

**Study to Improve the Physical
Modelling of the ACACIA
High Temperature Gas Cooled
Nuclear Reactor**

Graduation Thesis: **S.M. Verslype**

Graduation Professor:
Prof. dr. ir. H. van Dam
Supervisor:
Dr. ir. J.L. Kloosterman

Delft University of Technology
Faculty of Applied Sciences
Department of Applied Physics
Work performed at:
Interfaculty Reactor Institute
Department of Reactor Physics
March 2002

Summary

The ACACIA (AdvanCed Atomic Cogenerator for Industrial Applications) reactor is a gas cooled pebble bed reactor developed by NRG (A.I.van Heek 1997). To study the dynamic behaviour of this reactor, J.F. Kikstra developed a model of the ACACIA reactor in the Aspen Custom Modeller (ACM)(J.F.Kikstra 2001). The neutronics model in the Kikstra code is, in contrast to the modelling of the energy conversion system, rather limited. An attempt has been made to replace it by a more versatile multidimensional neutronics model. This involved two steps.

The first step has been to find a suitable neutron diffusion code to do the calculations. FX2 (R.A.Shober, T.A.Daly, & D.R.Ferguson 1978) has been selected and thoroughly tested to see if it was suitable to simulate the ACACIA reactor. Unfortunately FX2 does not take up scattering into account. Therefore all simulations, that have been done with one or more energy group boundaries below 3.3 eV, generate results that differ much from the results obtained by BOLD VENTURE (a neutron diffusion code) and KENO Va (a Monte Carlo neutron transport code). Calculations in which the energy group boundaries were selected in such a way that no up scattering occurred show good comparison between FX2 and BOLD VENTURE. Further research should be done to investigate if FX2 can generate realistic results with this 3.3 eV lower limit. On the other hand, the agreement between BOLD VENTURE and KENO Va confirms that a neutron diffusion code is suitable to simulate the behaviour of the ACACIA reactor.

The second step towards a more versatile neutronics model has involved improvement of the modelling of the reactor in ACM. The ACM model is not very transparent and hard to work with. Results of some smaller modifications are presented. These results show that under the transients considered, increase and decrease of the helium inlet temperature with 100 K, the temperature stays well below the fission products release temperature limit. The main modification was to model the reactor core in two dimensions in order to combine it with the two dimensional diffusion code. But in the end no results could be obtained from this core model due to convergence problems.

For future modifications to the ACM model, I recommend that the routines used to simulate the behaviour of the helium flow are simplified and made more stable. Since experience has shown that all convergence problems were related to the flow, this should be one of the first things to be changed in order to be able to make further improvements to the reactor model. Even completely replacing the ACM model of the core with a Fortran model is a serious option since ACM has proven to be time consuming for even the smallest modifications. The results presented in this thesis from the converging simulations of ACM support the proposition that the high temperature gas cooled pebble bed reactor is an inherently safe reactor for combined heat and power production.

Table of contents

Summary..... 2

1. Introduction..... 4

2. The High Temperature Pebble Bed Reactor 6

3. Generating a cross section library 9

3.1 Cross sections..... 9

3.2 Theory 11

 3.2.1 The CSAS code..... 11

 3.2.2 The BONAMI code 11

 3.2.3 The NITAWL-II code..... 12

 3.2.4 The XSDRNPM code 13

 3.2.5 The BOLD-VENTURE code..... 14

3.3 Model build-up 14

3.4 The Reactor Model..... 16

4. FX2, a 2 dimensional diffusion code..... 18

4.1 Diffusion theory 18

4.2 The FX2 code 19

4.3 Different calculations 21

4.4 BOLD-VENTURE and Keno Va 22

5. The Aspen Custom Modeller program 23

5.1 Aspen Custom Modeller 23

5.2 The helium flow 24

5.3 Solid elements 26

5.4 Neutronics 27

5.5 Modifications 30

 5.5.1 Two dimensional flow 30

 5.5.2 The helium cavity 31

 5.5.3 Friction correction 32

6. Results 33

6.1 FX2 results 33

6.2 Aspen Custom Modeller results 40

7 Conclusions and recommendations 47

7.1 The FX2 model 47

7.2 The ACM model 47

References..... 49

Appendix A, Heterogeneous meshes 51

Appendix B, Tables with the FX2 results 52

1. Introduction

In order to meet the ever-growing demand for electricity and international treaties, like the 1997 Kyoto Protocol and climate change treaty signed in Bonn, the use of nuclear energy is inevitable. Growth of the world population and more important the growth of the number of people that have access to electricity will be responsible for most of the increase, shown in figure 1.1. From figure 1.2, a dramatic increase in carbon dioxide production can be expected.

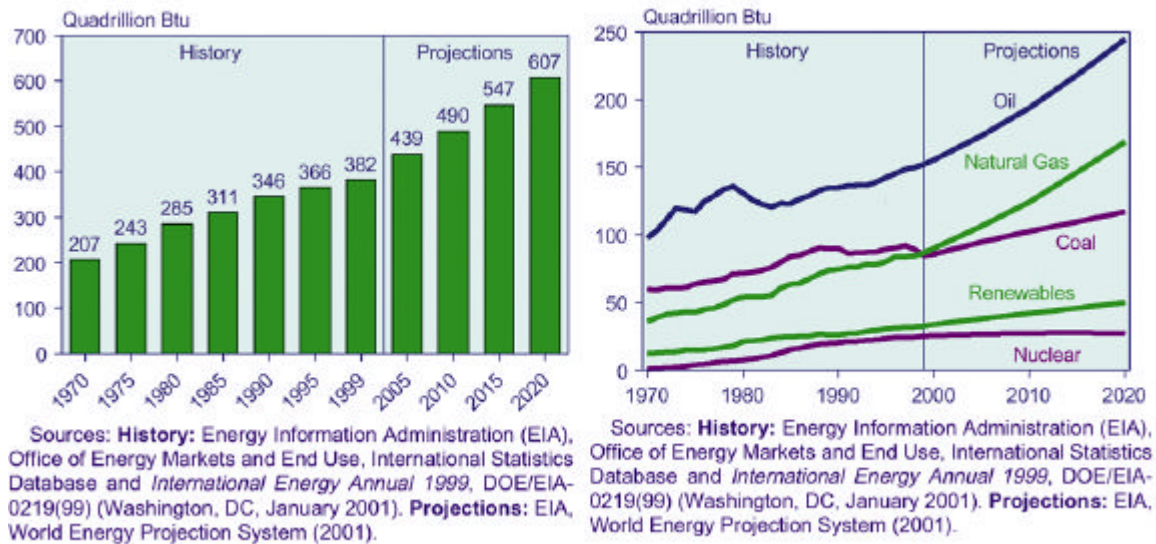


Figure 1.1; Projection of the worlds' energy use Figure 1.2 Projection of the worlds energy sources

At this moment only 0.5% of the natural uranium is used in a nuclear fission reactor. In order to make nuclear energy more sustainable, breeder reactors must be used to increase this percentage. A second option to make nuclear energy more sustainable is the reprocessing of nuclear waste. This has two functions: one is to reduce the time the waste is radioactive; the other is to recycle the fuel nuclides in the nuclear waste. The main disadvantage of nuclear fission energy production is the radioactivity of the waste, but also in this field many promising projects are being conducted.

