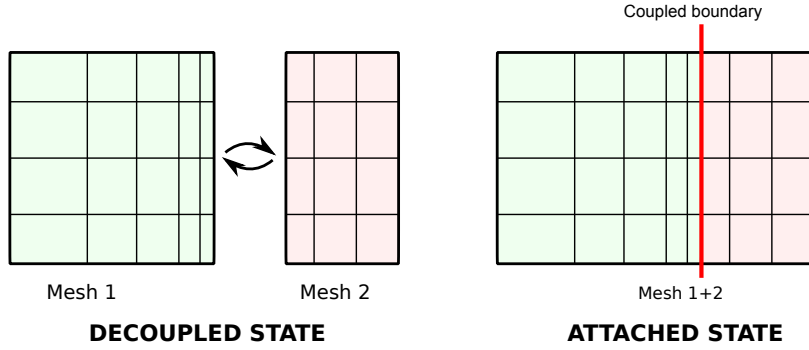
#### 3.1.1 Computational grid

The generation of the computational grid over which the equations are solved is generally a very important step in the computations. The structure and the resolution of the mesh will influence the precision, the accuracy, and the computational time of the calculation. To generate the grid, the open source platform SALOME[31] has been used. This tool can generate a mesh format imported by OpenFOAM, and can thus be used by the solver developed in the current work.

In view of this, real systems (in this case the nuclear reactor core) generally consists of different material regions and the mesh needs to fit the calculational demands specific to each region. As a consequence, the computational mesh that models a physical system is usually a combination of different meshes that are chosen to cover each region in a convenient way. Thus how to couple the different regions becomes important.

In general the field interdependence between multiple regions could be resolved in two ways: indirect and direct coupling (see Fig. 3.1). In the first one, the equations of all fields are solved in each region separately. For fields existing in neighboring regions (for this work: temperature and neutron density), the coupling between the regions would be resolved iteratively, e.g. using Neumann-Dirichlet method[32]. In the second approach, a direct coupling scheme could be used, and the equations resulting from the discretization in each mesh region would then be assembled in a single 'global' matrix. Such an approach must specifically address the coupled terms of neighboring computational cells in different mesh regions.

When multiple material regions are treated, as for the nuclear fuel pin consisting of gap, moderator, cladding and fuel, the applied equations will vary between the regions. For instance the temperature equation for the fuel region is given by (2.66), as contrasted by eq. (2.62) used in the moderator and the gap.



**Figure 3.1:** Multiple grid approaches. In the decoupled state separate matrix systems are solved, and iteratively linked by iteration. In the attached state, a single matrix system is directly solved for the composed mesh.

The coupled approach has the possible benefit of faster convergence as the region-to-region iterations are avoided. The solver implemented for this work has therefore been written to utilize this methodology (which is a feature of the development version OpenFOAM 1.6-ext[33]) for the temperature equation and the multi-group neutron flux equations. As these fields are solved for, the system is said to be in attached state. Contrary, when the regional fields, including turbulence model quantities, velocity and pressure are solved for, the mesh is considered in its separate parts, denoted as decoupled state.

### 3.1.2 Discretization of the equations

Given the mesh, the equations need to be discretized for each cell. In this work the finite volume method is utilized. Accordingly, the equations are integrated over control volumes, given by the cells of the mesh. In OpenFOAM, subroutines addressed to discretize the several terms and differential operators appearing in this thesis are available. For instance the general diffusion term  $\nabla(\Gamma \nabla X)$  (e.g. see equation (2.21)) can be discretized by a Gauss method. For the purpose of introducing the applied finite volume methodology and emphasizing the discretization schemes needed in the case of inhomogeneous  $\Gamma$ , as in the thermal conductivity in eq. (2.62) and (2.66) and the diffusion coefficient in eq. (2.21), the discretization of such an operator will be given.

Considering the one dimensional mesh in Fig. 3.2 the thermal diffusion term in eq. (2.62)

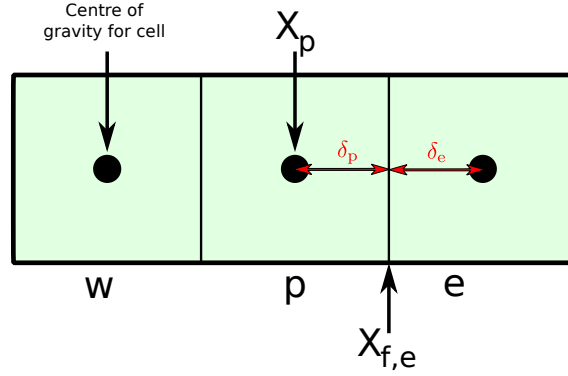


Figure 3.2: Example mesh in 1D.

integrated over the control volume given for cell  $p$  and applying Gauss theorem gives:

$$\int_{V_p} \nabla (K \nabla T) dV = \int_{S_p} K \nabla T \cdot \mathbf{n} dS = \sum_{x=\{e,w\}} (K \nabla T)_{f,x} \cdot \mathbf{n}_{f,x} A_x \quad (3.1)$$

where  $V_p$  denotes the volume of cell  $p$ ,  $S_p$  denotes the surface of cell  $p$ ,  $A_x$  a surface area,  $f$  the face value and  $\mathbf{n}$  denotes the unit surface normal. Considering only the first term of the sum in terms of Fourier's law, this term gives the heat flux at face  $e$ . To estimate the temperature gradient at the surface an approximate derivative can be formed as:

$$(\nabla T)_{f,e} \cdot \mathbf{n}_{f,e} \approx \frac{T_e - T_p}{\delta_p + \delta_e} \quad (3.2)$$

In the case of a spatially constant thermal conductivity  $K_{f,e} = K_e = K_p$ .

In this manner the integral over the control volume will be discretized in a finite number of facial terms. The same approach is followed for other terms, utilizing other schemes to perform interpolation and approximation of gradients. The convective terms, generally  $\nabla(\rho U X)$  and example in eq. (2.38), are treated with the Upwind scheme, which is first order accurate. The interpolations needed to compute e.g. the thermal conductivity at a cell face, as shown above, are performed using linear interpolation or using the harmonic scheme presented below.

### Harmonic interpolation scheme

Reconsidering the case of inhomogeneous thermal conductivity or neutron diffusion coefficient, as will be the case in this thesis, special care has to be taken to ensure conservation of energy and neutrons respectively.

In the case of two cells  $p$  and  $e$ , each with homogeneous thermal conductivity  $K_p$  and  $K_e$ , respectively, one can use a conservative interpolation based on a harmonic mean[34]:

$$K_{f,e} = \left( \frac{1 - \delta_e/(\delta_e + \delta_p)}{K_p} + \frac{\delta_e/(\delta_e + \delta_p)}{K_e} \right)^{-1} \quad (3.3)$$

This is of particular importance when the difference in thermal conductivity is very large, as in the case of the boundary between e.g. the fuel and the gap.

Furthermore it is seen that the presented harmonic interpolation compensates for non-uniform meshing, which is used in this work (see section 4.5.2). This scheme was therefore implemented in OpenFOAM, based on the structure of an existing harmonic interpolation scheme.<sup>1</sup>

<sup>1</sup>The harmonic interpolation is used in the region coupling schemes, but to the knowledge of the author the case existing corresponds to  $\delta_e = \delta_p$  (see Fig. 3.2), i.e. a uniform mesh, wherefore an additional scheme was implemented.

### 3.1.3 Boundary conditions

In order to close the space discretization, boundary conditions must be provided for all calculated quantities. For all symmetry boundaries a symmetry boundary condition is used, and it is modeled with the OpenFOAM boundary condition `symmetryPlane`. The thermophysical properties ( $\rho$ ,  $c_p$ ,  $K$  and  $\mu$ ) are generally updated, and computed as a function of the temperature. The temperature dependencies for these parameters are presented in section 4.1.

## 3.2 Group-wise macroscopic cross-sections and nuclear parameters

The group-wise macroscopic cross-sections and the neutronic diffusion coefficients defined in eqs. (2.14)-(2.18) are needed to determine the neutron density. To pre-calculate these sets of values, as discussed in section (2.1.6), the Monte Carlo nuclear code SERPENT is used.

The methodology used in the cross-section generation is based on a SERPENT case with the geometry of a single 2D fuel pin with reflective boundaries. Each specified material region is split in one or multiple regions allowing for radial resolution inside each material region. As the temperature of the material and the density of the moderator will in general vary for varying reactor power level, it is necessary to create new material specifications for each thermo-physical state desired. These temperature and density profiles are generally unknown, but the set of cross-sections evaluated in a first step can be used for reactor core calculations in turn giving better estimations of temperature and density profiles, iteratively used for new cross-section calculations. Finally, basic nuclear data (e.g. microscopic cross-sections) are needed, and can be supplied externally with one of the available nuclear libraries, such as the ENDFB-VII library[35].

Consequently, a series of SERPENT calculations need to be run, where each run gives macroscopic data for one of the regions inside one of the materials and corresponding to one thermo-physical state of interest. Accordingly, the final result is sets of values of group-wise parameters, given for each of the regions in which the physical system is split, and for temperature and density values corresponding to each simulated thermo-physical state. In this manner, it is possible to perform reactor core calculations by extracting from those sets of parameters the actual values of the group-wise parameters with respect to the position and to the physical conditions of the system.

To optimize the procedure for the generation of the group-wise cross-sections with SERPENT, a Python[36] based script was written and applied, and it executes the following steps:

1. **Parse configuration file:** read the configuration file for the run.
2. **Material adjustment:** generation of required material properties by combining the data read in the configuration file together with the data from a file with generic materials.
3. **SERPENT input:** build one SERPENT input for each geometrical region and case.
4. **Run serpent:** calculations with each of the input files are performed.
5. **Data extraction:** results are extracted from each run and combined in a MATLAB file.
6. **Cross-section generation:** output data is processed, and cross-section sets are prepared in a comma separated format read from OpenFOAM.

Typical SERPENT input files from the script (step 3) can be seen in Appendix C.

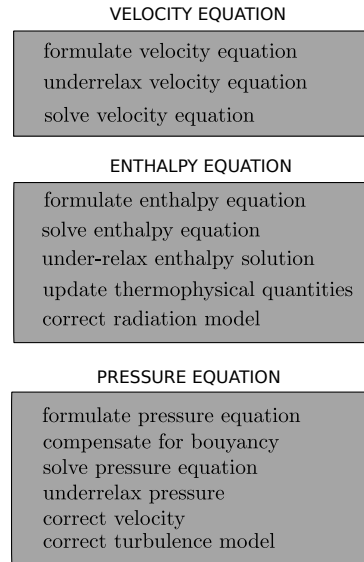
### 3.3 Solution of the coupled neutronic/thermal-hydraulic problem

As mentioned before, the core of the methodology is a computational tool that relies on an integrated algorithm where a CFD solver derived from OpenFOAM solvers is combined with multi-group neutron diffusion equation.

#### 3.3.1 Thermal-hydraulic CFD solver

##### OpenFOAM solvers

One of the existing OpenFOAM solvers was modified in order to solve the thermal-hydraulics. The solver under consideration, **buoyantSimpleRadiationFoam**, implements the steady-state laminar or turbulent flow of compressible fluids, including a separate thermal radiation heat transfer model, as explained in section 2.2. The pressure-velocity coupling in equation eq. (2.38) is solved by the SIMPLE algorithm. The energy equation is based on enthalpy of the fluid and includes radiation heat sources as in eq. (2.61).

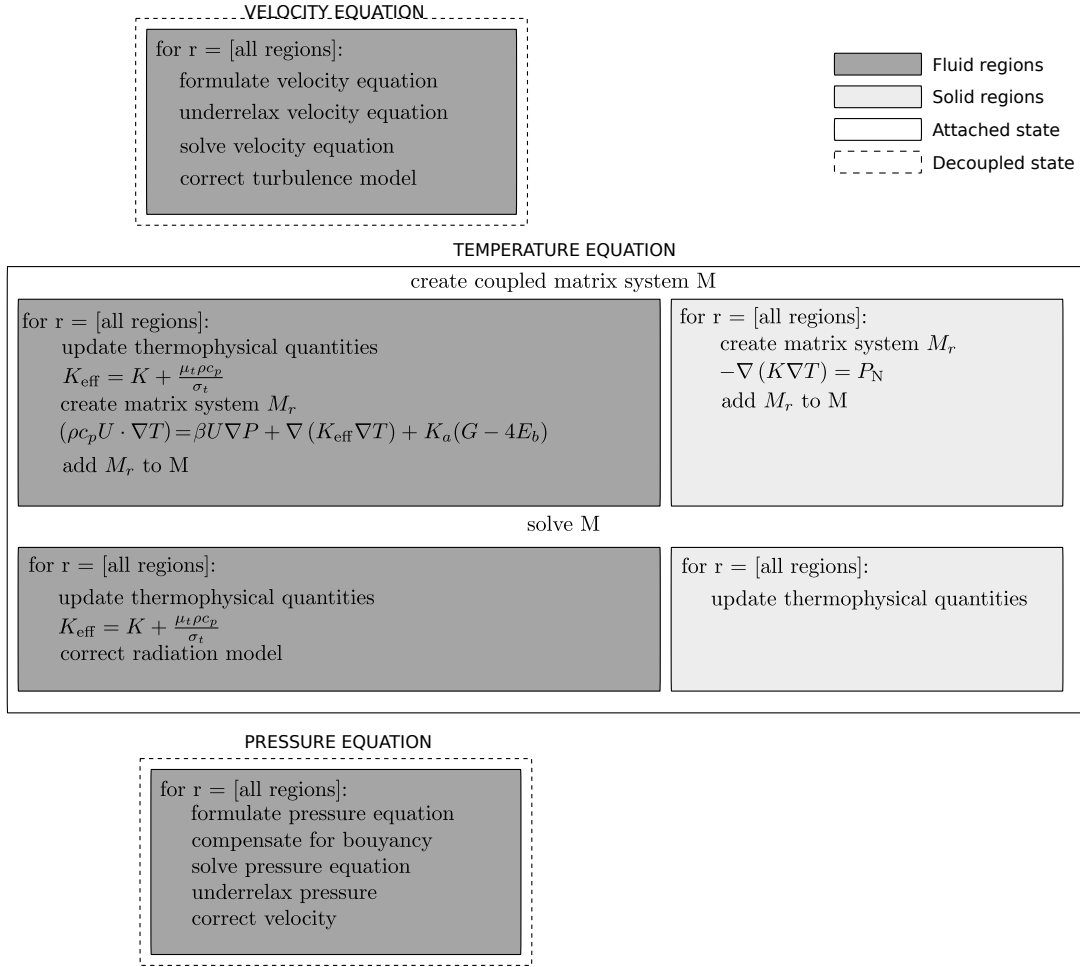


**Figure 3.3:** Overview of the applied methodology in the OpenFOAM solver **buoyantSimpleRadiationFoam**.

The general structure of the solver is seen in Fig. 3.3. A momentum predictor step, corresponding to eq. (2.38), gives an approximate velocity profile, including the turbulence term, as modeled by the  $k - \epsilon$  model, eqs. (2.42) and (2.43). The second step solves the enthalpy equation including the radiation heat sources, after which the thermophysical quantities and the radiation model, eq. (2.55), are updated. Finally, the pressure equation derived from the momentum equation is solved. An updated velocity profile is used to recompute the  $k - \epsilon$  model. No iterations are used within the thermal-hydraulics methodology.

##### Modified CFD solver

The OpenFOAM solver presented above has been modified for nuclear reactor core applications (an overview of the new version can be seen in Fig. 3.4). The enthalpy equation used in **buoyantSimpleRadiationFoam** has been removed and replaced by eqs. (2.62) and (2.66) for the fluid and solid regions respectively. No momentum or mass transfer is applied in the solid regions and is therefore discarded in these regions.



**Figure 3.4:** Overview of the applied methodology to thermal-hydraulics.

The general structure of the solver is unchanged; first a momentum predictor step, second the energy equation and third the pressure equation. The difference lies in the exchanged energy equations, including the power released by fission in the fuel regions.

Moreover, multiple connected meshes are used, which is not a standard feature in `buoyantSimpleRadiationFoam`<sup>2</sup>. The velocity and pressure equations are solved with the meshes in the decoupled state (see the dashed boxes in Fig. 3.4). In the case of this work, this must be done for the moderator and the gap regions. On the other hand, for the temperature, the mesh is used in attached state, and each matrix  $M_r$ , resulting from the discretization of the temperature equation in one specific region will be added to a global matrix  $M$ .

### 3.3.2 Implementation of the multi-group neutron diffusion equation

By making use of the general capabilities of equation discretization and solving in OpenFOAM, an algorithm for the numerical solution of the multi-group neutron diffusion equation in the form (2.21) was implemented.

Such an algorithm consists of a typical two-step procedure applied to neutronic calculations[11], and it is shown in Fig. 3.5. First, a solution is achieved for the neutron flux values at each cell in each neutron energy group. This step corresponds to computing the fission source,

<sup>2</sup>The basic idea of the coupled calculations is found in the OpenFOAM 1.6-ext specific solver `conjugateHeatTransfer`, which exemplifies the use of multiple connected meshes with different physics.