Nuclear energy, or more precisely nuclear fission, is a very promising source of energy with many advantages. It does not produce any carbon dioxide, it is cheap and very reliable. However this source should be well controlled in order to prevent disasters like the accident in Chernobyl (1986). Therefore, resources are being committed to gain a better understanding of the behaviour of nuclear reactors during transients. This also gave rise to development of inherently safe reactors. Computer simulations give the opportunity to obtain an understanding of the behaviour of nuclear reactors during different and worst case scenarios.

The goal of this research is to improve an existing model of the ACACIA (AdvanCed Atomic Cogenerator for Industrial Applications) reactor. The ACACIA reactor is a gas cooled pebble bed reactor developed by NRG(A.I.van Heek 1997). The properties of this reactor, which will be explained in chapter 2, allow operation at high temperatures and result in an inherently safe reactor. Especially this last property makes this type of reactor very promising. Much research

has already been done on this type of reactor, among which the PhD work of Kikstra (J.F. Kikstra 2001), who developed a model of the ACACIA reactor. This model describes the energy conversion system very detailed, but uses a rather limited model to describe the nuclear heat generating part of the reactor. It uses, for example, a point kinetics model to simulate the time dependent behaviour of the reactor core during transients and a fixed power profile to distribute the power to the different pebble bed slabs. So we set off on a quest to improve the neutronics of this model and to improve the model of the reactor itself. The code used for this simulation is the Aspen Custom Modeller (ACM), which is not a nuclear computer code, but a so-called solver. We wanted to improve the neutronics of the ACM model by linking the ACM code with a 2 dimensional nuclear diffusion code, FX2. The reason we wanted to replace the point kinetic model with FX2 was that this would enable the simulation to work at a wider range of conditions. The FX2 code and the ACM code will be described respectively in chapters 4 and 5. The reason we selected FX2 is that it can interpolate between different cross section libraries generated at different temperatures. This opens the opportunity for FX2 to be combined with ACM.

In chapter 2 the ACACIA reactor will be discussed in more detail. Before we can use the FX2 code a cross section library must be generated. This turns out to be a rather complex procedure using different codes. This will be described in chapter 3. FX2 will be the subject of chapter 4. The original ACM model and the modifications will be discussed in chapter 5. In chapter 6 results of both the FX2 simulations as well as the ACM simulations will be presented. Finally in chapter 7 the conclusion and recommendations for further research are presented.

2. The High Temperature Pebble Bed Reactor

When the accident at Chernobyl occurred a stronger need arose for safer and even inherently safe nuclear reactors, which will contain its fissile material under all possible circumstances. The High Temperature Reactor can be considered inherently safe, due to the selection of moderator, coolant and fuel containment.

The type of HTR to our interest is the pebble bed reactor. In this type of reactor the fission materials are being contained in TRISO coated particles, figure 2.1. These particles in their turn are embedded in the graphite of a larger, 5 cm diameter, pebble. This pebble is enclosed in a carbon shell with a 0.5 cm radius. The TRISO coated particles are small particles (1 mm diameter) that consist of a uranium core, a porous layer of carbon and several layers of very hard pyrolytical carbon and silicon carbide. The porous layer will absorb the fission products release during burn up. The hard shell, which consists of silicon carbide and pyrolytic carbon, around that will keep all fission materials inside the particles and will not crack unless very high temperatures are reached. The reactor core consists of a carbon hollow cylinder filled with these pebbles. Through the gaps between these pebbles the coolant, helium, can flow.

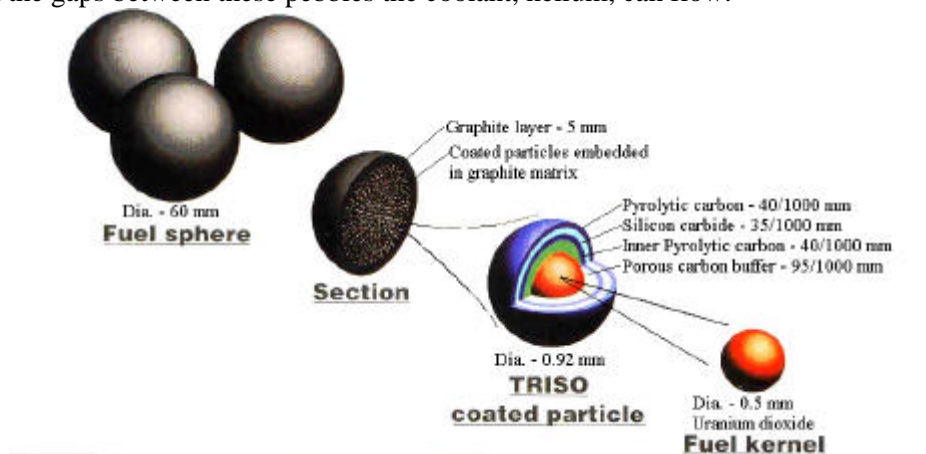


Figure 2.1; The TRISO coated particle and the pebble.

The HTR is a thermal reactor, which means that the neutrons, released at high energies by the fission process, are being slowed down (moderated) in order to cause fission again. This moderation is done by the carbon surrounding the TRISO coated particles and the core. The layers of carbon surrounding the core are called the reflectors. These reflectors have two functions. The first is to reflect the neutrons back into the core, thereby reducing the leakage. The second is to moderate the neutrons by absorbing their kinetic energy and thereby slowing them down. In most nuclear reactors the moderation is done by water. This material is also responsible for cooling, thus leading to a coupling of coolant and moderator. In HTR's these functions are separated. This makes it easier to simulate the reactor because the moderation is less temperature dependent. Next to its reflecting and moderating capabilities, carbon is a very good heat conductor with a large heat capacity. These properties are very welcome in our design. In case of an accident the good heat conductivity will transport the heat outside the core and the large heat capacity will make sure the temperature increase will be small.

As has been mentioned above, helium is used as a coolant. This gas does not have a significant influence on the behaviour of the neutrons and is chemically inert. The helium enters the reactor from below in the third reflector layer (number 4 in figure 2.2) before it enters the core from

above. For reactivity control, the HTR can use control rods in its reflectors. The legend of figure 2.2 shows that the inner reflector consists of carbon for only 85%; the rest is made of channels for the control rods.

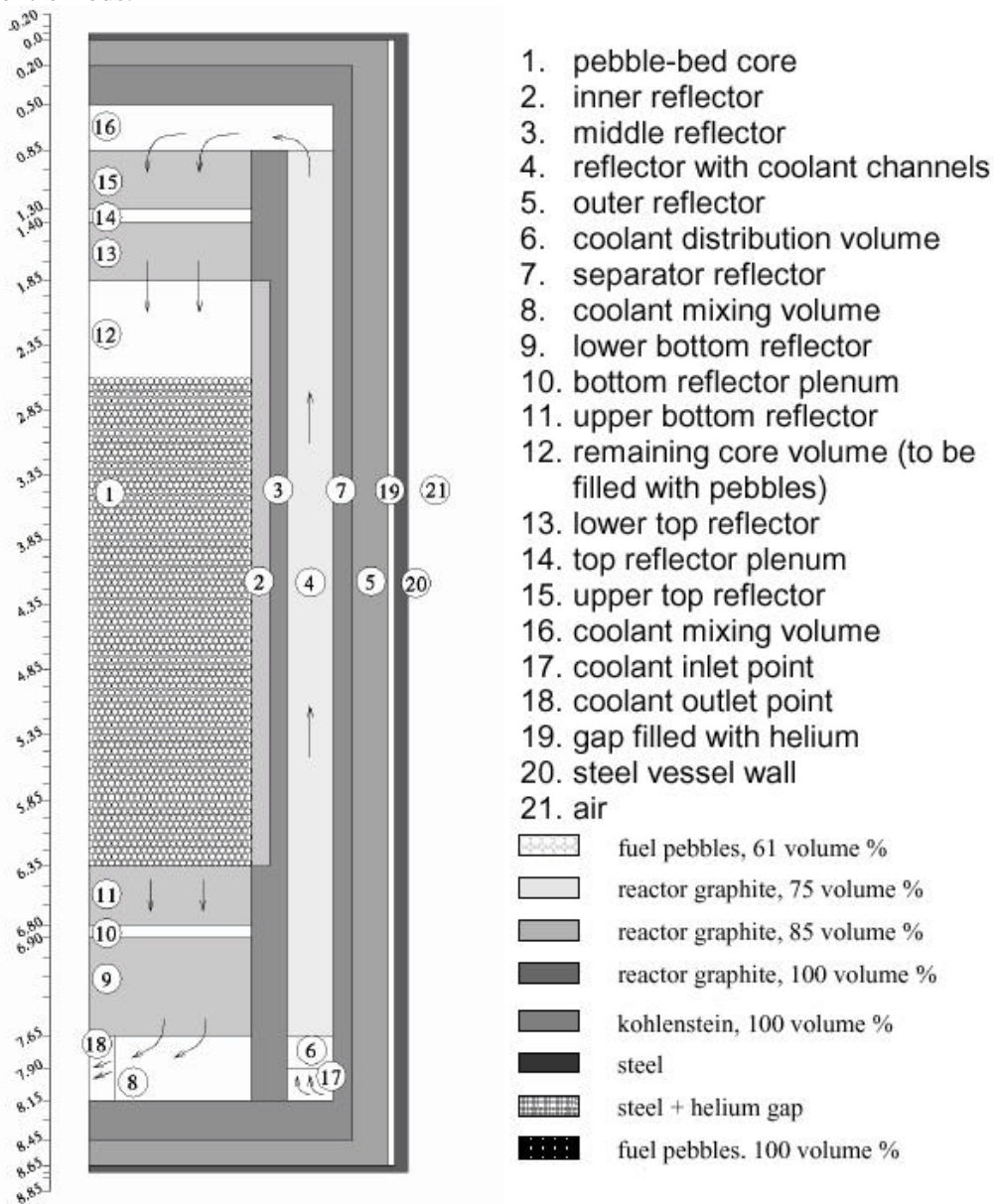


Figure 2.2; A schematic view of the ACACIA reactor(E.C.Verkerk 2000)

For the pebble bed reactor several fuel management strategies have been developed. The one used for the ACACIA reactor is the pue-à-pue schema. This means that the ACACIA starts with a large cavity above the pebble bed. During operation pebbles are added to the core, slowly filling the cavity above the core.

The ACACIA design uses a direct Brayton cycle in a closed cycle gas turbine, figure 1.3. The direct cycle uses the same gas as coolant in the reactor and as working gas in the energy conversion system. This is in contrast to other reactors where the heat is transferred to a

secondary cycle. One of the advantages of using one direct cycle is that it works at a higher temperature than an indirect cycle and thereby increases the efficiency. The main disadvantages are the small chance that fission products are transported through the system and that the Brayton cycle is relatively new. Therefore it can not be called “proven technology” which is an important argument in the social environmental discussion about nuclear energy.

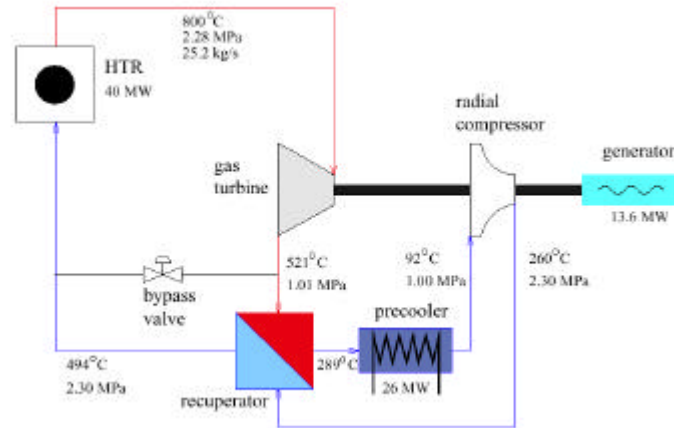


Figure 1.3; The closed direct Brayton cycle

In case of a transient, one of the worst things that can happen to a nuclear reactor is loss of coolant flow (LOFA). When this occurs the coolant stops flowing and the heat is no longer transported outside the core by forced convection. So can the core become hotter and hotter and finally melt? This will not happen in this case. With increasing temperature the fission process will drop due to resonance broadening and the increasing absorption of neutrons in the epithermal energy range. The resonance broadening is a property of the cross section that will be explained in chapter 3. This behaviour is described by a negative temperature coefficient of reactivity, which means that with increasing temperature the power production of the core will drop. Due to natural heat transport and the large heat capacity of the graphite the maximum temperature reached will remain well below the maximum temperature the pebbles and particles can resist. Due to the very thorough containment of the fission material the core will stay intact. So the combination of the confinement, the good heat conduction and the negative temperature coefficient makes this reactor inherently safe.

The type of pebble bed HTR we are simulating here is a small industrial cogenerator. The ACACIA design is a combined heat and power plant for industrial applications. The total power production is 40 MW_{th}. This means it is a very small reactor compared to commercial PWR's e.g., which in many cases have a total power production of 3GW_{th}. The use of the ACACIA reactor can be found for example at generating combined heat and power for a large factory.