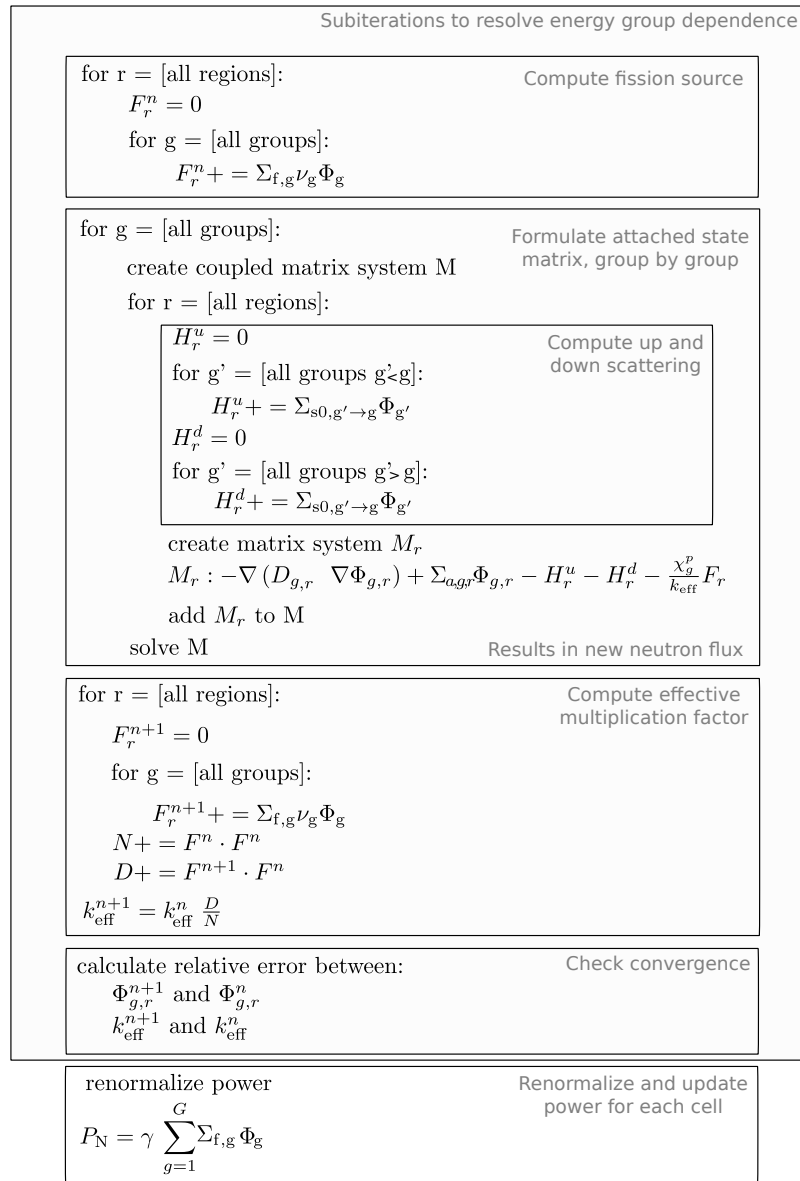


### 3.3. SOLUTION OF THE COUPLED NEUTRONIC/THERMAL-HYDRAULIC PROBLEM

the up- and downscattering, assembling a system matrix for the attached state problem, which is finally solved. This is repeated for each energy group, as indicated by the loop over all energy groups in Fig. 3.5.

Second, the eigenvalue problem for the effective multiplication factor  $k_{\text{eff}}$  is obtained from the multi-group spatial fluxes. The computed  $k_{\text{eff}}$  is used to correct the fission source term in the next iteration of the neutronics. The power iteration method was used to solve the eigenvalue problem, converging to the largest eigenvalue whose eigenvector corresponds to the physical neutron flux[11].

Given the neutron flux in all groups and regions, the power density in the fuel is recalculated, as indicated by the last step in Fig. 3.5. In the case of this work, the total power released in the fuel is a set condition.



**Figure 3.5:** Overview of the applied methodology to neutronics.

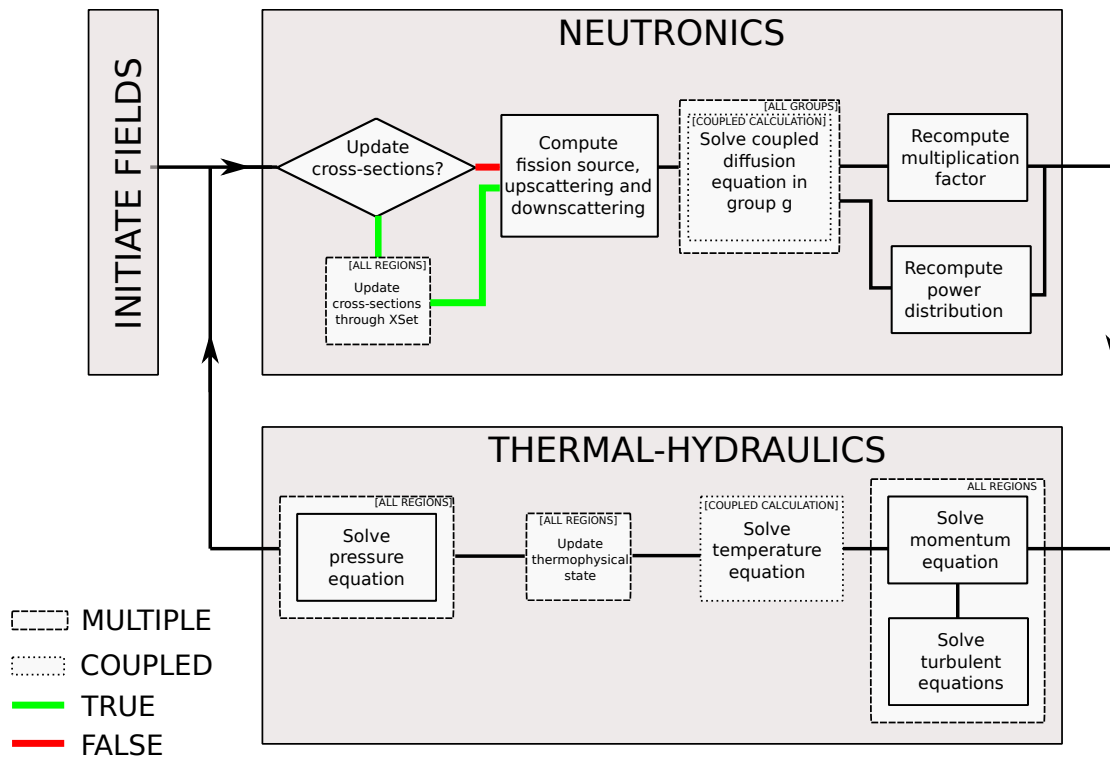
### 3.3.3 Algorithm for the coupled problem

The methodology for the thermal-hydraulic problem as is explained in subsection 3.3.1 and the methodology for solving the neutronics (see previous subsection) are integrated in the computational scheme displayed in Fig. 3.6.

Given a guess for the initial conditions of the system, and given a proper set of thermal-hydraulic and nuclear parameters, the neutron density field is solved as explained above. After a number of sub-iterations (set as a parameter, mostly sufficient with one or two) on only the neutronics, the thermal-hydraulics is solved using the updated fuel power profile. After the momentum predictor step, the temperature equation is solved finally followed by the pressure equations. One loop in Fig. 3.6 is denoted an outer iteration. The updated energy state of the system is conditionally used to recompute the nuclear parameters.

In the start of the simulation, the cross-sections are based on the initial guess not updated until the thermal-hydraulics and the neutronics have reached separately physical values. In the same manner the temperature equation, dependent on the fuel power, pressure and velocity profiles, is not solved until the neutronics and the thermal-hydraulics (excluding the temperature equation) both have physical profiles, occurring after just a few outer iterations.

The decoupled state calculations need to be performed on each relevant material region, giving multiple calculations as indicated by the dashed line in Fig. 3.6. The neutronics is computed in attached state (dotted line) but with a loop over the energy groups, giving multiple attached state calculations (dashed line).



**Figure 3.6:** Overview of the applied methodology to solve the complete set of equations.

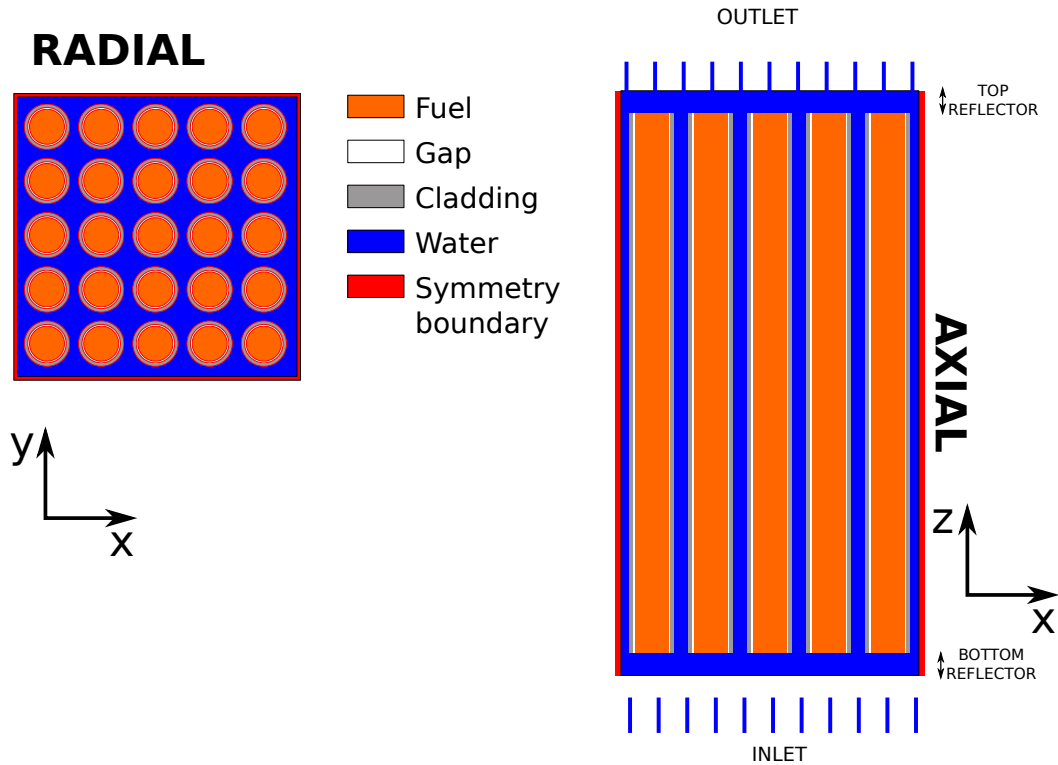
## Chapter 4

# Application

The computational tool developed and described in Chapter 3 is applied to the case of a simplified PWR fuel assembly (see the description in section 4.1). The methodology is first verified for a 2D test case (section 4.4), followed by a test for the 3D configuration (section 4.5).

### 4.1 Description of the fuel assembly

The fuel assembly that is used to verify the methodology is a 5x5 fuel bundle. Since the main focus of the application is to test the coupled neutronics and thermal-hydraulics scheme, some convenient simplifications of the design are assumed. The general geometry of the fuel assembly can be seen in Fig. 4.1, with the dimensions reported in Table 4.1. The general characteristics of the test system is given in Table 4.2.



**Figure 4.1:** 5x5 simplified PWR fuel assembly. Not to scale.

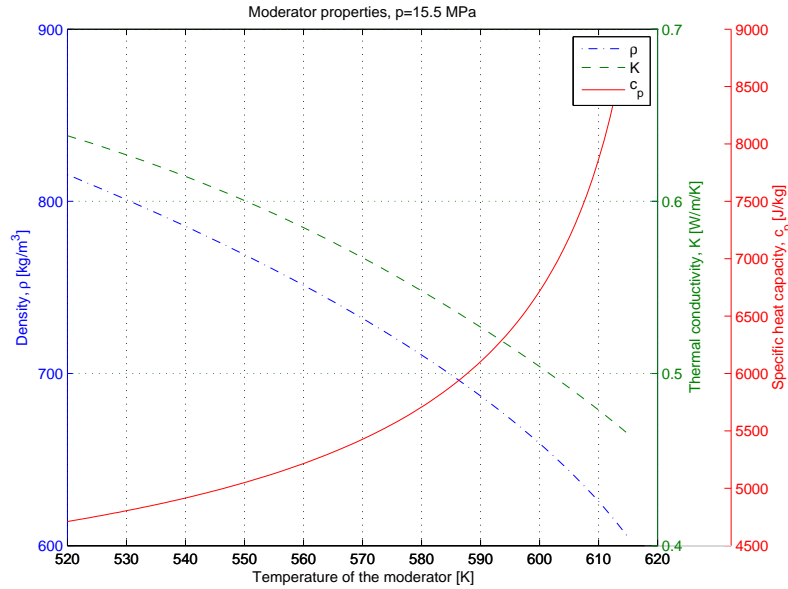
**Table 4.1:** Dimensions for PWR 5x5 fuel assembly.

Fuel pin radius	4.1 mm
Cladding inner radius	4.2 mm
Cladding outer radius	4.8 mm
Pitch	12.5 mm
Fuel height	3 m
Bottom reflector height	0.2 m
Top reflector height	0.2 m

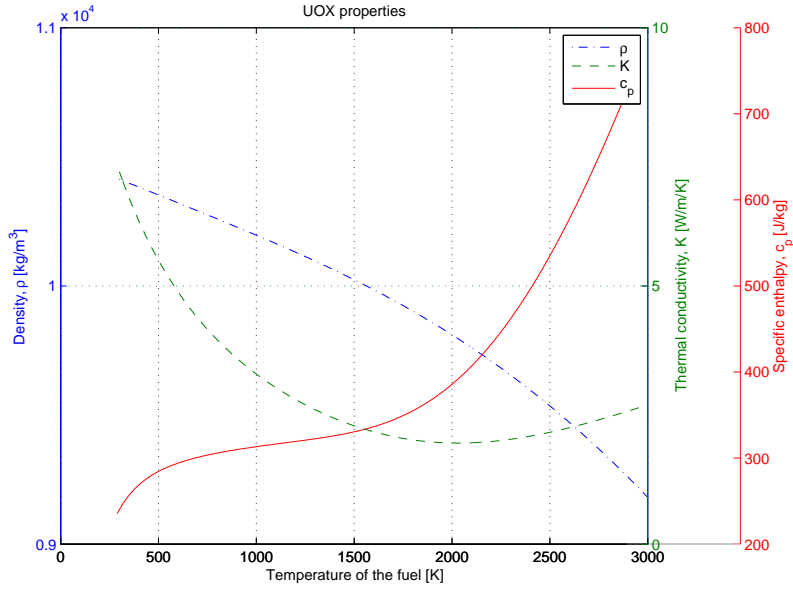
**Table 4.2:** Characteristics of PWR 5x5 fuel assembly.

Fuel	UOX or MOX
Moderator	Water, 1000 ppm boron
Gap	Helium, 0.1 MPa
Cladding	Zircaloy-2

The moderator/coolant is borated water, entering at a high pressure and flow rate at the inlet, flowing upward along the axial direction (see Fig. 4.1). The fuel bundle is open (i.e. there is no fuel box) and cross flow is allowed within the sub-assembly. Symmetry boundary conditions are applied at the x-y plane boundaries (indicated by the red color in Fig. 4.1). The simplified model includes no spacer. The thermophysical data for water that are needed in the calculations are taken from NIST[37]. As the axial pressure drop for a PWR core channel is small, isobaric data at a pressure equal to 15.5 MPa (typical value for the primary system of a PWR) is considered. The dependence of specific heat at constant pressure  $c_p$ , the density  $\rho$  and the laminar thermal conductivity  $K$  can be seen in Fig. 4.2. Do note that subcooled boiling possibly occurring in the moderator is disregarded.

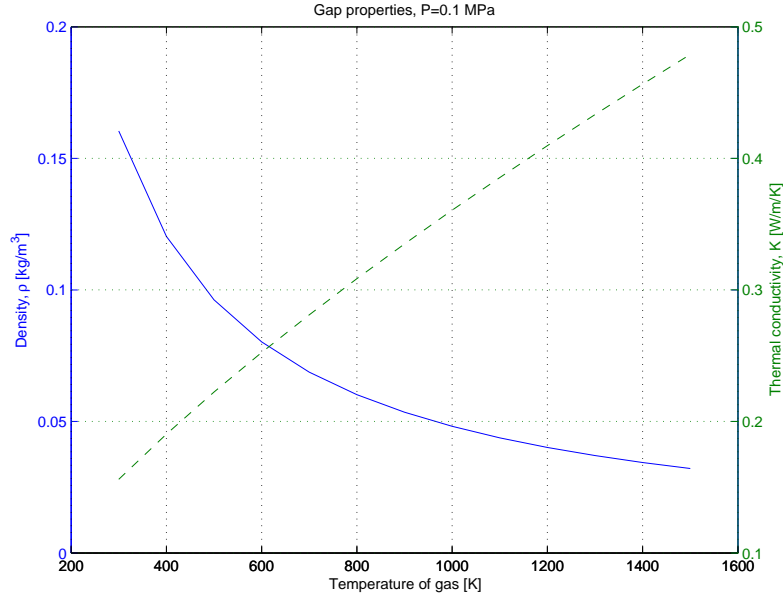
**Figure 4.2:** Water properties temperature dependence.

The fuel rods are placed in the bundle according to a checkerboard pattern of UOX (2% uranium-235) and MOX (2% plutonium, with isotopic composition typical to breed plutonium[38]). Each fuel rod includes a gap filled with Helium and a Zircaloy-2 cladding, and is assumed to be homogeneous in the axial direction. In the current case, fresh fuel is assumed. The UOX thermophysical properties are used also for the MOX fuel, except for the density. Den-



**Figure 4.3:** UOX properties temperature dependence.

sity of the UOX, thermal conductivity and specific heat capacity, as functions of temperature, are displayed in Fig. 4.3[39]. As seen from the figure, the density of UOX is temperature dependent. Still, such thermal contraction/expansion of the fuel has been neglected in the current work. Density, thermal conductivity and specific heat capacity of the helium are evaluated at the pressure of 0.1 MPa (i.e. atmospheric pressure), and their dependence with temperature is shown in Fig. 4.4[37]. Since fresh fuel is modeled, and since only steady state calculations are performed, no burn-up dependence is included. The exact isotopic compositions used for each material can be seen in Appendix B. Similarly, the material properties for Zircaloy-2 are evaluated for the



**Figure 4.4:** Helium properties temperature dependence.

range of temperature of relevance in the nuclear core calculations. The temperature dependence for the specific heat capacity and the thermal conductivity is outlined in Fig. 4.5[40].

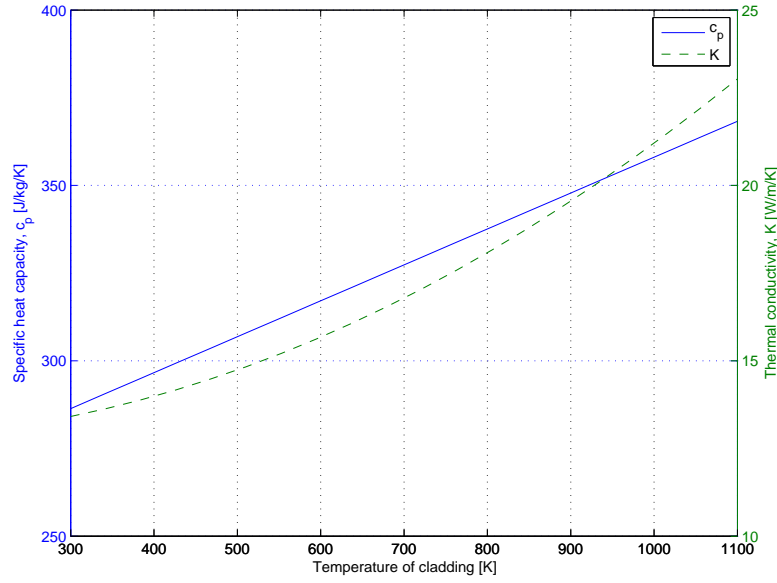


Figure 4.5: Zircaloy-2 properties temperature dependence.

## 4.2 Matrix solvers

Existing OpenFOAM solvers are used to solve the matrices describing the discretized problem. Details are reported in Table 4.3. In the specific, variations of conjugate gradient solvers are applied, except for the attached state temperature equations, where a Gauss-Seidel method is applied in order to avoid numerical instability.

**Table 4.3:** Summary of solvers used for each field, including typical convergence criteria and under-relaxation ( $\varrho$ ).

Field	Type of solver	Conv. criteria	$\varrho$	Solver	Preconditioner
$T$	Gauss-Seidel	$10^{-5}$	1.0	smoothSolver	GaussSeidel
$\Phi$	Stab. biconjugate gradient	$10^{-5}$	1.0	CG	Cholesky
$P$	Precond. biconjugate gradient	$10^{-4}$	0.3	PCG	DIC
$G$	Precond. biconjugate gradient	$10^{-6}$	1.0	PCG	DIC
$U$	Precond. stab. biconjugate gradient	$10^{-4}$	0.7	PCG	DILU
$k$	Precond. stab. biconjugate gradient	$10^{-8}$	0.7	PCG	DILU
$\epsilon$	Precond. stab. biconjugate gradient	$10^{-8}$	0.7	PCG	DILU

As depicted in Fig. 3.6, the solution of the coupled neutronic/thermal-hydraulic problem is achieved by solving single equations or a subset of equations in separated steps. The quantities that are determined in one step are then used to solve the equations in other steps. For this type of iterative schemes it might be necessary to utilize under-relaxation of the solved quantities such that:

$$X^{\text{new}} = X^{\text{solved}} - (1 - \varrho)(X^{\text{solved}} - X^{\text{old}}) \quad (4.1)$$

where  $X$  is a generic unknown to be determined in the present step. Thus, the updated quantity (with subscript new) is obtained as a linear combination of the two values estimated at a certain iteration (solved) and at the previous iteration (old).

The applied under-relaxation factors are presented in Table 4.3. For the neutronics and the temperature, the presented methodology fully avoids under-relaxation, i.e.  $\varrho = 1$ . For the pressure-velocity coupling, standard values are chosen.

## 4.3 Boundary conditions

To close the set of discretized equations all fields will need boundary conditions. A list of the applied boundary conditions used for each quantity is given in Table 4.4.

**Table 4.4:** Boundary conditions used for the different fields of the quantities computed

Field	Boundary	Type	Value	Implementation
T	Inlet	Constant value	540K	fixedValue
	Outlet	Zero gradient	-	zeroGradient
	Region to region	Coupled quantity	-	regionCouple
$\Phi_g$	Inlet	Constant value	$0\text{m}^{-2}\text{s}^{-1}$	fixedValue
	Outlet	Constant value	$0\text{m}^{-2}\text{s}^{-1}$	fixedValue
	Region to region	Coupled quantity	-	regionCouple
P	Inlet	Zero gradient	-	zeroGradient
	Outlet	Constant value	15.5MPa	fixedValue
	Region to region	Zero gradient	-	zeroGradient
k	Inlet	Constant quantity	$0.375\text{ m}^2\text{s}^{-2}$	fixedValue
	Outlet	Zero gradient	-	zeroGradient
	Fluid to solid	Wall function	-	zeroGradient
$\epsilon$	Inlet	Constant quantity	$12.0\text{ m}^2\text{s}^{-3}$	fixedValue
	Outlet	Zero gradient	-	zeroGradient
	Fluid to solid	Wall function	-	zeroGradient
K	Region to region	Coupled quantity	-	regionCouple
$D_g$	Region to region	Coupled quantity	-	regionCouple
U	Inlet	Constant value	$(0, 0, 3)\text{ms}^{-1}$	fixedValue
	Outlet	Zero gradient	-	zeroGradient
	Fluid to solid	No slip	$(0, 0, 0)\text{ms}^{-1}$	fixedValue
G	All	Marshak	-	marshakRadiation

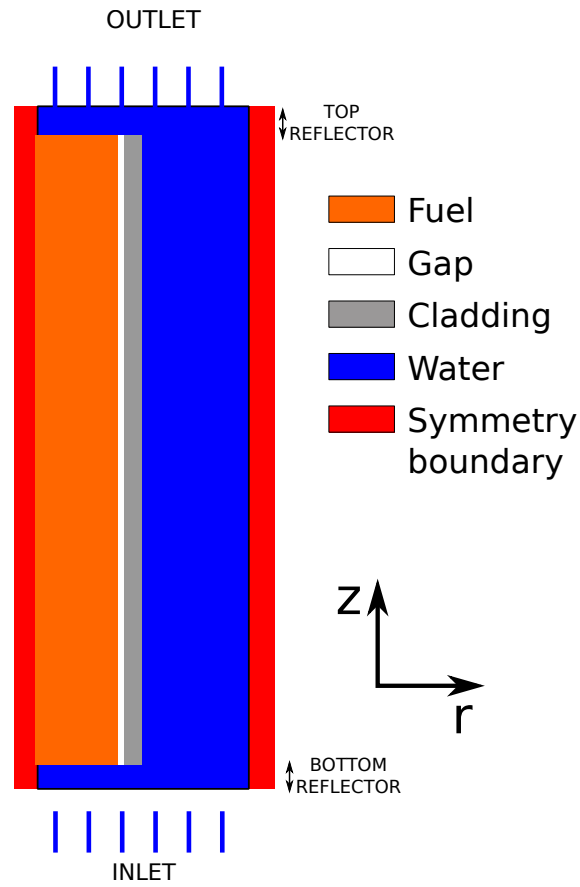
## 4.4 Two-dimensional case

A two-dimensional case related to the fuel assembly described in section 4.1 is studied for carrying out a first test of the developed tool. This choice has two practical advantages. First, a two-dimensional problem has one degree of freedom less as compared to the three-dimensional case, and thus a preliminary verification of the tool can be performed with decreased computational costs. Second, this step is useful to estimate radial density and radial temperature profiles that are needed for the generation of the macroscopic cross-sections.

For this purpose, a single UOX fuel rod with the surrounding moderator/coolant is considered. The system corresponds to an axial cut of the three-dimensional problem. The resulting system is outlined in Fig. 4.6. Because of the symmetry of the radial section with respect to the centerline of the fuel (corresponding to the z-direction in Fig. 4.6), calculations are run for a system that is only one half of the full fuel pin geometry. The radial sizes of all regions and the reflector height is preserved from the 3D case, whereas the fuel pin height is reduced to 1 m, further optimizing the computational time.

### 4.4.1 Computational mesh

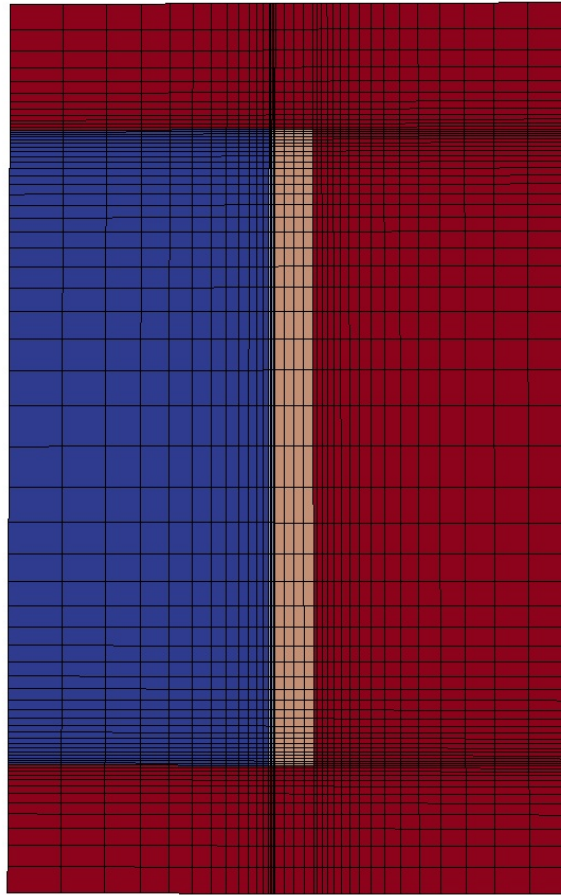
The computational mesh covering the 2D system is created with SALOME (see section 3.1.1) and is displayed in Fig. 4.7. The mesh is graded in radial ( $r$ ) and axial direction ( $z$ ), so that a better resolution can be achieved where the gradients are larger. Radially this corresponds primarily to the moderator/cladding interface, where a no slip boundary condition is applied (compare to Table 4.4), and axially it corresponds to the flow widening and narrowing in the



**Figure 4.6:** Two-dimensional PWR system. Not to scale.

top and bottom reflector. In total the mesh consists of 1152 elements. In the implementation of lower dimensional simulations in OpenFOAM the mesh is still consisting of 3D cells, although only a single layer thick. Consequently the presented mesh, has a thickness of 0.1 mm. Thus, the computational cells can be considered fully as volumes.



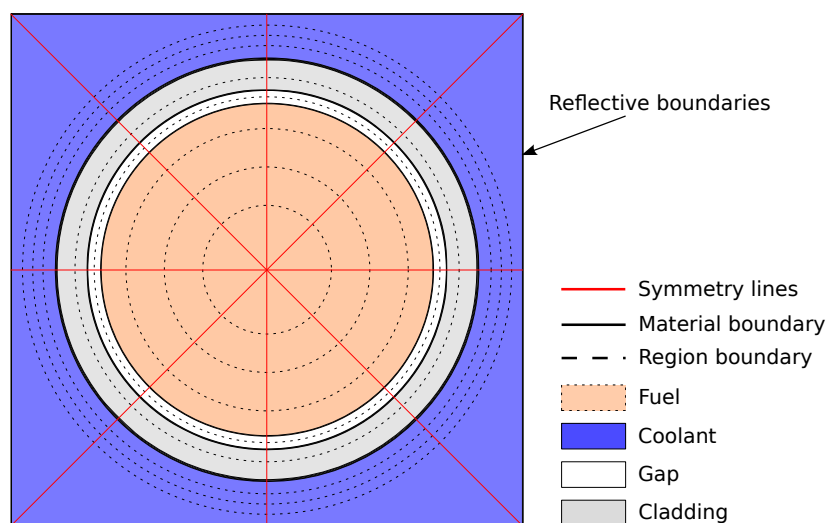


**Figure 4.7:** Mesh used for 2D system. The figure has been compressed by a factor 100 in axial direction

### 4.4.2 Two-dimensional neutron macroscopic cross-sections

The macroscopic cross-sections are generated according to the methodology described in section 3.2. In this specific case, the fuel region and the moderator region are divided in four radial sub-regions each, whereas the gap and the cladding are modeled with only one region each (see Fig. 4.8). The radial temperature is assumed to have a parabolic profile with maximum temperature in the center of the fuel (corresponding to the fuel symmetry line in Fig. 4.6). The density profile in the moderator is determined from the temperature (compare to Fig. 4.2). Three discrete neutron groups were assumed in the simulation.

The geometry for the cross-section generation is the same for both the two-dimensional test case and the three-dimensional test case (see section 4.5.1). For the two-dimensional case the cross-sections on the diagonal from the centre of the fuel to a corner was used (corresponding to the red diagonal lines in Fig. 4.8).



**Figure 4.8:** Geometry, including regional sub-divisions, for the SERPENT runs.

### 4.4.3 Convergence

The final solution is found by the iterative procedure as given in the methodology (section 3.3.3). As a characterization of the coupling of the neutronics and thermal-hydraulics, the equation residuals for neutronics and thermal-hydraulic quantities can be seen in Fig. 4.9 for the first 100 iterations and in Fig. 4.10 for all iterations applied.

To resolve the coupling of the system in a smooth manner the neutronics only is solved for 10 iterations. This gives a reasonable power profile in the fuel, without the thermal-hydraulics feedback, i.e. only based on the initial value of the temperature field. As is seen from Fig. 4.9b, without the thermal-hydraulic feedback it is enough with 10 iterations for the neutronic equations to converge.

After the first 10 iterations the thermal-hydraulics equations are added, and after another 10 iterations, solving the separate neutronics and thermal-hydraulics problems, updating of the cross-sections is added. This will close the loop of dependencies and the temperature will thus have a feedback on the neutronics.

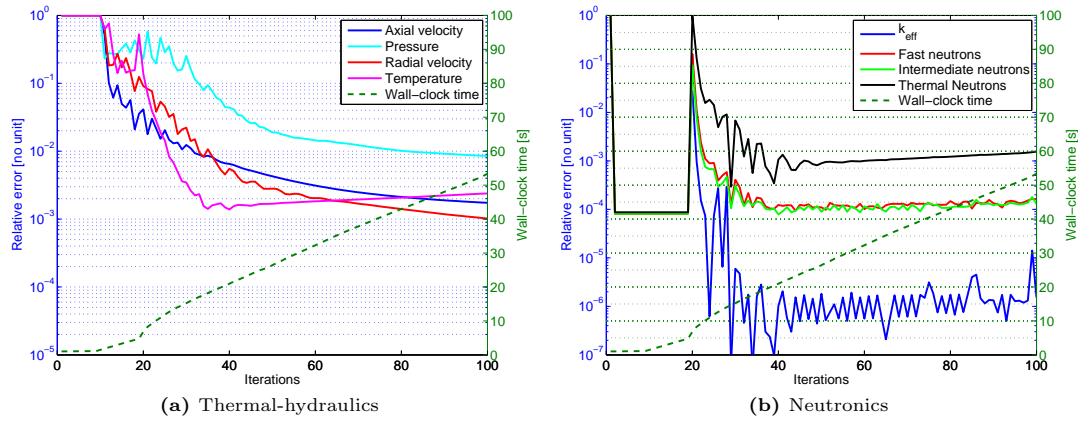
The exact number of iterations skipped in each step is of no larger significance, rather to be understood in combination with the remaining residuals for the mentioned iteration.