The pebble bed is a very promising design, of which the basic concept stretches back to the forties (F.Daniels 1944). The combination of a high efficiency and inherently safe behaviour makes this type of reactor very worthwhile to consider. A test reactor was built in Julich, Germany in 1966. This reactor proved its inherent safety characteristics when the cooling was shut down without scrambling the reactor, that is inserting the control rods to force a shut down of the reactor (K.Kugeler & R.Schulten 1989). All temperatures stayed well below those at which the TRISO coated particles would burst. At this moment a new pebble bed HTR has been built in China, the HTR-10, and in South Africa a larger 110 MW_{th} pebble bed HTR is being developed.

3. Generating a cross section library

3.1 Cross sections

The goal of this research is to replace the point kinetics model of the ACM code by a neutronics code that is more detailed. One type of code that meets these demands is a neutron diffusion code. Before we can use a computer code that solves the neutron diffusion equations, an appropriate library of nuclear cross-sectional data has to be generated. A cross section library consists of a list of all the nuclides used in the model and their appropriate cross sections. These cross sections are divided into a number of energy groups spanning the typical energy range between 0 and 20 MeV. The energy dependence of the cross section can be seen in figure 3.1. One should pay attention to range of the cross section, that can vary between 0.001 and 10.000 barn (10^{-24} cm²).

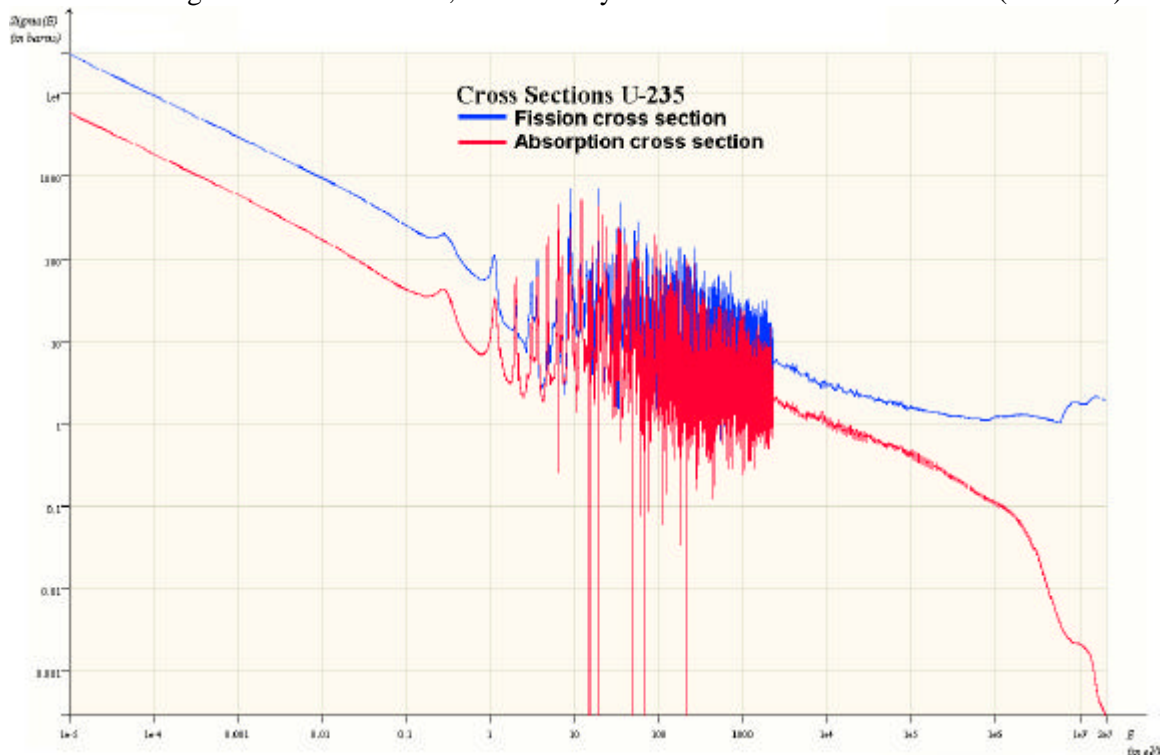


Figure 3.1; Cross sections as a function of energy of Uranium 235.

In figure 3.1 one can clearly see the resonances, the sharp peaks between the 1 and 100 keV. The position of a resonance corresponds with an energy level of the nuclide. The resonances are not sharp lines but have a certain width. This width is a function of the heat movement of the nuclide. With increasing temperature the width of the resonance will increase and the maximum value of the resonance will decrease. This is called the Dopplereffect and is an important property of nuclear reactors. In chapter 2 the resonance broadening was mentioned. Increasing the temperature results in a broader resonance and a higher average cross. And this leads with this type a reactor to more absorption of neutrons in uranium 238.

The part of the spectrum where resonances occur can be divided into two parts; The resolved and the unresolved part. The resolved part consists of distinguishable resonances. This is in contrast to the unresolved part where individual resonances are indistinguishable.

The cross sections are stored in a master library for a large number of nuclides. Next to the cross sections this master library also contains information about the resonances. The master library is the source for all the information about cross sections for the problem dependent cross section library that one wants to generate.

One of the goals of this research is to find the lowest number of energy groups for which simulations generate similar results compared to more detailed calculations. When condensing is applied, that is reducing the number of energy groups, in a simulation, the cross sections are modified in such a way that the total reaction rate remains constant.

$$\mathbf{s}_g = \frac{\int dE \sigma(E) \phi(E)}{\int dE \phi(E)} \quad (3.1)$$

Here σ represents the microscopic, density independent, cross section and ϕ the neutron flux. In this equation g represents the number of energy groups that are collapsed. If we want to simplify the reactor geometry we have to apply collapsing over different zones in a similar way

$$\bar{\mathbf{s}}_g = \frac{\sum_j^{IZM} N^j \mathbf{s}_g^j \int \mathbf{f}_g^j(\bar{r}) d\bar{r}}{N \sum_j^{IZM} \int \mathbf{f}_g^j(\bar{r}) d\bar{r}} \quad (3.2)$$

Here IZM is the number of zones, N^j stands for the atomic density in zone j and N for the average density in the cell. The number of energy groups and the number of zones in the reactor model plays an important role in the CPU time needed to solve the diffusion calculation. Therefore the objective is to reduce the number of energy groups as much as possible and work with a highly simplified model of the geometry. For each number of energy groups we run a simulation and for each geometry model a separate cross section library is needed. Of course these simulations must show good comparison with more detailed models.

The calculations are done in five sequential steps. This is a necessity due to the fact that some steps need the input of the previous one and due to computational restrictions. An overview of the calculations can be seen in figure 3.2. In order to obtain validation for future simulations, the data used is the same as Verkerk used in his simulations (E.C. Verkerk 2000). The goal is a library of cross-sectional data for a homogenised reactor core and its reflectors. We use the CSAS (N.F. Landers & L.M. Petrie 1998) code, which calls NITAWL II (N.M. Greene, L.M. Petrie, & R.M. Westfall 1998), BONAMI (N.M. Greene 1998) and the XSDRNPM (N.M. Greene & L.M. Petrie 1998) codes, to take into account the temperature dependent resonance shielding. Next to CSAS we use separate XSDRNPM calculations for homogenisation and collapsing. Finally we use BOLD VENTURE (D.R. Vondy, T.B. Fowler, & G.W. Cunningham III 1981) and WAX (N.M. Greene et al. 1992) to merge the different libraries. These codes will be discussed next.

3.2 Theory

3.2.1 The CSAS code

As mentioned above the CSAS code calls three other codes. The CSAS driver prepares input data for the codes it calls to run. The code itself does not do any neutronics calculations. It simply calls other codes for which only one input file has to be written. This makes the CSAS driver code relatively user friendly. The code has several options for different sequences of codes that should be called. The option we are interested in calls respectively BONAMI, NITAWL and XSDRNPM. The first two codes account for the temperature dependency of the cross sections. All cross sections will be generated at a temperature of 920 K of the carbon. Calculations done by Bende have shown an average temperature difference of 36 K between the fuel kernel and the temperature of the graphite pebble (E.E. Bende 2000). So all fuel nuclide cross sections will be calculated at a temperature of 956 K.

3.2.2 The BONAMI code

BONAMI stands for BONdarenko AMPX Interpolator. This is a code that accesses an AMPX library with the Bondarenko factors to calculate resonance shielding using the Bondarenko method (I.I. Bondarenko 1964). The major advantages of the Bondarenko method are its simplicity and its speed. It is used for the resonances in the unresolved part of the spectrum (~2keV–100keV). The BONAMI code starts with equation 3.1. In this equation the flux is the unknown parameter. The BONAMI code uses a simple expression for the flux.

$$f(u) \sim \frac{1}{\Sigma_t(u)} \quad (3.3)$$

In this equation S_t is the total macroscopic cross-section. In this equation the energy is replaced by the lethargy.

$$u(E) = \ln \frac{E_0}{E} \quad (3.4)$$

The lethargy is used because it is logarithmic, a useful property as one will see later on. In this equation E_0 is a reference energy. With the approximation for the flux, equation 3.1 becomes

$$s_g = \frac{\int_g du \frac{s(u)}{\Sigma_t}}{\int_g du \frac{1}{\Sigma_t}} \quad (3.5)$$

If we rewrite the total macroscopic cross section as a sum of the cross section of the nuclide we are interested in and of all other nuclides, equation 3.5 becomes

$$s_g^i = \frac{\int_g du \frac{s^i(u)}{N_i s_t^i(u) + \sum_{j \neq i} N_j s_t^j(u)}}{\int_g du \frac{1}{N_i s_t^i(u) + \sum_{j \neq i} N_j s_t^j(u)}} \quad (3.6)$$

When we substitute the summation by

$$\mathbf{s}_0^i \equiv \sum_{j \neq i} N_j \mathbf{s}_t^j(u) / N_i \quad (3.7)$$

We can simplify (3.6) to

$$\mathbf{s}_g^i = \frac{\int_g du \frac{\mathbf{s}^i(u)}{\mathbf{s}_t^i(u) + \mathbf{s}_0^i(u)}}{\int_g du \frac{1}{\mathbf{s}_t^i(u) + \mathbf{s}_0^i(u)}} \quad (3.8)$$

The Bondarenko method ignores some fine structure by using an effective constant value for s_0 within an energy group. This allows the Bondarenko method to precalculate cross-sections, at different s_0 's, without having to know the exact composition in which the cross-sections might be used. We can calculate the infinite dilution average ($\sigma_0^i = \infty$) by

$$\mathbf{s}_{g, ID}^i = \frac{\int_g du \mathbf{s}^i(u)}{\int_g du} \quad (3.9)$$

With the infinite dilution average the Bondarenko method calculates the collapsed cross section by

$$\overline{\mathbf{s}}_g(\mathbf{s}_0, T) \equiv F_g(\mathbf{s}_0, T) \overline{\mathbf{s}}_{g, ID} \quad (3.10)$$

In this equation $F_g(s_0, T)$ is the Bondarenko factor. This factor is calculated by interpolating in tables of precalculated data for different s_0 's and temperatures. For the most important nuclides these tables are generated at 5- 8 s_0 's and 5-8 temperatures. The s_0 's range from 10^{-3} to 10^4 in logarithmic steps and the temperature ranges from 300 to 2500 K. In this way the Bondarenko method only has to interpolate between precalculated tables at a given composition and temperature to obtain the collapsed cross sections.

3.2.3 The NITAWL-II code

The NITAWL code has two major functions. On basis of a user specified system description it calculates the problem dependent resonance shielding by applying the Nordheim Integral Treatment (L.W.Nordheim, G.Birkhoff, & E.P.Wigner 1998). Additionally the code reads the cross sections from an AMPX master library and produces a new AMPX working library.

The Nordheim Integral Treatment involves the numerical solution of the integral.

$$\mathbf{f}(E) \Sigma_T(E) = \sum_{i=1}^3 \left[((1 - P_0^*(E)) / \mathbf{a}) \int_E^{E/(1-\mathbf{a})} \mathbf{f}(E') \Sigma_{si}(E') \frac{dE'}{E'} + P_0^*(E) \Sigma_{Ti}(E) W(E) \right] \quad (3.11)$$

This is the integral for the collision density as a function of the energy. In this integral $i=1$ for the absorber, $i=2$ for the first admixed moderator and $i=3$ for the second admixed moderator. The presence of the lattice of fuel particles is accounted for by the Dancoff factor, P_0^* . This factor

represents the chance of neutron leaving the fuel region and having their first collision in another fuel zone. In our case this factor was 0.338¹.

The model defined by this integral involves several approximations

- 1) Each resonance is treated separately without consideration of other resonance nuclides in the system.
- 2) The algorithm uses the flat source approximation.
- 3) Neutron transport into and out of the absorber region can be treated with first flight escape probabilities.
- 4) The presence of other absorber lumps in the system can be accounted for by the Dancoff factor. This factor corresponds to the first flight transmission probability across the moderator.

These assumptions lead to some restrictions. Restrictions on the number of resonance isotopes in the material and restrictions on the geometry are the most important ones. Because we use a sphere geometry and an enriched uranium fuel core, these approximations are acceptable.