As seen from Fig. 4.9 the initial convergence is fast. Few iterations are needed to decrease the relative error in the effective multiplication factor below  $10^{-5}$ . After the rapid decrease in

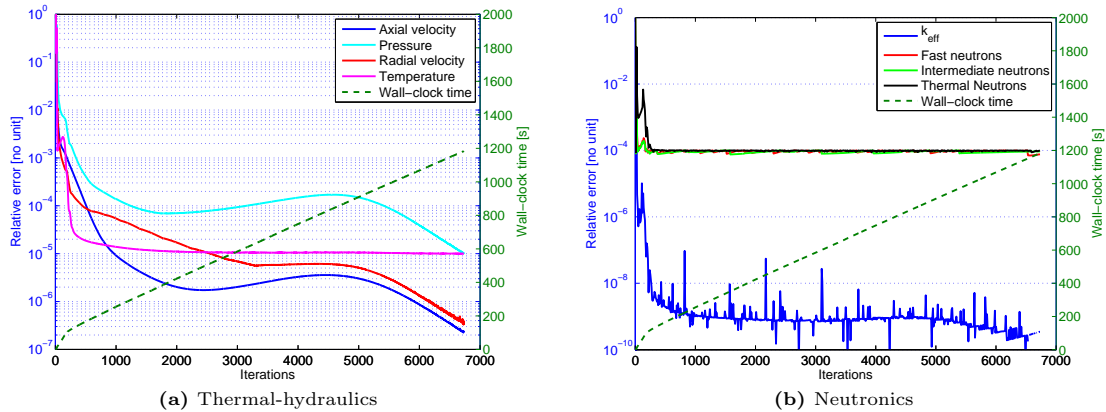
relative error the convergence slows. Primarily the pressure is slowly converging, lagging behind the convergence of all other thermal-hydraulic fields. The slow convergence is rather a property of the methodology used to solve the thermal-hydraulics (the SIMPLE algorithm), than the coupling of the different physics fields. Possible slow convergence of segregated solver algorithms like the SIMPLE-algorithm is a known phenomenon[41].

Considering the effective multiplication factor, the initial fast convergence is continued to a relative error  $10^{-9}$ . As seen from Fig. 4.11 the remaining error as compared to the final converged value of  $k_{\text{eff}}$  is less than  $10^{-5}$  in less than 500 iterations. Thus, the effective multiplication factor has converged much earlier than the thermal-hydraulics. The number of iterations needed to converge the solution will thus depend on the fields of interest. From the perspective of the neutronics fewer iterations are needed, whereas to get a precise pressure profile a very large number of iterations are needed.

For the coupled neutronics/thermal-hydraulics the residual decreasing rate will be limited by the coupling methodology rather than the convergence of the applied matrix solvers. For example the initial fast convergence of the neutronics, without thermal-hydraulics feedback, is very much slowed as the solution of the thermal-hydraulics is added. This is seen from the slope of the residual curves for all groups of neutron flux in Fig. 4.9, comparing the slope between iteration 1-10 and 20-40.



**Figure 4.9:** Convergence behavior for first 100 iterations for the coupled calculations on 2D mesh. Thermal-hydraulics solved from the tenth iteration and the cross-sections are updated from the 20th iteration.



**Figure 4.10:** Convergence behavior for coupled calculations on 2D mesh.

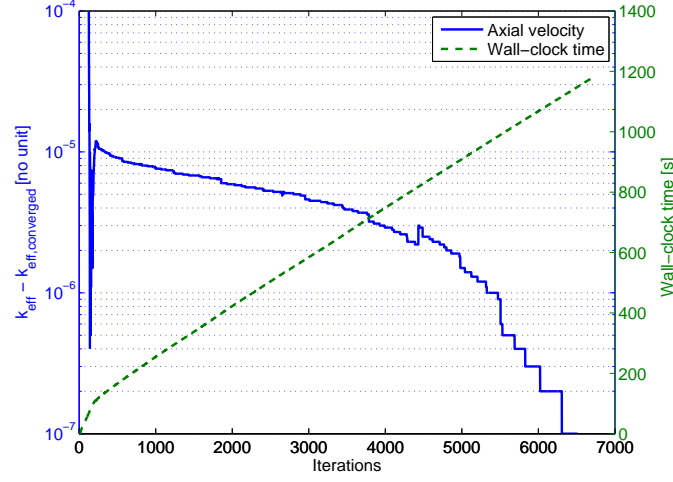


Figure 4.11: Relative error in  $k_{\text{eff}}$  compared to converged value.

#### 4.4.4 Results

The results presented in this section are computed with a fuel power density of  $24 \text{ MW/m}^3$ . The moderator temperature, density, pressure and axial velocity are shown in Fig. 4.13 whereas the axial dependence of these quantities for  $r = 8 \text{ mm}$  (corresponding to a radial position close to the moderator symmetry line in Fig. 4.6) are shown in Fig. 4.12. The behavior of the moderator is predicted correctly. Due to the power generated in the fuel region, the temperature of the upward forced flow increases monotonically; the density decreases; the pressure decreases within a fraction of a bar along the channel as expected; the axial velocity increases due to the density change, and it decreases when the fluid reaches the larger volume on the top of the fuel region.

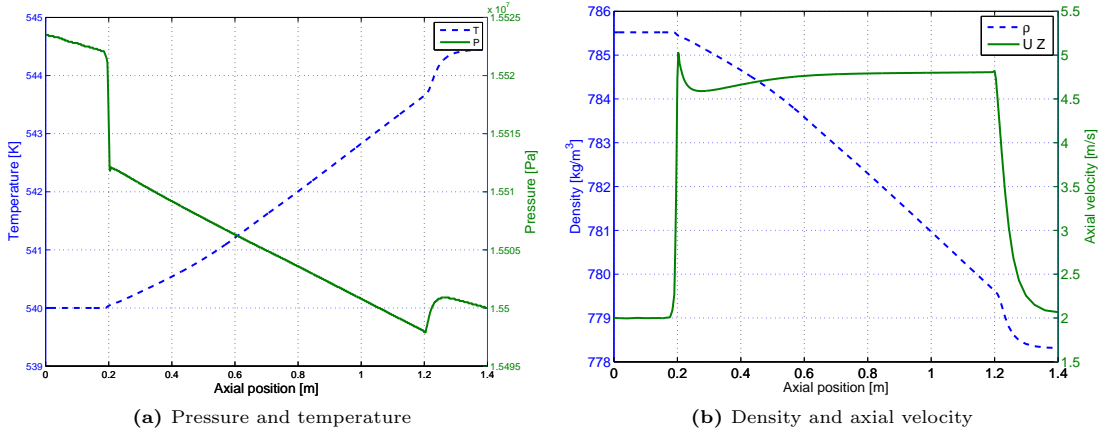
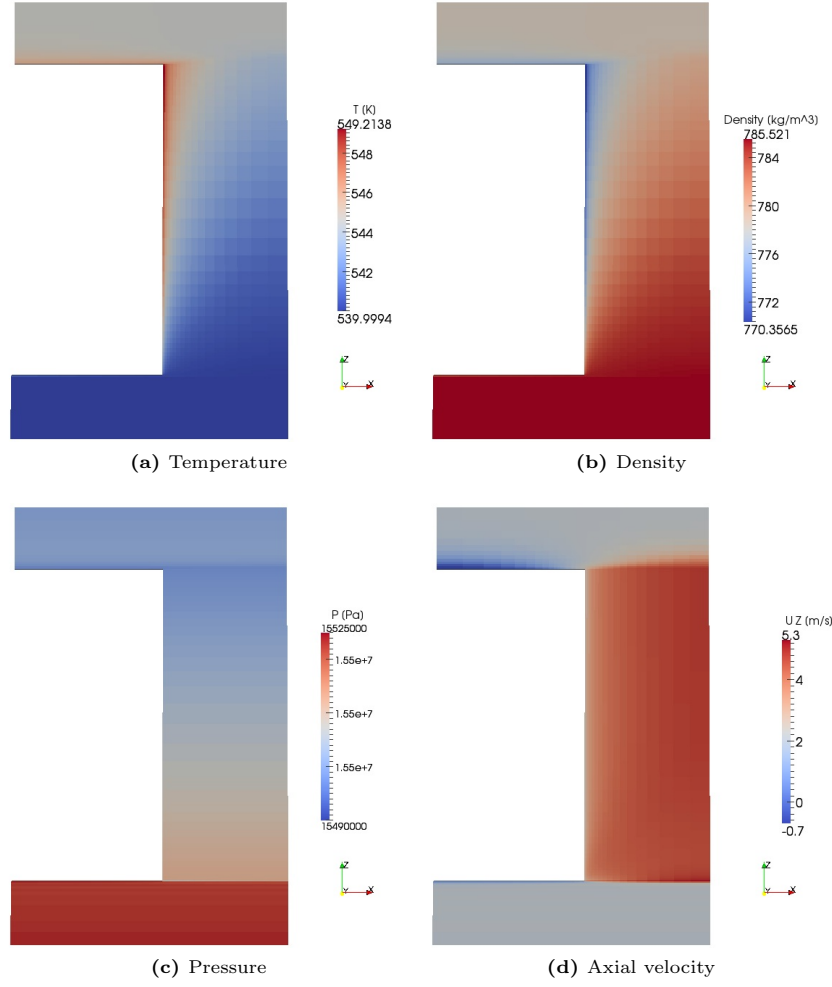
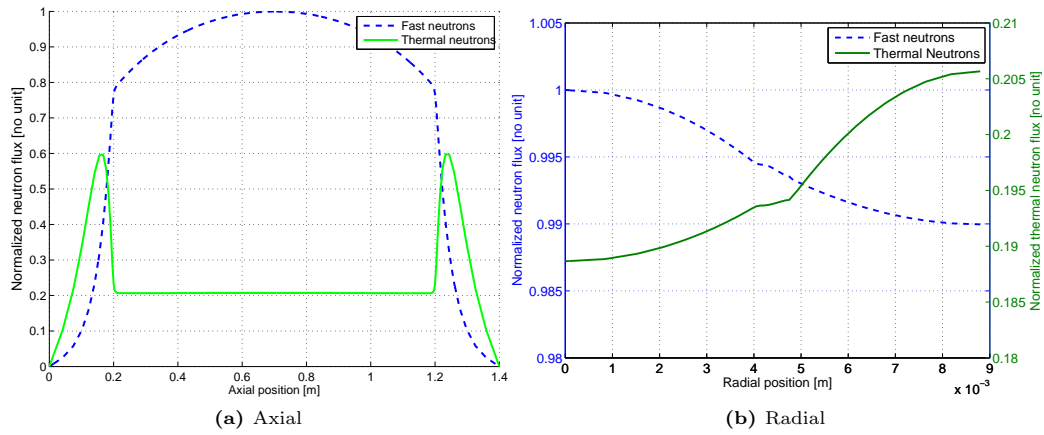


Figure 4.12: Moderator properties for the coupled problem, 2D calculations, axial dependence for  $x = 0.008 \text{ m}$  corresponding to moderator.

Figure 4.14 shows the neutron flux and the results are reasonable. The fast neutron flux has a maximum slightly below mid-elevation, whereas the axial thermal neutron flux is peaked in the top and bottom reflectors as the fast neutron current outward from the fuel is thermalized (see Fig. 4.14a). As regards the radial profiles, the thermal neutrons increases in the moderator due to the slowing-down of the fast neutrons (see the full blue line in Fig. 4.14b). On the other hand the fast neutron density is higher in the fuel, since the fissions release neutrons with high energy (see the dashed green line in Fig. 4.14b).

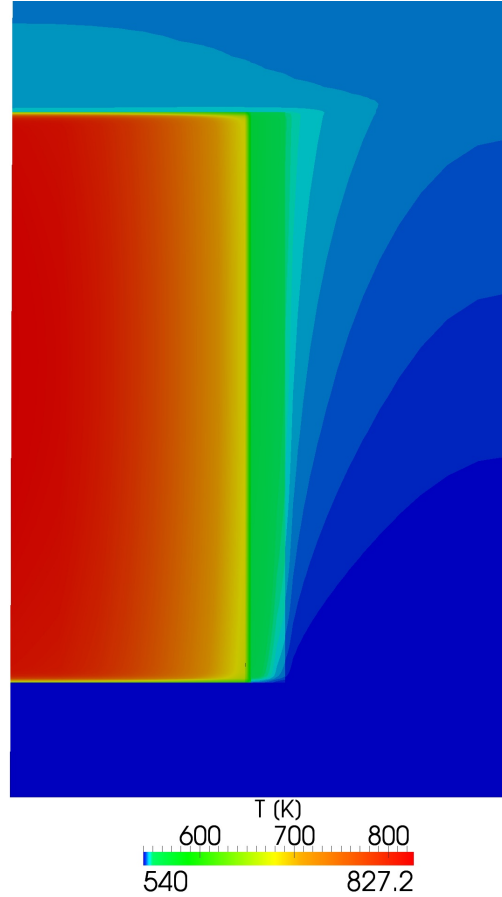


**Figure 4.13:** Moderator properties for the coupled problem, 2D calculations.



**Figure 4.14:** Axial (in moderator at  $r=8$  mm) and radial (at mid-elevation,  $z=0.7$  m) dependence of the group-wise neutron flux. Intermediate neutron group excluded in the plots.

The temperature dependence for the full system, including fuel, gap and cladding, can be seen in Fig. 4.15. The radial temperature gradient is decreasing from a value of 800 K at the fuel centerline to less than 600 K in less than 5 mm.



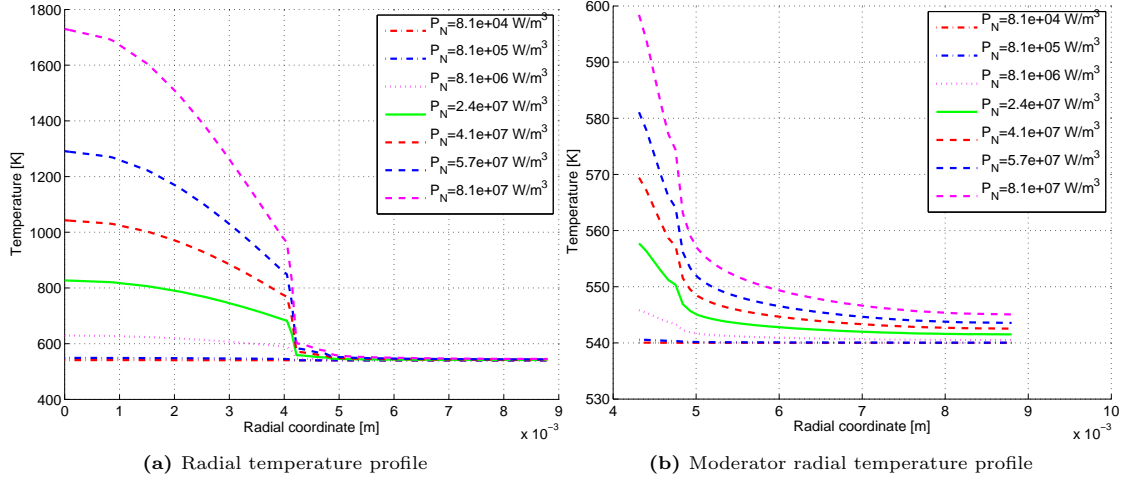
**Figure 4.15:** Temperature dependence of the full two-dimensional system (interpolated representation).

#### 4.4.5 Power dependence

The state of the system will depend on the power level in the fuel. When the power level changes, the temperature of the system changes as well, and this affects the axial and radial neutron flux via the cross-sections.

In order to estimate cross-sections as realistically as possible within the applied calculational scheme of a single pin environment (i.e. the fuel pin together with the surrounding moderator), realistic temperature and density profiles are needed (see section 3.2). For this purpose, the two-dimensional case discussed above, with the mesh defined in section 4.4.1 is employed to compute the radial temperature profiles for a set of power densities.

The resulting radial temperature profiles is seen in Fig. 4.16. As seen from the temperature profiles, the temperature gradient is largest in the gap (radial position  $4.1 - 4.2 \times 10^{-3}\text{m}$ ), as helium has the lowest thermal conductivity (see section 4.1). The profile in the moderator is relatively flat, due to the thermal conductivity enhanced by the turbulence of the flow. Furthermore, for higher power densities, the fuel temperature is higher and the moderator density lower, so that a negative feedback leads to a lower multiplication factor (see Table 4.5).



**Figure 4.16:** Radial temperature at axial mid-elevation for varying power density. Total power in the fuel given by  $P_N$ .

**Table 4.5:** Effective multiplication factor,  $k_{\text{eff}}$ , as a function of power density for the 2D case

Power density, $P_N$ [ $\text{W/m}^3$ ]	Total power [W]	$k_{\text{eff}}$
$8.12 \times 10^4$	0.1	1.0037
$8.12 \times 10^5$	1	1.0032
$8.12 \times 10^6$	10	1.0002
$2.44 \times 10^7$	30	0.99714
$4.06 \times 10^7$	50	0.99031
$5.68 \times 10^7$	70	0.98475
$8.12 \times 10^7$	100	0.97779

#### 4.4.6 Influence of the computational mesh

The discretization of the equation is based on the finite volume approach, i.e. a spatial homogenization is performed for each of the terms appearing in the equations, over each of the cells within the computational mesh. This procedure is an approximation, and thus an error is introduced. This error depends on the resolution of the selected computational mesh, and, generally, the smaller the cells, the smaller is the error. A quantitative value of the error introduced for each field is difficult to obtain. Instead a few key results can be studied, chosen as to represent the system. These key results are then compared for different mesh resolutions, and used as estimators of the discretization error, and of the error reduction achieved by mesh refinement (e.g., see [42]).