3.2.4 The XSDRNPM code

XSDRNPM is a discrete ordinates code that solves the one dimensional Boltzman equation for different geometries. The code assumes (in case of a slab geometry) that the system extends to infinity along the other two axes. The essential basis of the discrete ordinance is that the angular distribution of the neutron flux is evaluated in a number of discrete directions instead of using spherical harmonics. It is possible to obtain any degree of accuracy by evaluating enough directions. This S_n method is a linear approximation of the angular flux distribution μ . The S stands for segments and n gives the number of segments. Carlson(B.G.Carlson 1953) proposed this method. XSDRNPM uses a discrete mesh where it calculates the flux on each mesh cells boundary by solving

$$w_m \mathbf{m}_m (\mathbf{f}_{g,i+1,m} - \mathbf{f}_{g,i,m}) + w_m \mathbf{s}_{g,i+1/2} \mathbf{f}_{g,i+1/2,m} (x_{i+1} - x_i) = w_m \mathbf{S}_{g,i+1/2,m} (x_{i+1} - x_i) \quad (3.12)$$

in which μ_m stands for the angular direction of quadrature m , g for the energy group and i for the mesh cell. The code solves this equation for each quadrature, energy group and mesh cell.

XSDRNPM has two major functions. The first one is to perform a 1D discrete ordinates calculation to obtain a value for the neutron flux. The second function is to use the calculated fluxes from its spectral calculation to collapse the input cross-section library either over a number of energy groups, a number of zones or both. In this last case the cross sections are calculated by

$$\bar{\mathbf{s}}_g = \frac{\sum_j^{IZM} N^j \sum_{g \in G} \mathbf{s}_g^j \int \mathbf{f}_g^j(\bar{\mathbf{r}}) d\bar{\mathbf{r}}}{N \sum_j^{IZM} \int \mathbf{f}_g^j(\bar{\mathbf{r}}) d\bar{\mathbf{r}}} \quad (3.13)$$

¹ This value was calculated by V. Berthou using Dancoff MC (November 2000)

In which

$$N = \frac{\sum_j^{IZM} V^j N_j}{\sum_j^{IZM} V^j} \quad (3.14)$$

G is the number of energy group that will be collapsed into one IZM is the number of regions we apply collapsing over, V is the volume and N_j the atomic density. It can be seen that the weighting is done to the flux for energy group collapsing as well as for cell weighting.

3.2.5 The BOLD-VENTURE code

BOLD-VENTURE solves the neutron diffusion equations. Next to solving these diffusion equations, this code can merge different cross section libraries. At this stage this is the only option we are interested in. BOLD VENTURE will be discussed in chapter 4, where it will be used to verify the results obtained by the FX2 code. BOLD VENTURE is used to merge ISOTXS libraries. For AMPX weighted libraries we use the WAX code to do the same job.

3.3 Model build-up

Creating a cross section library of an entire reactor involves several steps. One always starts with a master library containing information of many nuclides including the Bondarenko factors and resonance information. Here the XMAS 172 group master library was used. This library comes from the JEF 2.2 library. Before we can simulate the reactor we have to obtain the cross sections for a homogenised core. Homogenising the core involves two steps. The first step is homogenising the smallest element in the core, the TRISO coated particle. The second step towards a homogenised core is homogenising the pebble. With these cross sections of the homogenised core available the reactor could be simulated and any energy group collapsing applied. This has to be done in three steps. Two steps involve simulating the reactor in one direction. This is a limitation of the XSDNPMN code, it can only calculate in one direction at a time. The last step was to merge these two libraries. For an overview of the calculations see figure 3.2.

Step 1: Homogenising the TRISO coated particle

This step consists actually of three smaller steps. The first one is to take into account the resonance shielding in the unresolved part of the energy spectrum. This is done by the BONAMI code. The second step is to take into account the resonance shielding in the resolved part of the energy spectrum. This is done by the NITAWL II code. The actual homogenisation is done by XSDRNPM code. This code homogenises the TRISO coated particle and some carbon, from the pebble, surrounding it. With these steps done a library of a homogenised centre of the pebble was generated. These three codes were run by using the CSAS driver code, that is a code that does not do any calculations itself, but prepares the input files for the selected codes to run.

Step 2: Homogenising the core

Since the resonance shielding was taken into account in the previous step, only one code was used. XSRDNPM homogenised the pebble and the helium surrounding it. In this way the entire core, pebbles and helium, was homogenised.

Step 3&4: Simulating the reactor in axial and radial direction

For these two steps XSDRNPM was used with one difference compared to previous calculations. Instead of homogenising the cross sections over all zones in the model, the cross sections were only homogenised over each region.