In the current case, the results of calculations with five different meshes are compared. These meshes consist of a different number of cells (see Table 4.6). The structure of the mesh is equivalent to the basic mesh displayed in Fig. 4.7, which also corresponds to Mesh 2 in Table 4.6. Mesh 1 is the coarsest mesh and Mesh 5 is the finest one. As regards the key parameters for the evaluation of the discretization error, the effective multiplication factor,  $k_{\text{eff}}$ , the fuel centerline temperature at mid-elevation and the total outlet enthalpy are considered. It must be noticed that the prediction of the effective multiplication factor is also influenced by the numerical procedure used to solve the neutron equation, and that the calculated total enthalpy depends also on the thermo-physical properties of the system (see section 4.1).

Because these parameters are compound parameters, with multiple dependencies, no standard mesh convergence estimator is applied. Instead the relative changes of the quantities are discussed in a qualitative manner, giving a general idea of the importance of the mesh resolution.

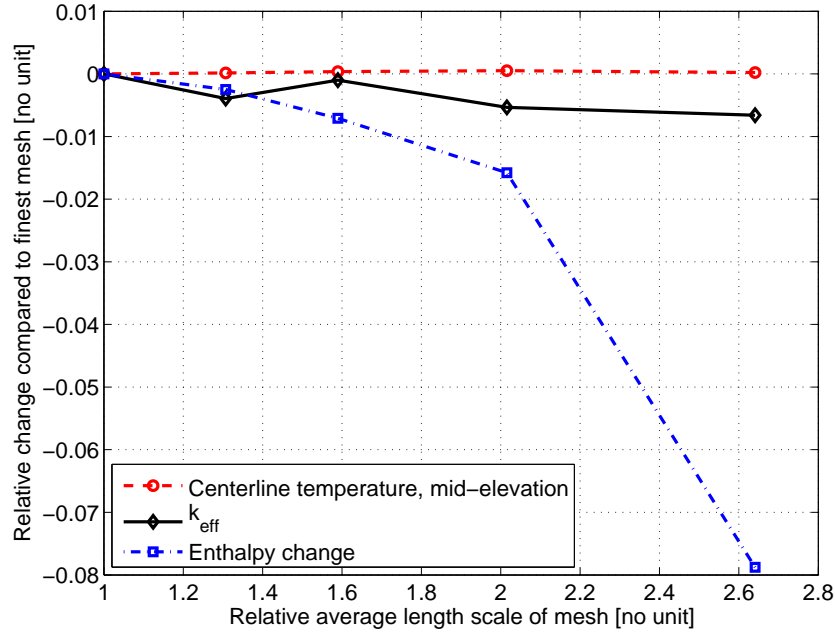
The results obtained by making use of the five meshes are shown in Fig. 4.17 and summarized in Table 4.7. In this set of calculations, the power density is  $24 \text{ MW/m}^3$ . It can be seen that the mesh resolution does not affect significantly the effective multiplication factor and the fuel centerline temperature, but the enthalpy. This outcome suggests that the same computational mesh for the neutronics and for the thermal-hydraulics is not an optimal choice. As the neutronics shows a smaller mesh dependence, a more coarse mesh could be used for the neutronics as compared to the thermal-hydraulics. However, since the neutron fields converges more rapidly (see discussion in section 4.4.3) the use of a finer mesh also for the neutronics will influence the total running time to a relatively small extent.

**Table 4.6:** Characterization of 2D meshes.

	Mesh 1	Mesh 2	Mesh 3	Mesh 4	Mesh 5
Cells	1152	2592	5280	9504	21216
Average volume [ $\text{m}^3$ ]	1.07e-09	4.75e-10	2.33e-10	1.30e-10	5.81e-11
Relative cell volume	18.42	8.19	4.02	2.23	1.00

**Table 4.7:** Characterization of 2D meshes, converged results.

	Mesh 1	Mesh 2	Mesh 3	Mesh 4	Mesh 5
$k_{\text{eff}}$	0.99839	0.99714	0.99283	0.99574	0.99184
$\Delta H$ [W]	32.60	30.70	30.44	30.30	30.22
$T$ [K]	827.28	827.03	827.15	827.33	827.45



**Figure 4.17:** Relative change in centerline temperature, enthalpy increase and  $k_{\text{eff}}$  for 2D meshes.



#### 4.4.7 Modeling of the heat transfer in the gap

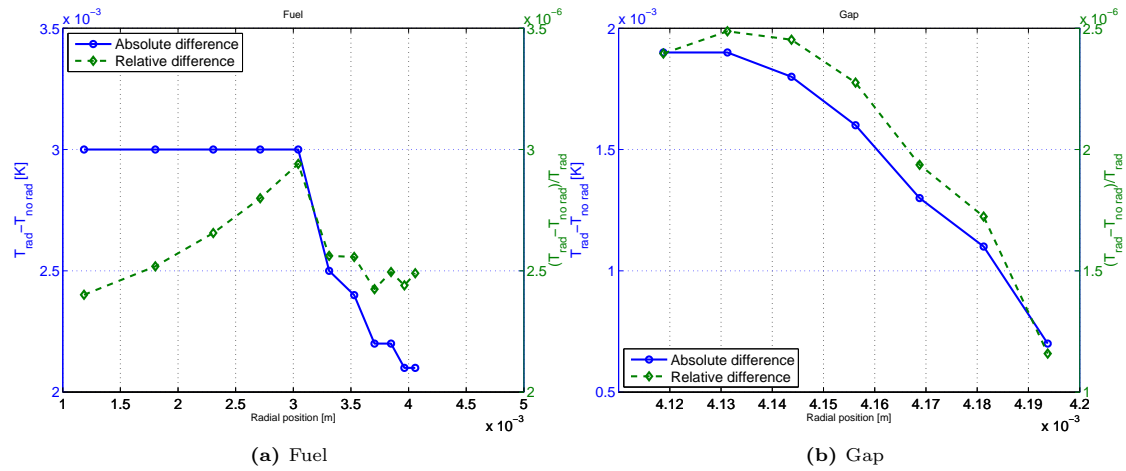
##### Influence from thermal radiation heat transfer

To study the importance of the radiation heat transfer, which is applied in the gap only, two different cases are compared. The first one is the standard used approach employed in the above calculations, computed for an emissivity of UOX being 0.83[43], the emissivity of the helium 0.95[44] and emissivity of the gap 0.6[45]. These values are chosen as the largest ones according to their ranges of uncertainty, so that the radiation heat transfer is not underestimated. In the second case, the radiation model is excluded, i.e. the radiation heat source is removed from the temperature equation of the gap (see eq. (2.62)).

The two runs are performed with Mesh 2, but with an increased radial resolution only in the gap, in such manner that the heat transfer in this region can be evaluated with a higher accuracy. The power density is of  $57 \text{ MW/m}^3$ . The comparison between the gap temperature calculated with or without the thermal radiation model is displayed in Fig. 4.18. Furthermore, the change in the effective multiplication factor is reported in Table 4.8.

**Table 4.8:** Comparison of  $k_{\text{eff}}$  applying versus disregarding the thermal radiation in the gap.

	$k_{\text{eff}}$
With	0.98450
Without	0.98447



**Figure 4.18:** Comparison of gap temperature including vs not-including the thermal radiation. Temperature at mid-elevation.

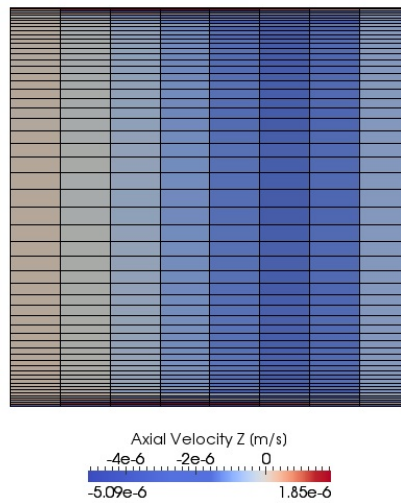
The results show that the radiation heat transfer in the gap has a very small influence, emphasized both by the temperature profile and the effective multiplication factor. However, it can be seen that the extra contribution given by the thermal radiation model to the total heat transfer is taken in account correctly, since the computed fuel and gap temperatures are lower, as expected.

##### Influence from convective heat transfer in the gap

In the model of the gap, the importance of the convective heat transfer is also investigated. Although the gap is a closed volume, the helium is non-stationary because of the temperature gradients, causing buoyancy effects. The same kind of phenomenon is also present in the moder-

ator, but it is very small in comparison to the main part of the convection, driven by the velocity of the forced flow.

To determine the influence of the convective heat transfer gap, the velocity of the helium is calculated for the case of  $P_N = 57\text{MW/m}^3$ , again using Mesh 2 with the increased radial resolution in the gap. The result is seen in Fig. 4.19. The left part of the figure, referring to the fuel-gap interface, has a positive velocity, whereas the right part, corresponding to the gap-cladding side has a negative value. Thus, the calculation shows that the expected circular flow in the gap due to the decreased density of the heated gas and increased density at the cooling side, can be captured. However, the magnitude of the velocity field in the gap is very low (e.g., compared with the moderator velocity shown in Fig. 4.13d), and, consequently, the heat transfer from convection is of negligible importance.



**Figure 4.19:** Axial velocity field in the gap. Geometry compressed by a factor 10000 in axial direction.

### Conclusion on gap modeling

The above considerations on the thermal radiation and the convective heat transfer gap point out that the gap can be modeled with reasonable accuracy as a solid, with the thermo-physical properties of helium, where the heat transfer occurs by conduction.

#### 4.4.8 Parallelization

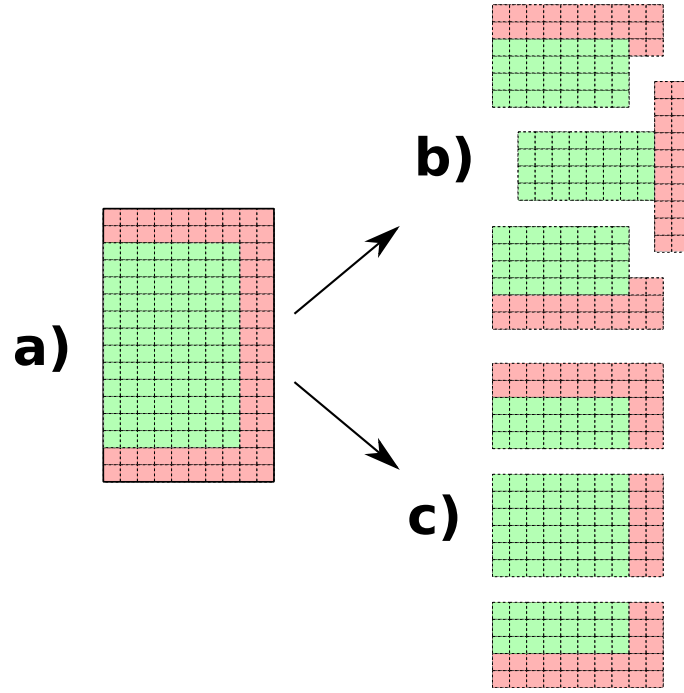
The type of coupled calculations presented lead to large matrix systems, comprising a very large number of computational cells. It is therefore desirable to parallelize the calculations. For this purpose, the physical system is split properly in a number of sub-systems, solved in parallel, on multiple distributed memory systems. This is possible with OpenFOAM, using built-in support for MPI[46].

#### Procedure

As mentioned above, the physical system must be decomposed, and procedures to perform this task is available in OpenFOAM. However, special care has to be taken when the coupled equations in the attached state are treated. An example of a decomposition of two-material region can be

seen in Fig. 4.20. By applying the simplest decomposition method available in OpenFOAM (i.e. **simple**), each material region will be split in equally large chunks distributed on the different nodes, possibly giving an undesirable boundary between the regions (see (b) in Fig. 4.20). The equal size decomposition might lead to ill-shaped regions, in turn giving numerical instabilities for the solution of the matrix on each CPU. To overcome this issue, a decomposition method based on fixed geometrical planes has been implemented for the purpose of this work. With this decomposition, the surface between the nodes will consist of fewer cells (as in Fig. 4.20).

The application results in a list of all cells in the attached mesh, and the corresponding node the cell belongs to. This list is read by the standard decomposition application in OpenFOAM. In this way the decomposition (c) in Fig. 4.20 is achieved.



**Figure 4.20:** Decomposition problem: a) overall system; b) equal size decomposition; c) geometrical decomposition

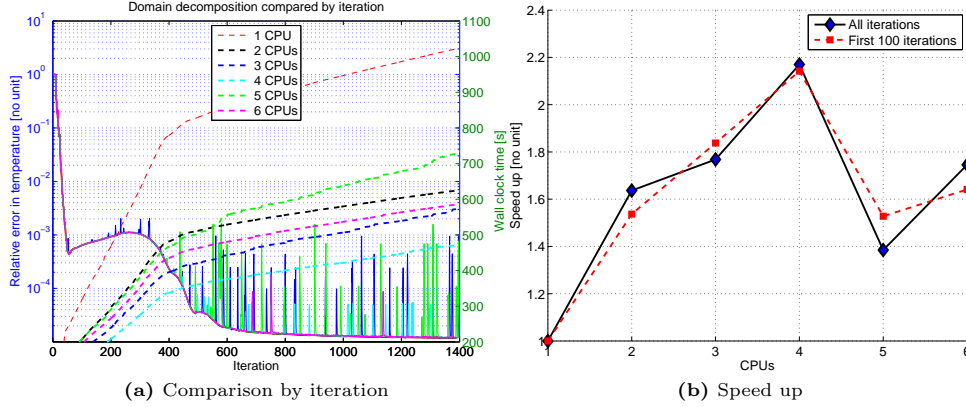
## Performance

The performance of the parallelization will, in general, depend both on the decomposition scheme and the implementation of the parallelized methodology. To exemplify the parallelization of the implemented OpenFOAM solver, Mesh 2 with  $P_N=30$  W is run for a variable number of CPUs. The speed up of the parallelized code is defined as:

$$\text{Speed up}_X = \frac{\text{Wall clock time, 1 CPU}}{\text{Wall clock time, X CPUs}} \quad (4.2)$$

where the wall clock time is the real time, whereas the CPU-time would measure the sum of the time elapsed on each CPU.

The result of such run can be seen in Fig. 4.21. The convergence profile, as exemplified by the temperature equation convergence, has the same behavior independent of the number of CPUs. This is expected as the parallelization in OpenFOAM is implemented to exchange shared cell values at each step of the iterative matrix solvers. The elapsed time is at maximum for a single processor. It is further seen that the speed up is neither linear with the number of CPUs, nor



**Figure 4.21:** Parallelization of 2D coupled problem.

double as fast for 2 CPUs as for a single CPU. The communication between the CPUs brings a certain overhead cost for the exchange of matrix elements.

Furthermore, as shown in the general convergence behavior for a single CPU, the thermal-hydraulics will need many outer iterations, going back and forth between velocity/pressure and temperature, increasing the amount of data sent between the CPUs.

For the computer used<sup>1</sup>, the results of using 5 and 6 CPUs are more that of hardware character, than of the code.

## 4.5 Three-dimensional case

Considering the large amount of input data needed to run calculations for a nuclear fuel assembly and the complexity of the overall methodology, a script has been written such that the entire process can be optimized and automatically executed. The script sets up a file structure as demanded for general OpenFOAM run and the files needed for the specific OpenFOAM application solving the nuclear reactor multi-physics problem. Specially for the many boundaries and boundary conditions arising in a multiple pin run, it is time consuming to manually setup the case<sup>2</sup>.

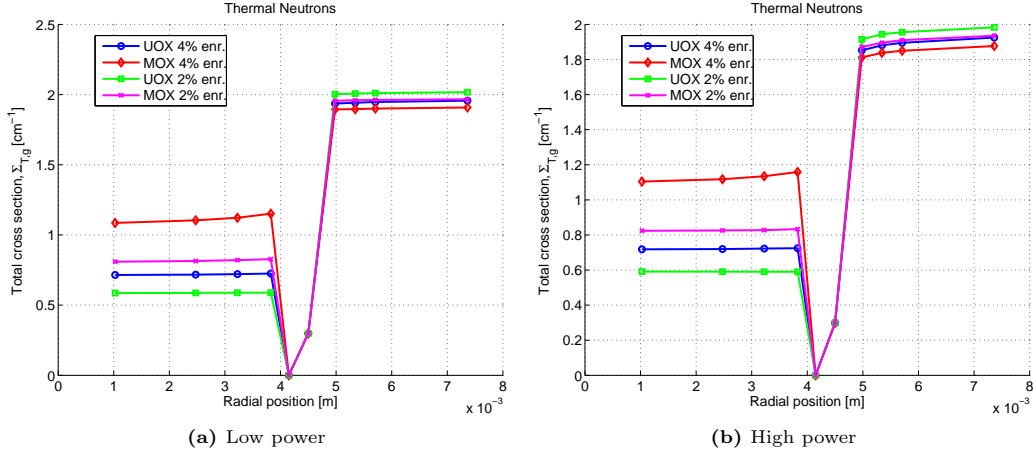
### 4.5.1 Neutron macroscopic cross-sections for the three-dimensional case

To generate the proper set of macroscopic cross-sections and neutronic parameters for the three-dimensional fuel assembly, the methodology described in section 3.2 is employed.

Four types of nuclear fuels are studied: UOX enriched with 2% and with 4% uranium-235 and MOX with 2% and with 4% plutonium. The cross-sections are condensed according to a two energy group structure, i.e. thermal and fast neutrons. The radial temperature and density profiles estimated in section 4.4.5 (complemented by extra high-temperature runs from the 2D case) are used for all the four cases, although computed for UOX with 2% uranium-235. As for the set of cross-sections created for the 2D application, the fuel and the moderator are split in four radial regions, and the gap and the cladding in a single region (see Fig. 4.8). The generated 2D cross-sections are used for all axial levels in the 3D system, thus assuming the same radial temperature profile for all axial positions. The total thermal cross-sections for the lowest and the highest power from section 4.4.5 can be seen in Fig. 4.22.

<sup>1</sup>All simulations are performed using an Intel®Core™i7-2670QM CPU, with 4 hyper-threaded CPUs at a frequency of 2.20GHz.

<sup>2</sup>The file structure for a 5x5 pin problem gives rise to an input structure with more than 3000 files (considering cross-sections sets of 6 state points).



**Figure 4.22:** Comparison of thermal absorption cross-section for low and high power case.

The variations of the cross-sections along the radial direction arise from the neutron flux profile that appears as weighting function in eqs. (2.14)-(2.18). Such a neutron flux depends on the composition of the fuel, e.g. it will differ for UOX with 2% and with 4% uranium-235.

On the other hand, when the two different power levels are compared, it can be seen that the effect of warmer moderator (i.e. lower density) will lower the total cross-section in the moderator. Besides, the fuel temperature must be taken in account since it impacts also on the total cross-sections.

This example points out the need of cross-sections that depends on space, material composition and the thermophysical state of the material. This is true not only for the total cross-sections, but all material dependent quantities, including cross-sections, fission neutron yield, fission neutron spectrum and diffusion coefficient.

It should be noted that the applied temperature profiles are not generally true, but depends, for instance, on the moderator inlet velocity which affects the axial moderator temperature distribution. The profiles used and the generated macroscopic cross-sections are rather to be seen as a best approximation based on the 2D geometry of a single fuel pin environment (i.e., a fuel pin together with the surrounding moderator).

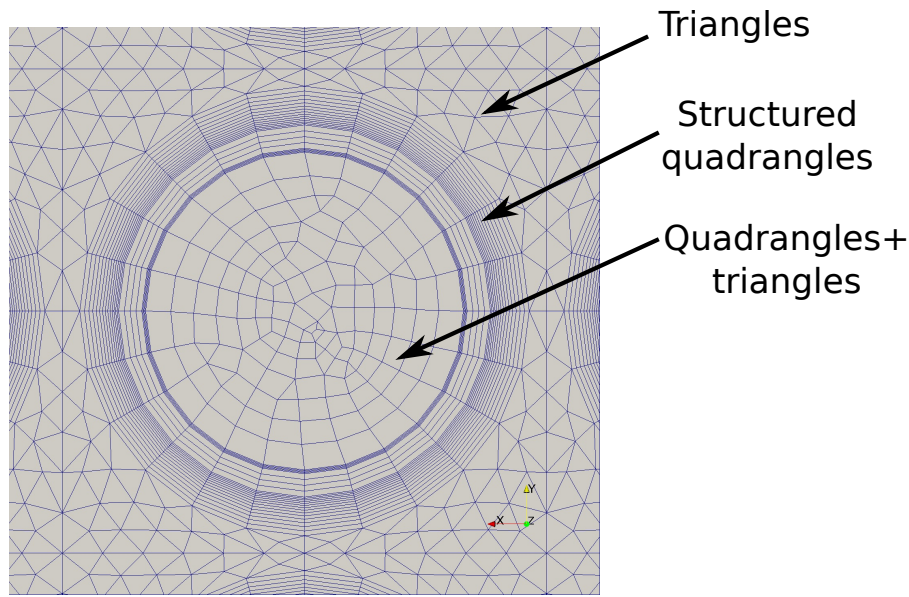
### 4.5.2 Computational mesh

For the three-dimensional calculations a hybrid mesh is created, utilizing both structured and unstructured regions, thus, consisting of both prism and hexahedral shaped elements. The general radial structure of the mesh is outlined in Fig. 4.23. General characterization data of the mesh is given in Table 4.9. The axial structure is similar to the axial character of the 2D mesh, with refinements close to the flow widening and narrowing. The single pin mesh is repeated in a lattice structure for the multiple pin runs.

The increased radial resolution in the moderator, close to the cladding wall, is adjusted to fit the purpose of using unresolved wall functions for the velocity and turbulence profile.

**Table 4.9:** Characterization of the 3D mesh for a single fuel pin.

Cells	65160
Axial intervals	80
Average volume [m <sup>3</sup> ]	8.15e-09
Max skewness	0.65
Max aspect-ratio	3200



**Figure 4.23:** Radial hybrid mesh generated for 3D run of multiple fuel pins.

### 4.5.3 Results

The results of the calculations performed for the simplified PWR assembly described in section 4.1 with UOX (2% uranium-235) and MOX (2% plutonium), and with a total power equal to 1 MW, are presented in this subsection.

Figure 4.24 presents the temperature distribution of the full system and the power density in the fuel rods at mid-elevation. The temperature in the MOX pins (Fig. 4.24a) is higher because of a higher fission cross-section, leading to a higher pin power (Fig. 4.24b).

The different power density in UOX and MOX elements leads also to slightly different moderator temperature (Fig. 4.25), depending on the nearby fuel material. Comparing figures 4.24 and 4.25, the moderator temperature is slightly higher around the MOX fuel pins.

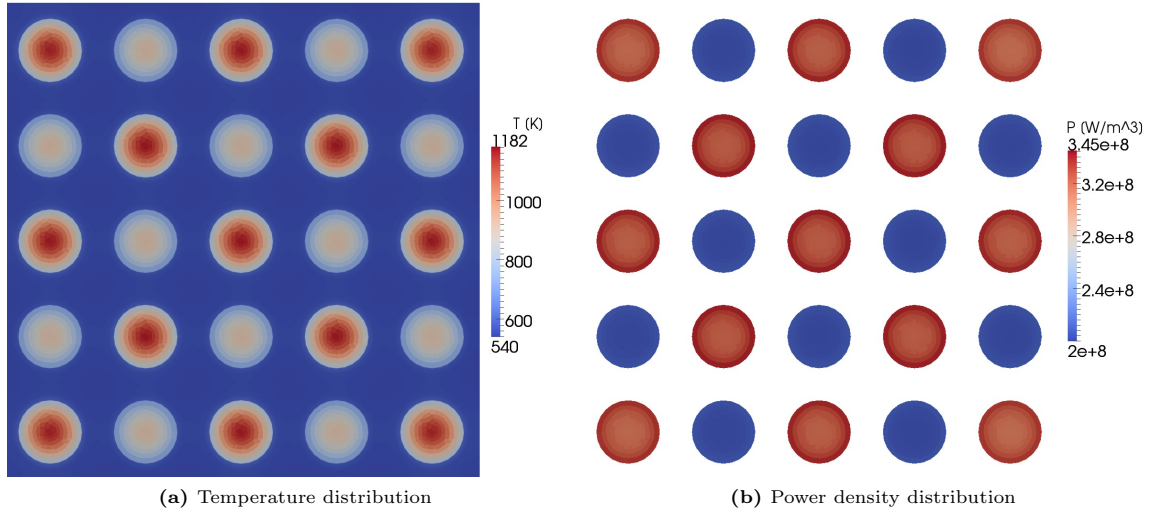
The axial moderator inhomogeneity is displayed in figure 4.26, for a cut along the x-axis, through the middle row of pins. The moderator is seen to be warmer in the vicinity of the MOX pin. In general the temperature gradients in the moderator, along the radial direction, are small.

As regards the neutronics, the thermal flux has its minimum magnitude in the MOX pins (figure 4.27a), and the fast flux reaches its maximum in the MOX pins. This is consistent with what is expected as the largest number of fissions, i.e. the largest number of fast neutrons released from the fission process, occurs in those pins.

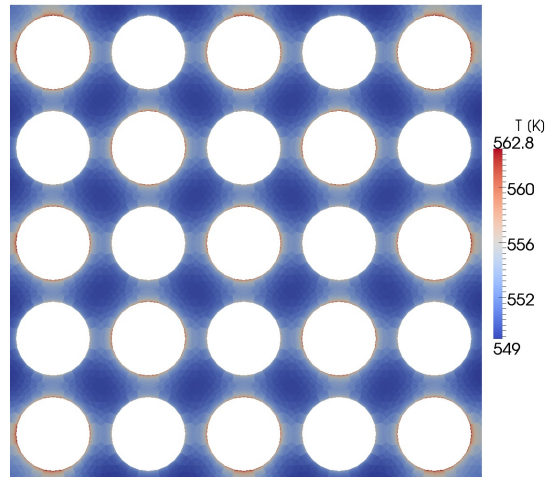
Comparing the power density (figure 4.24b) and the thermal neutron flux (figure 4.27a), it can be seen that a shielding effect takes place. In fact the power density is higher in the outer part of the fuel pin, whereas the thermal neutron density decreases approaching the center of the fuel.

The radial variation of the total thermal cross-sections for the moderator region is shown in figure 4.28. Such a distribution arises not only because of the spatial density and temperature distributions of the water. The checkerboard pattern that can be observed, is a result of the fact that the cross-sections are calculated for a single fuel pin environment only. Thus, there is no compensation for the surrounding, which leads to the non-physical discontinuous moderator cross-sections. To avoid this a larger geometry would have to be implemented for the cross-section generation in the moderator, or an interpolation scheme could be implemented for the cross-sections in the moderator. The first option would cripple the modularity of the applied methodology, and the second approach has the disadvantage that it would not be correctly done without compensating for the flux profiles, again introducing the need for further and larger





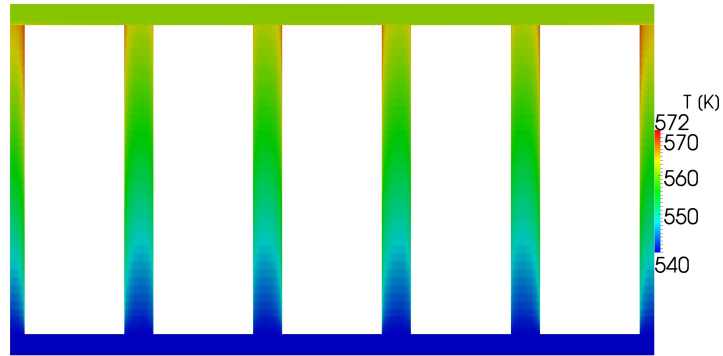
**Figure 4.24:** Radial temperature and power density distribution at mid-elevation. Higher temperature and power density in the MOX pins.



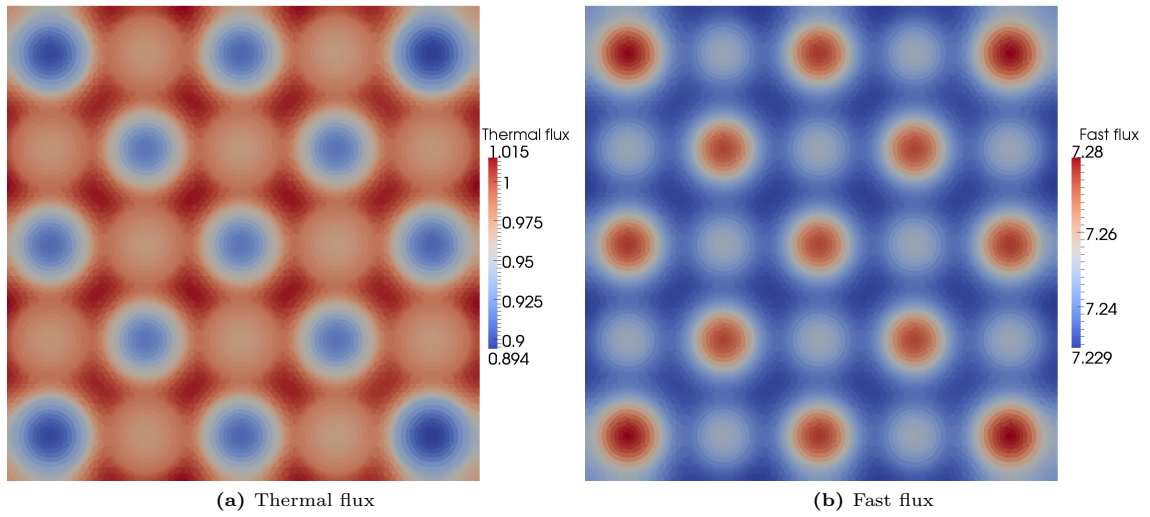
**Figure 4.25:** Radial temperature distribution in the moderator at mid-elevation.

cross-section calculations.

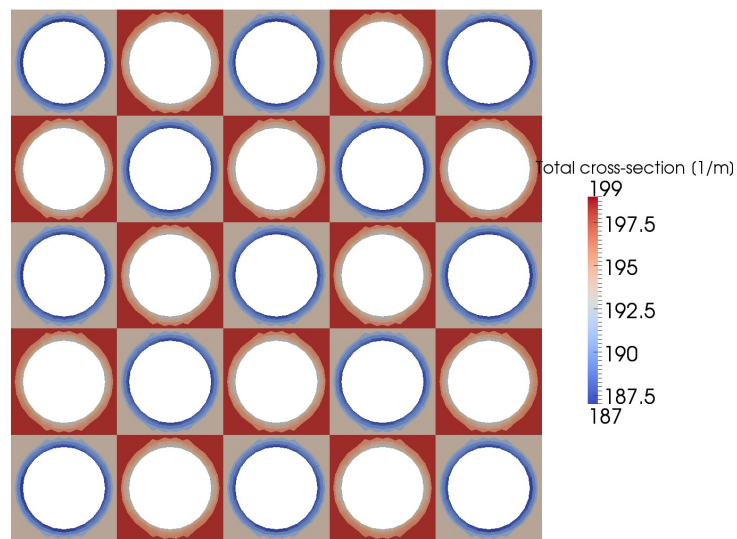
Radial variations of the cross-sections are also found inside the cell of the same pin. These variations are not only due to a temperature effect (as the temperature was shown to be almost constant in figure 4.25), but also due to the difference in neutron flux profiles for different radial positions as explained in section 4.5.1.



**Figure 4.26:** Axial temperature dependence in the moderator for plane through center of 5x5 lattice.



**Figure 4.27:** Radial neutron density distributions at mid-elevation.



**Figure 4.28:** Thermal total cross-section,  $\Sigma_{T,1}$ , radial dependence for mid-elevation.



#### 4.5.4 Moderator homogenization

As shown above, the computational tool developed in this work can predict spatial distributions of the moderator properties within the sub-channels of the fuel assembly. This level of detail is generally not captured with other approaches, e.g. with thermal-hydraulic system codes for nuclear reactors.

An example of the error introduced by assuming a homogeneous spatial distribution of the moderator properties in the fuel assembly is given. This is to be seen as an example of the possibilities of the high-resolution calculations, and not as an exact comparison to any existing methodology. In the below example, only the moderator is averaged, thus there is still a non-constant temperature profile in the fuel, the cladding and the gap.

In order to get a correct axial moderator profile, and, thus, ensuring a correct energy extraction in the moderator, the methodology is first applied to calculate a converged 'inhomogeneous' result. The moderator is axially averaged over axial levels of the mesh. The averaging is calculated using the enthalpy of the moderator such that:

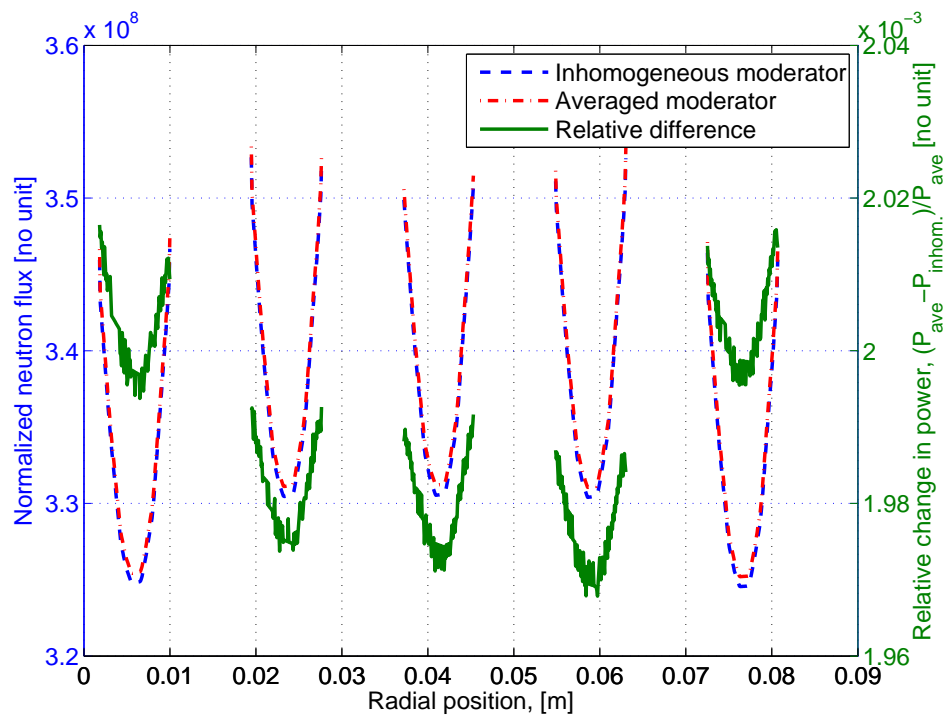
$$H_j = \frac{\sum_{i \in j} H_i V_i}{\sum_{i \in j} V_i} \quad (4.3)$$

where  $j$  denotes the axial level in the mesh, and  $i \in j$  corresponds to all cells in the axial level  $j$ . The calculated average enthalpy, for each axial, level is then converted to temperature. This calculated temperature is used for updating the cross-sections for the averaged state.