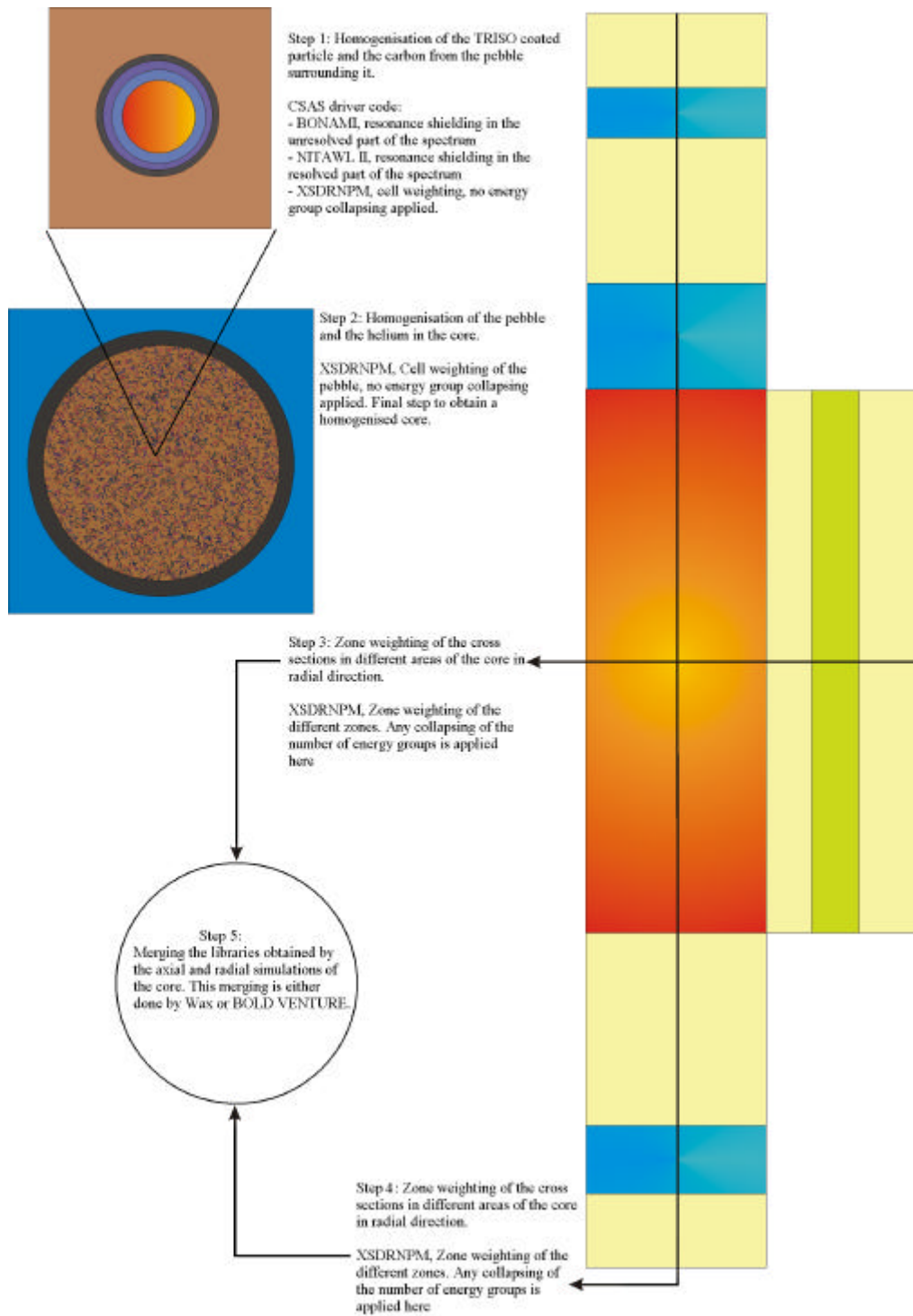


Figure 3.2; Overview of the calculation sequence

In this way we obtained a library with the cross sections of all nuclides in all zones. So for example carbon that was used in the core and the reflectors had separate cross sections for each region it was used in. One of the consequences of this zone weighting was that when a different reactor model with a different zone structure was used, the axial and radial simulation of the core had to be done again. When we applied collapsing of the energy groups this was also in these two steps.

In step 3 & 4 the same cross section library was used to start with. This library contained for example carbon that had been modified in the previous homogenisation steps. This was correct for the carbon in the core, but not for the carbon in the reflectors. This resulted in a slight overestimation of the $k_{\text{effective}}$ and therefore an extra step was done in the calculations, but only for the inter-code comparison. In this extra step, between steps 2 and 3, carbon, boron and helium were added to the library. These three nuclides were the only ones that appeared outside the core. Carbon and boron can be found in the reflectors and helium is the coolant. The cross sections of these three nuclides were not weighted or homogenised, only the resonance shielding was taken into account.

The fifth and final step was to merge the libraries of the core in radial and axial direction together to obtain one library for future simulations. Running all these code with their different input files and their different nuclide densities is time consuming. To avoid this, a PERL script is used that generates all the necessary input files, with their specific densities, and which calls the codes to run. Since some of these codes have been around for quite a long time and some input is still in fixed format, a PERL script makes the interface a lot more user-friendly.

3.4 The Reactor Model

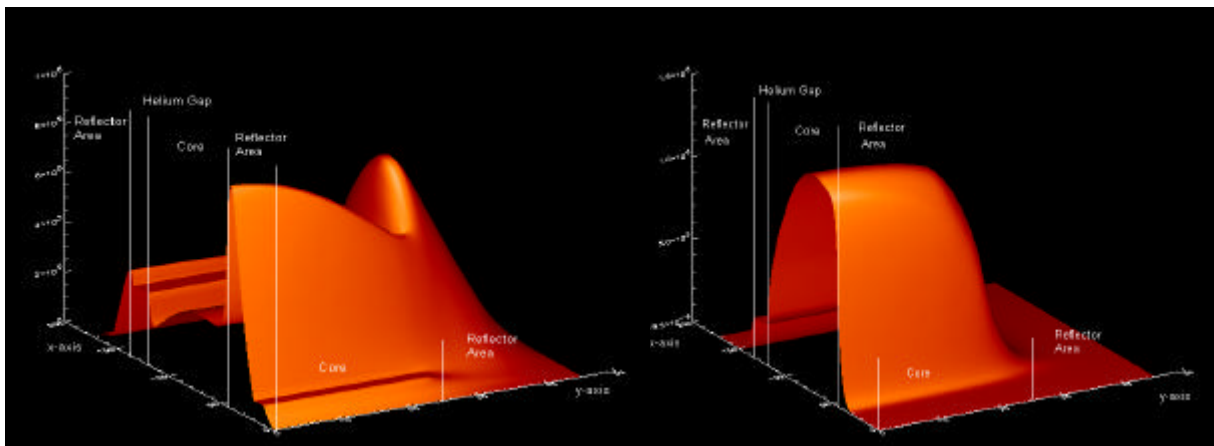


Figure 3.3; Thermal flux (< 6 eV) on the left side, fast flux (> 6 eV) on the right side of the type 3 model. The x-axis contains the mesh points in radial direction, the y-axis in axial direction. Both these pictures were taken on a 5 cm mesh in both directions.

Two geometry models of the ACACIA reactor will be discussed. The first is a very simple one build for speed. This model that we will call the type 1 model, can be divided into two sub models. One is the model of the axial direction and the other is the one in radial direction. The model simulates the top and bottom reflector as one. This implies that the densities and the cross sections are averaged over both zones. The same applies for the top and bottom outer reflectors. In radial direction the reflectors are simulated by separate zones. The core is simulated by one homogeneous zone. In figure 3.4 an overview of this model can be seen.

The second model is a more complex model, which takes into account some restrictions of the diffusion theory. Due to the approximations made in the diffusion theory it is difficult to calculate the flux in the border zones. So a more detailed model is necessary, especially where the flux gradient is large. The shape of the flux can be seen in figure 3.3. A large gradient can be seen for example in the reflector area, where the moderator, carbon, modifies the fast flux in thermal flux. In the core region next to the reflectors there is an enormous change in the thermal flux due to the increased fission by the reflected thermal neutrons. This model is expected to be more precise in the areas where the core and the reflectors meet. In axial direction multiple zones simulate both bottom and top reflectors. In radial direction the first inner reflector and the part of the core next to it are divided in multiple zones.