After the homogenization of the moderator, the neutronics is solved again, using nuclear properties based on the homogeneous conditions. The thermal-hydraulics is not further solved. Do note that there will still be inhomogeneous conditions in the fuel, cladding and gap, as these regions are left as for the converged inhomogeneous case.

The radial power density at mid-elevation, for the fuel pins aligned with a diagonal passing through the center of the fuel assembly (i.e. through a diagonal of MOX fuel pins), for the homogenized and inhomogeneous moderator is seen in Fig. 4.29. From comparison, it can be seen that the power density increases in the MOX fuel pins for the homogeneous case. As the radial temperature variations are small (compare to Fig. 4.25), the relative change from the moderator averaging will also be small.

Again, this example does not correspond exactly to the case of any commercial or standard code, but is merely an example of the capability of the high-resolution calculations as applied in this thesis.



**Figure 4.29:** Power profile at mid-elevation, along a diagonal passing through center fuel pin.

# Chapter 5

## Conclusions

In the presented work, a methodology has been designed for high-resolution, steady-state calculations in which neutronics and thermal-hydraulics of single-phase flow are coupled. This kind of methods are of particular interest in nuclear reactor applications where more detailed simulations can be useful for studying better safety options as well as more advanced fuel designs.

### 5.1 Summary of the methodology

The methodology is based on a newly developed computational tool in which the steady-state multi-group neutron diffusion equation has been integrated in a CFD solver for the thermal-hydraulics (chapter 3). The new integrated solver has been built by combining and modifying capabilities that are available in the open source CFD software OpenFOAM, which allows performing all calculations within the same code, and it is targeted at the specific configuration of a nuclear fuel assembly. Accordingly, it includes models for heated single phase turbulent flow of the moderator/coolant and heat transfer in the fuel, in the gap, and in the cladding. Thermo-physical properties of the materials used in the CFD solver have been incorporated in the form of temperature dependent models. The numerical algorithm used for the solution of both the neutron and thermal-hydraulic fields relies on discretization operators and matrix solvers provided in OpenFOAM, and the computational grid is created with the software SALOME.

Moreover, a procedure that makes use of the nuclear Monte Carlo code SERPENT has been implemented for generating proper set of macroscopic neutron cross-sections and neutron parameters that are needed for the coupled calculations.

### 5.2 Summary of the results

In order to verify the methodology, simulations of a simplified fuel assembly under steady-state PWR conditions have been performed (chapter 4). First a two-dimensional case, and then the full three-dimensional case have been analyzed. The results discussed in sections 4.4 and 4.5 are consistent and show that the methodology has been implemented successfully.

From the two-dimensional case, it can be seen that (section 4.4):

- The simulations of the axial profiles for moderator temperature, density, velocity and pressure drop are coherent;
- The axial neutron flux for the thermal and the fast groups are predicted in a reasonable manner. For instance, the accumulation of thermal neutrons due to the slowing down of the fast neutrons that enter in the moderator is captured;
- As expected because of the large gradient, a significant temperature decrease from the center of the fuel to the periphery is observed;

- Thermal radiation and convective heat transfer in the gap are negligible;
- An increased power density leads to a lower effective multiplication factor, which is an expected feedback of the system.

In the case of the three-dimensional fuel assembly, where two different types of nuclear fuel (UOX and MOX) were placed according to a checkerboard pattern, it has been found that:

- Radial and axial temperature distributions in the fuel and in the moderator are consistent with the calculated neutronic power;
- Fast and thermal neutron fluxes at the outer region of the fuel pins are reproduced;
- Differences in the calculated power distribution are obtained by comparing the current calculations and the case in which moderator properties are homogenized.

The procedure for evaluating the macroscopic cross-sections has been tested by comparing the results for four different nuclear fuels (including UOX with 2% and 4% uranium-235 and MOX with 2% and 4% plutonium). For instance, as expected, the total thermal macroscopic cross-sections, along the radial direction, in a fuel pin environment,

- are higher in the MOX fuel than in the UOX region;
- are lower in the moderator region when higher power densities are considered.

Nevertheless, one must bear in mind that the macroscopic cross-sections are prepared for a single fuel pin environment and no compensation for surroundings was included.

As regards the computational scheme developed in this work, it has been shown that:

- The iterative coupling between the neutronics and the thermal-hydraulics has been proven functional and stable, utilizing partial convergence of the neutronics and the thermal-hydraulics (section 4.4.3);
- The computational effort was shown to be considerable, primarily for the solution of the velocity-pressure coupling. The neutronics needs fewer iterations to reach a satisfying result (section 4.4.3);
- The integrated scheme for the coupling between neutronics and thermal-hydraulics gives an efficient management of the data common to the two fields (in comparison with external coupled schemes, that require external software for data transfer between neutronics and thermal-hydraulics). Moreover, the internal coupling developed in this work allows also all the variables to converge in a more coherent manner;
- The solution of all the fields on the same mesh is somewhat inefficient. In fact the neutronics could generally be solved on a coarser mesh. On the other hand, the thermal-hydraulics needs much finer meshes for regions of steep gradients, and for sufficient resolution of solid-fluid boundaries;
- Parallelization of this kind of calculations is potentially advantageous, depending on the decomposition methods and the hard-ware architecture (section 4.4.8).

### 5.3 Future work

Future work would primarily focus on the extension of the methodology in such a manner that more realistic cases can be simulated. For instance, as mentioned in the introduction of the thesis, this kind of high-resolution simulations can be useful to study nuclear BWR fuel assembly performances, as radial void fraction distribution can have non-negligible effects. Therefore, two-phase models should be included. Besides, in order to investigate more realistic fuel assemblies with spacers, non-uniform fuel rods, etc. more sophisticated models would be necessary.

The numerical algorithm applied is such that the neutronic and thermal-hydraulic fields are solved separately with a sparse matrix solver. Consequently, the amount of memory is limited to one matrix at a time. On the other hand, the development of solution strategies using block matrices, including many fields in the same (still sparse) matrix system, is attractive since this approach potentially allows faster convergence between the fields. This approach would however increase the peak memory.

In addition, as discussed above, the same mesh is used for the two fields. It could be beneficial to study methods for determining optimal meshes with respect to the neutronics and to the thermal-hydraulics, and using different meshes for the two fields.



# References

- [1] OECD. *OECD Factbook 2011: Economic, Environmental and Social Statistics*. OECD, 2011. ISBN: 978-92-64-11150-9.
- [2] World Nuclear Association. *The Nuclear Renaissance*. 2011. URL: <http://www.world-nuclear.org/info/inf104.html> (visited on 04/16/2012).
- [3] Pau L. Joskow and John E. Parsons. *A domain decomposition method for the advection-diffusion equation*. Version Revision February 7, 2012. 2012.
- [4] IAEA. *IAEA/PRIS - Power Reactor Information System*. 2012. URL: <http://pris.iaea.org/public/> (visited on 04/16/2012).
- [5] Informationskreis Kernenergie. *Reaktortypen - Worum geht es?* 2012. URL: <http://www.kernfragen.de/kernfragen/technik/04-Reaktortypen/4-01-Worum-geht-es.php> (visited on 04/17/2012).
- [6] Christophe Demaziere. *Physics of Nuclear Reactors. Lecture Notes*. Chalmers University of Technology, 2010.
- [7] Purdue Advanced Reactor Core Simulator. *PARCS: Computer code*. URL: <https://engineering.purdue.edu/PARCS> (visited on 04/26/2012).
- [8] Idaho National Library. *RELAP5*. 2012. URL: <http://www.inl.gov/relap5/> (visited on 04/26/2012).
- [9] Kostadin Ivanov and Maria Avramova. Challenges in coupled thermal-hydraulics and neutronics simulations for LWR safety analysis. *Annals of Nuclear Energy* **34** (2007).
- [10] *SIMULATE-3 Methodology. Advanced Three-Dimensional Two-Group Reactor Analysis Code*. Version STUDSVIK/SOA-95/18. Studsvik, 1995.
- [11] Christophe Demaziere. *Modelling of Nuclear Reactors. Lecture Notes*. Chalmers University of Technology, 2011.
- [12] Dave Knott and Akio Yamamoto. *Handbook of Nuclear Engineering. Chapter 9: Lattice Physics Computations*. Springer Science+Business Media LLC, 2010.
- [13] Alain Hebert. *Handbook of Nuclear Engineering. Chapter 8: Multigroup Neutron Transport and Diffusion Computations*. Springer Science+Business Media LLC, 2010.
- [14] Paul J. Turinsky. Advances in multi-physics and high performance computing in support of nuclear reactor power systems modeling and simulation. *Nuclear Engineering and Technology* **44.2** (2012).
- [15] Oak Ridge National Center for Computational Sciences. *Oak Ridge National Laboratory - Computing resources*. 2012. URL: <http://info.nccs.gov/computing-resources/> (visited on 04/17/2012).
- [16] F. Jatuff et al. Effects of void uncertainties on the void reactivity coefficient and pin power distributions for a 10x10 BWR assembly. *Annals of Nuclear Energy* **33** (2006).

## REFERENCES

---

- [17] T. Ikehara et al. Effect of Subchannel Void Fraction Distribution on Lattice Physics Parameters for Boiling Water Reactor Fuel Bundles. *Journal of Nuclear Science and Technology* **45** (2008).
- [18] David P. Weber et al. High-Fidelity Light Water Reactor Analysis with the Numerical Nuclear Reactor. *Nuclear Science and Engineering* **155** (2006).
- [19] Justin W. Thomas, Volkan Seker and Thomas J. Downar. Reactor Physics Simulations with Coupled Monte Carlo Calculation and Computational Fluid Dynamics (2007).
- [20] Jianwei Hu and Rizwan-uddin. Coupled Neutronics and Thermal-Hydraulics Using MCNP and FLUENT. *Transactions of the American Nuclear Society* **98** (2008).
- [21] Jeffrey Neil Cardoni and Rizwan-uddin. Nuclear Reactor Multi-Physics Simulations with Coupled MCNP5 and STAR-CCM+. *International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering* (2011).
- [22] *OpenFOAM. User Guide*. Version 2.0.0. OpenFOAM Foundation, 2011.
- [23] *OpenFOAM. Programmer's Guide*. Version 2.1.0. OpenFOAM Foundation, 2011.
- [24] *PSG2/Serpent - a Continuous-energy Monte Carlo Reactor Physics Burnup Calculation Code*. VTT Technical Research Centre of Finland, Sept. 2011.
- [25] Jaako Leppänen. "Development of a New Monte Carlo Reactor Physics Code". Thesis. VTT, 2007.
- [26] *MCNP - A General Monte Carlo N-Particle Transport Code. LA-UR-03-1987*. Version Revised 10/3/03. Los Alamos National Library, 2003.
- [27] Ronald L. Panton. *Incompressible flow*. Third edition. Wiley, 2005. ISBN: 0-4761-26122-X.
- [28] H. K. Versteeg and W. Malasekera. *An introduction to Computational Fluid Dynamics*. Second edition. Prentice Hall, 2007. ISBN: 978-0-13-127498.
- [29] Lars Davidsson. *An introduction to Turbulence Models*. Chalmers University of Technology, Nov. 2003.
- [30] *ANSYS CFX-Solver Theory Guide*. Version Release 12.0. ANSYS, Inc., 2009.
- [31] *Salome. The Open Source Integration Platform for Numerical Simulation*. Open CASCADE, 2012. URL: <http://www.salome-platform.org/user-section/online-documentation>.
- [32] Erik Brakkee and Peter Wilders. *A domain decomposition method for the advection-diffusion equation*. Technische Universiteit Delft, 1994.
- [33] *OpenFOAM extensions*. 2012. URL: <http://sourceforge.net/projects/openfoam-extend/> (visited on 05/10/2012).
- [34] Suhas V. Patankar. *Numerical Heat Transfer and Fluid Flow*. McGraw Hill, 1980.
- [35] T-2 Nuclear Information Service. *ENDF/B-VII Incident-Neutron Data*. 2007. URL: <http://t2.lanl.gov/data/neutron7.html> (visited on 04/20/2012).
- [36] *Python Programming Language*. 2012. URL: <http://www.python.org/> (visited on 05/19/2012).
- [37] National Institute of Standards Technology. *Thermophysical Properties of Fluid Systems*. 2011. URL: <http://webbook.nist.gov/chemistry/fluid/>.
- [38] Christophe Demaziere. Reactor Physics Calculations on MOX fuel in Boiling Water Reactors (BWRs). *Proc. 7th Information Exchange Meeting on Partitioning and Transmutation (IEMPT7)* (2002).



- 
- [39] S.G. Popov V.K. Ivanov J.J. Carbajo G.L. Yoder. *Thermophysical Properties of MOX and UO<sub>2</sub> Fuels Including the Effects of Irradiation*. Technical report. Oak Ridge National Laboratory, 2000. URL: <http://www.ornl.gov/~webworks/cpr/v823/rpt/109264.pdf>.
  - [40] International Atomic Energy Agency. *Thermophysical properties database of materials for light water reactors and heavy water reactors*. Technical report. 2006. URL: [http://www-pub.iaea.org/MTCD/publications/PDF/te\\_1496\\_web.pdf](http://www-pub.iaea.org/MTCD/publications/PDF/te_1496_web.pdf).
  - [41] *ANSYS Fluent Theory Guide*. Version Release 14.0. ANSYS, Inc., Nov. 2011.
  - [42] Journal of Fluids Engineering. *Editorial Policy Statement on the Control of Numerical Accuracy*. URL: <http://journaltool.asme.org/Content/JournalDescriptions.cfm?journalId=9&Journal=FE> (visited on 04/27/2012).
  - [43] J.K. Fink. Thermophysical properties of uranium dioxide. *Journal of Nuclear Materials* **279** (2000).
  - [44] Yun Je-Cho et al. Experimental study on measurement of emissivity for analysis on SNU-RCCS. *Nuclear Engineering and Technology* **38** (2006).
  - [45] E.V. Murphy et al. Emissivity of zirconium alloys in air in the temperature range 100 – 400 °C. *Journal of Nuclear Materials* **60** (1976).
  - [46] *MPI-2: Extensions to the Message-Passing Interface. Standard*. Version MPI 2.1. Message Passing Interface Forum, 2008. URL: <http://www.mpi-forum.org/docs/mpi2-report.pdf> (visited on 11/20/2011).

## REFERENCES

---

# Acknowledgements

I would like to express my gratitude towards Professor Christophe Demazière and Assistant Professor Paolo Vinai, who have both supervised and supported my work<sup>1</sup>. Thank you Paolo for all your good ideas and discussions on the work, and especially for all the time and effort you have spent helping me to write the final report. I have appreciated your advises and discussions on the academic world. Thank you Christophe for all the inspiring discussions and always taking time to listen to all my (impossible) ideas. Furthermore, thank you for your courses within the Master Programme in Nuclear Engineering. These were major sources of inspiration for this thesis work. Thank you both for always having an open door, taking time for my questions and very much believing in my work.

Furthermore, I would like to give thanks to the Division of Nuclear Engineering. I enjoyed the coffee, the floor ball, the chess and the discussions with all of you. Thanks for a great Master's Programme.

Finally, thank you Cornelia for your love, support and patience!

---

<sup>1</sup>Officially Assistant Professor Paolo Vinai has been the supervisor, and Professor Christophe Demazière the examiner.

## ACKNOWLEDGEMENTS

---

# Appendix A

## OpenFOAM overview and usage

This chapter is a brief overview of the basics of OpenFOAM from the perspective of the current work. The chapter is aimed to introduce the implementation of the thesis work, and to give the interested reader a glance of the structure and potential use of OpenFOAM.

### A.1 OpenFOAM overview

OpenFOAM (**Open Field Operation and Manipulation**) is a C++ library aimed at solving field equations[22]. The code consists partly of a general framework for solving partial differential equations (PDE) in a numerical manner (both using the finite element approach and the finite volume approach, although only the later has been utilized in this work), and partly of a set of example solvers. The first part contains all general structures needed to solve the PDEs, among other including mesh applications, standard boundary conditions, matrix solvers, general physical models and discretization procedures. The second part includes the OpenFOAM implementations of many of the standard CFD algorithms, including SIMPLE, PISO and Euler-Lagrangian solvers.

The library of code is written in an object oriented manner, meaning that relevant entities like boundary conditions, meshes, physical models (including turbulence and radiation models) are formulated as separate units. This gives a modular code, allowing new models to be defined from scratch, or by extending existing models. In this thesis one example of such a structure is the implemented physical model of temperature dependent parameters (see section 4.1). Two new thermophysical models have been implemented, one for the fluid regions and one for the solid regions in the nuclear reactor core, both based on a general basic OpenFOAM module for thermophysical models.

The use of object orientation within the library also allows a run-time selection of, e.g., discretization schemes and boundary conditions. This means that a developed solver could be used for many different setups and cases, by new choices of the run-time selectable parts, without the need to rewrite or recompile the code. In this thesis, this is e.g. utilized to come up with optimal choices of the matrix solvers (see table 4.3).

Another central part of the implementation of OpenFOAM solvers is the mimic of mathematical writing used to specify partial differential equations, which is further discussed in the implementation of the new solver (section A.2).

The OpenFOAM user interface for specifying and running cases is not based on any graphical user interface. Instead OpenFOAM case specification is based on a file hierarchy system. This hierarchy is placed within a root folder, in turn containing subfolders with specification of mesh and physical models (**constant**), system specifications like matrix solvers, convergence criteria and discretization schemes (**system**) and most often an initial guess of all fields, including boundary conditions, specified in one time step subfolder (e.g 0). An example of the structure used for the 2D fuel pin application (section 4.4) is seen in figure A.1.

The input structure lends itself well for scripting. This is in the current work used to test parameters (like in the parallelization test in section 4.4.8 and the power dependence in section 4.4.5) and to setup cases (as for the script for the 3D application of the 5x5 assembly, briefly described in section 4.5).

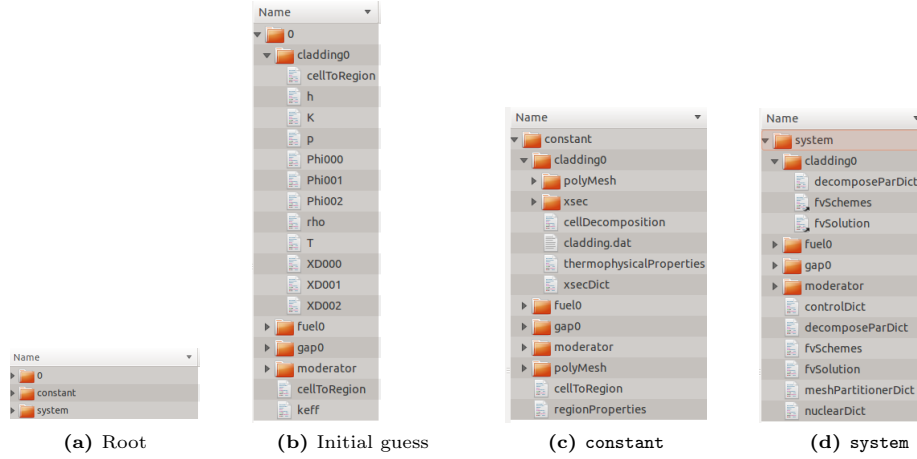


Figure A.1: Example of input structure for the solver developed in this thesis.

## A.2 Implementation of the new solver

To implement a new solver in OpenFOAM a few general steps will be followed, as outlined below. In the specific, more steps, e.g. including header files, specifying namespaces etc., are needed. Such typical procedures are however common for any C++ object oriented code, and thus left out from the presentation.

**1. Initiate mesh and fields:** The first step in the implementation is to create the mesh and the physical fields and models which are to be used in the solver. As described above, these entities are in general objects.

**2. Define PDE:** Given the mesh and the fields specified on the mesh, the PDE is specified using the OpenFOAM equation format. This format includes mathematical differential operators (such as the Laplacian operator in the case of the temperature equation for the cladding eq. (2.67)) as well as basic arithmetic operations as  $+$ ,  $-$ ,  $/$ ,  $\times$  [23]. The specified PDE will be discretized according to the discretization schemes specified in the input file `system/fvSchemes`. The very general equation format allows any, not only CFD equations, to be implemented in new solvers. This is e.g. used for the neutronics equations implemented for the coupled solver (section 3.3.2).

**3. Solve PDE** The next step is to solve the specified PDE. In the simplest case, this is performed automatically, using the specified matrix solvers (`system/fvSolution`) applied on the discretized (sparse) matrix from the previous step.

**4. Iteration and convergence control** Most solvers will need to be run in an iterative manner, as is e.g. the case with the coupled neutronics and thermal-hydraulics solver in the current work (section 3.3.3). For steady-state runs, the end of the iteration needs to be specified by convergence criteria, and for time dependent runs also time-stepping needs to be defined. The new solver will need to define what structures are to be checked for convergence, e.g. the matrix residual of one or multiple fields can be used.

The general structure is not fully applicable for all situations, e.g. not in the coupled problem, where coupled matrices are used, as discussed in section 3.3.3. Still, it can give an idea of the

potential versatility of OpenFOAM, and the way new problems can be specified within the framework, avoiding new code for common steps, such as the discretization, matrix solvers, etc.

Except for the implementation of new solvers, new physics will often need new physical models used by the solver. An example of this is the implemented cross-section model, using the sets of cross-sections pre-calculated by SERPENT (see section 3.2).

# Appendix B

## Materials

### B.1 Serpent material compositions

The isotopes are identified by atomic number (XX) and mass number (YYY) as XXYYY. The atomic fraction or mass fraction entered in Serpent must not be normalized.

**Table B.1:** Isotopic composition of gap, moderator and cladding

(a) Isotopic composition of helium used in the gap ( $\rho = -1 \times 10^{-3} \text{ g} \cdot \text{cm}^{-3}$ ).		(b) Isotopic composition of borated water used for moderator ( $\rho$ varied). Amount of boron corresponds to 1000 ppm of natural boron	
Isotope	Atomic fraction	Isotope	Mass fraction
08016	1.0	1001	1.120977e-01
		8016	8.879023e-01
		5010	1.990000e-04
		5011	8.010000e-04
(c) Isotopic composition of Zircaloy-2, 98.5% natural zirconium and 1.5% natural tin, with $\rho = 6.6 \text{ g} \cdot \text{cm}^{-3}$ [40]			
Isotope	Mass fraction		
50112	1.455000e-04		
50114	9.900000e-05		
50115	5.100000e-05		
50116	2.181000e-03		
50117	1.152000e-03		
50118	3.633000e-03		
50119	1.288500e-03		
50120	4.887000e-03		
50122	6.945000e-04		
50124	8.685000e-04		
40090	5.067825e-01		
40091	1.105170e-01		
40092	1.689275e-01		
40094	1.711930e-01		
40096	2.758000e-02		



**Table B.2:** Isotopic compositions of UOX and MOX fuels. Same density,  $\rho = 10.3\text{g}\cdot\text{cm}^{-3}$ , used for all fuels. Plutonium isotopic composition corresponds to reactor-grade plutonium[38].

(a) UOX fuel with 2% enrichment.

Isotope	Atomic fraction
92235	8.510638e-05
92238	4.117647e-03
8016	8.405507e-03

(b) UOX fuel with 4% enrichment.

Isotope	Atomic fraction
92235	1.702128e-04
92238	4.033613e-03
8016	8.407652e-03

(c) MOX fuel with 4% enrichment.

Isotope	Atomic fraction
94238	2.268908e-06
94239	4.686192e-05
94240	2.158333e-05
94241	6.721992e-06
94242	6.033058e-06
92235	2.919149e-05
92238	4.088824e-03
8016	8.402968e-03

(d) MOX fuel with 4% enrichment.

Isotope	Atomic fraction
94238	4.537815e-06
94239	9.372385e-05
94240	4.316667e-05
94241	1.344398e-05
94242	1.206612e-05
92235	2.859574e-05
92238	4.005378e-03
8016	8.401825e-03

# Appendix C

## Code

### C.1 Serpent

Example input file for Serpent generated by XSER

```
% —— Basic settings:
set title "Uniform temp- Region 0"
set acelib "/home/klas/Program/Serpent/xsdata/endfb7/krj-serpent_endfb7_20120123"
set bc 2
set sym 8
set pop 5000 500 20
set nfg 3 6.250e-07 1.000e-02

% —— Include materials file (with adjusted temperatures)
include "../material/mat.test.adjusted" % —— Surfaces:
surf 0 cyl 0.000 0.000 2.050e-01
surf 1 cyl 0.000 0.000 2.899e-01
surf 2 cyl 0.000 0.000 3.551e-01
surf 3 cyl 0.000 0.000 4.100e-01
surf 4 cyl 0.000 0.000 4.200e-01
surf 5 cyl 0.000 0.000 4.800e-01
surf 6 cyl 0.000 0.000 5.162e-01
surf 7 cyl 0.000 0.000 5.525e-01
surf 8 cyl 0.000 0.000 5.887e-01
surf 9 sqc 0.000 0.000 6.250e-01

%% —— Cells:

%% Regions NOT for group constants
cell 1 11 void -0
cell 3 11 MOXT540.00 0 -1
cell 5 11 MOXT540.00 1 -2
cell 7 11 MOXT540.00 2 -3
cell 9 11 airT540.00 3 -4
cell 11 11 claddingT540.00 4 -5
cell 13 11 waterT540.00D0.77 5 -6
cell 15 11 waterT540.00D0.77 6 -7
cell 17 11 waterT540.00D0.77 7 -8
cell 19 11 waterT540.00D0.77 8 -9
cell 21 11 outside 9
cell 22 12 outside 9

%% Regions for group constants
cell 2 12 MOXT540.00 -0
cell 4 12 void 0 -1
cell 6 12 void 1 -2
cell 8 12 void 2 -3
cell 10 12 void 3 -4
cell 12 12 void 4 -5
cell 14 12 void 5 -6
cell 16 12 void 6 -7
cell 18 12 void 7 -8
cell 20 12 void 8 -9
set gcu 12
```

```

%% Fill universe 10 with wanted region and surrounding
cell 23 10 fill 12 -0
cell 24 10 fill 11 0
cell 25 0 fill 10 -9
cell 26 0 outside 9

```

Listing C.1: Example of a Serpent input file generated by XSER

**Example material file for Serpent generated by XSER**

```

%—— Temperature adjusted material
mat MOXT540.00 -10.3 tmp 540.00
  94238.03c 2.26891e-06
  94239.03c 4.68619e-05
  94240.03c 2.15833e-05
  94241.03c 6.72199e-06
  94242.03c 6.03306e-06
  92235.03c 1.04255e-05
  92238.03c 4.10735e-03
  8016.03c 8.40249e-03

%—— Temperature adjusted material
mat airT540.00 -1e-3 tmp 540.00
  8016.03c 1.00000e+00

%—— Temperature adjusted material
mat claddingT540.00 4.004642E-02 tmp 540.00
  40090.03c 3.95500e-02
  26056.03c 1.38300e-04
  24052.03c 7.07200e-05
  8016.03c 2.87400e-04

%—— Temperature adjusted material
mat waterT540.00D0.77 -0.770 moder lwtr550.00 1001 tmp 540.00
  1001.03c 4.72400e-02
  5010.03c 8.64100e-06
  5011.03c 3.46900e-05
  8016.03c 2.36200e-02
therm lwtr550.00 lwe7.10t

%—— Temperature adjusted material
mat MOXT560.00 -10.3 tmp 560.00
  94238.03c 2.26891e-06
  94239.03c 4.68619e-05
  94240.03c 2.15833e-05
  94241.03c 6.72199e-06
  94242.03c 6.03306e-06
  92235.03c 1.04255e-05
  92238.03c 4.10735e-03
  8016.03c 8.40249e-03

%—— Temperature adjusted material
mat airT560.00 -1e-3 tmp 560.00
  8016.03c 1.00000e+00

%—— Temperature adjusted material
mat claddingT560.00 4.004642E-02 tmp 560.00
  40090.03c 3.95500e-02
  26056.03c 1.38300e-04
  24052.03c 7.07200e-05
  8016.03c 2.87400e-04

%—— Temperature adjusted material
mat waterT560.00D0.75 -0.752 moder lwtr550.00 1001 tmp 560.00
  1001.03c 4.72400e-02
  5010.03c 8.64100e-06
  5011.03c 3.46900e-05
  8016.03c 2.36200e-02

%—— Temperature adjusted material
mat MOXT900.00 -10.3 tmp 900.00
  94238.09c 2.26891e-06
  94239.09c 4.68619e-05
  94240.09c 2.15833e-05
  94241.09c 6.72199e-06
  94242.09c 6.03306e-06
  92235.09c 1.04255e-05

```

```
92238.09c 4.10735e-03
8016.09c 8.40249e-03

%— Temperature adjusted material
mat airT650.00 -1e-3 tmp 650.00
8016.06c 1.00000e+00

%— Temperature adjusted material
mat claddingT600.00 4.004642E-02 tmp 600.00
40090.06c 3.95500e-02
26056.06c 1.38300e-04
24052.06c 7.07200e-05
8016.06c 2.87400e-04

%— Temperature adjusted material
mat waterT570.00D0.73 -0.732 moder lwtr550.00 1001 tmp 570.00
1001.03c 4.72400e-02
5010.03c 8.64100e-06
5011.03c 3.46900e-05
8016.03c 2.36200e-02

%— Temperature adjusted material
mat MOXTi000.00 -10.3 tmp 1000.00
94238.09c 2.26891e-06
94239.09c 4.68619e-05
94240.09c 2.15833e-05
94241.09c 6.72199e-06
94242.09c 6.03306e-06
92235.09c 1.04255e-05
92238.09c 4.10735e-03
8016.09c 8.40249e-03

%— Temperature adjusted material
mat airT900.00 -1e-3 tmp 900.00
8016.09c 1.00000e+00

%— Temperature adjusted material
mat claddingT700.00 4.004642E-02 tmp 700.00
40090.06c 3.95500e-02
26056.06c 1.38300e-04
24052.06c 7.07200e-05
8016.06c 2.87400e-04

%— Temperature adjusted material
mat waterT580.00D0.71 -0.712 moder lwtr600.00 1001 tmp 580.00
1001.03c 4.72400e-02
5010.03c 8.64100e-06
5011.03c 3.46900e-05
8016.03c 2.36200e-02
therm lwtr600.00 lwe7.12t

%— Temperature adjusted material
mat MOXTi200.00 -10.3 tmp 1200.00
94238.12c 2.26891e-06
94239.12c 4.68619e-05
94240.12c 2.15833e-05
94241.12c 6.72199e-06
94242.12c 6.03306e-06
92235.12c 1.04255e-05
92238.12c 4.10735e-03
8016.12c 8.40249e-03

%— Temperature adjusted material
mat airT950.00 -1e-3 tmp 950.00
8016.09c 1.00000e+00

%— Temperature adjusted material
mat claddingT740.00 4.004642E-02 tmp 740.00
40090.06c 3.95500e-02
26056.06c 1.38300e-04
24052.06c 7.07200e-05
8016.06c 2.87400e-04

%— Temperature adjusted material
mat waterT590.00D0.69 -0.688 moder lwtr600.00 1001 tmp 590.00
1001.03c 4.72400e-02
5010.03c 8.64100e-06
5011.03c 3.46900e-05
8016.03c 2.36200e-02
```

```

%—— Temperature adjusted material
mat MOXT1500.00 -10.3 tmp 1500.00
  94238.15c 2.26891e-06
  94239.15c 4.68619e-05
  94240.15c 2.15833e-05
  94241.15c 6.72199e-06
  94242.15c 6.03306e-06
  92235.15c 1.04255e-05
  92238.15c 4.10735e-03
  8016.15c 8.40249e-03

%—— Temperature adjusted material
mat airT1000.00 -1e-3 tmp 1000.00
  8016.09c 1.00000e+00

%—— Temperature adjusted material
mat claddingT800.00 4.004642E-02 tmp 800.00
  40090.06c 3.95500e-02
  26056.06c 1.38300e-04
  24052.06c 7.07200e-05
  8016.06c 2.87400e-04

%—— Temperature adjusted material
mat waterT600.00D0.66 -0.661 moder lwtr600.00 1001 tmp 600.00
  1001.06c 4.72400e-02
  5010.06c 8.64100e-06
  5011.06c 3.46900e-05
  8016.06c 2.36200e-02

%—— Temperature adjusted material
mat MOXT1800.00 -10.3 tmp 1800.00
  94238.18c 2.26891e-06
  94239.18c 4.68619e-05
  94240.18c 2.15833e-05
  94241.18c 6.72199e-06
  94242.18c 6.03306e-06
  92235.18c 1.04255e-05
  92238.18c 4.10735e-03
  8016.18c 8.40249e-03

%—— Temperature adjusted material
mat airT1300.00 -1e-3 tmp 1300.00
  8016.12c 1.00000e+00

%—— Temperature adjusted material
mat claddingT1100.00 4.004642E-02 tmp 1100.00
  40090.09c 3.95500e-02
  26056.09c 1.38300e-04
  24052.09c 7.07200e-05
  8016.09c 2.87400e-04

%—— Temperature adjusted material
mat waterT620.00D0.59 -0.594 moder lwtr600.00 1001 tmp 620.00
  1001.06c 4.72400e-02
  5010.06c 8.64100e-06
  5011.06c 3.46900e-05
  8016.06c 2.36200e-02

```

Listing C.2: Example of a Serpent material file generated by XSER