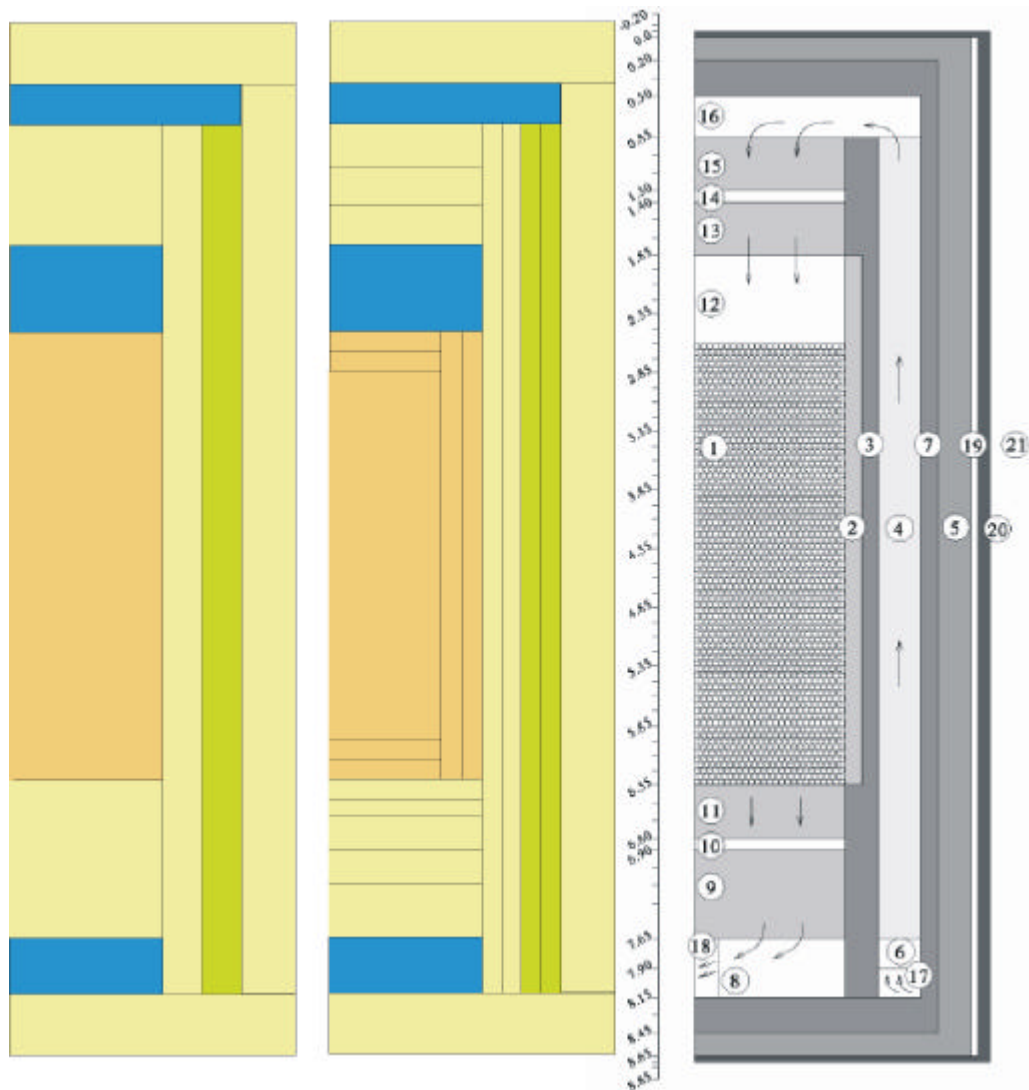


Figure 3.4; From left to right a schematic view of the type 1 model, the type 3 model and the ACACIA Reactor. Each box represents one zone.

4. FX2, a 2 dimensional diffusion code

4.1 Diffusion theory

FX2(R.A.Shober, T.A.Daly, & D.R.Ferguson 1978) is a two dimensional neutronics code that solves the multigroup neutron diffusion equation. The diffusion equation is an approximation of the neutron transport equation(J.E.Hoogenboom & H.van Dam 1998).

$$\frac{1}{v} \frac{d\mathbf{f}(r, E, \Omega, t)}{dt} = S(r, E, \Omega, t) + \int_0^\infty \int_{4p} \Sigma_s(r, E' \rightarrow E, \Omega' \rightarrow \Omega) \mathbf{f}(r, E', \Omega', t) dE' d\Omega' - \Omega \cdot \nabla \mathbf{f}(r, E, \Omega, t) - \Sigma_t(r, E) \mathbf{f}(r, E, \Omega, t) \quad (4.1)$$

Because it is an approximation the use is limited, but on the other hand it has a number of advantages because of its simplicity. The first approximation is to expand the angle dependency of the flux to Legendre polynomials(L.Massimo 1976).

$$\mathbf{f}(r, E, \Omega, t) = \frac{1}{4p} \{ \mathbf{f}(r, E, t) + 3\Omega \cdot \mathbf{J}(r, E, t) \} \quad (4.2)$$

By multiplying the neutron transport equation by Ω and then integrating over 4π , we obtain

$$\frac{1}{v} \frac{d\mathbf{J}(r, E, t)}{dt} = - \int \Omega [\Omega \cdot \nabla \mathbf{f}(r, E, \Omega, t)] d\Omega - \Sigma_t(r, E) \mathbf{J}(r, E, t) + \int \Omega S(r, E, \Omega, t) d\Omega + \iiint \Omega \Sigma_s(r, E' \rightarrow E, \Omega' \rightarrow \Omega) \mathbf{f}(r, E', \Omega', t) dE' d\Omega' d\Omega \quad (4.3)$$

The first term of the right part of the equation can be rewritten by using the first approximation (4.2) as

$$- \int \Omega [\Omega \cdot \nabla \mathbf{f}(r, E, \Omega, t)] d\Omega = - \frac{1}{3} \nabla \mathbf{f}(r, E, t) \quad (4.4)$$

The second approximation is to assume that the source S is isotropic, in which case the integral over S will be zero. The term, which describes the scattering of the neutrons, can be rewritten as

$$\iiint \Omega \Sigma_s(r, E' \rightarrow E) \mathbf{f}(r, E', \Omega', t) dE' d\Omega' d\Omega = \overline{m_0} \Sigma_s(r, E) \mathbf{J}(r, t) \quad (4.5)$$

Here $\overline{m_0}$ is the average cosine of the scattering angle. A third approximation is that the change in current in time is negligible compared to the other terms in (4.3). So with

$$\frac{1}{|\underline{J}|} \frac{\partial \underline{J}}{\partial t} \ll v \Sigma_t \quad (4.6)$$

We can write the current as a function of the flux

$$\underline{J}(\underline{r}, E, t) = -D(\underline{r}, E) \underline{\nabla} \mathbf{f}(\underline{r}, E, t) \quad (4.7)$$

with the diffusion constant

$$D(\underline{r}, E) = \frac{1}{3\Sigma_{tr}(\underline{r}, E)} \quad (4.8)$$

in which

$$\Sigma_{tr} = \Sigma_t - \overline{m}_0 \Sigma_s \quad (4.9)$$

Now we can generate the diffusion equation by integrating (4.1) over 4π and substituting (4.7) in this equation

$$\begin{aligned} \frac{1}{v} \frac{\partial \mathbf{f}(\underline{r}, E, t)}{\partial t} = \underline{\nabla} \cdot \mathbf{D}(\underline{r}, E) \underline{\nabla} \mathbf{f}(\underline{r}, E, t) - \Sigma_t(\underline{r}, E) \mathbf{f}(\underline{r}, E, t) + \\ S(\underline{r}, E, t) + \int_0^{\infty} \Sigma_s(\underline{r}, E' \rightarrow E) \mathbf{f}(\underline{r}, E', t) dE' \end{aligned} \quad (4.10)$$

The approximations made in previous derivation have some restrictions on the range of applications this theory can be used for. Because of the expansion of the angle dependency the diffusion equation is not valid in regions where the flux has a large gradient or in very strong absorbers. The change in flux may not vary much over a distance of

$$l_t = \frac{1}{\Sigma_t} \quad (4.11)$$

This means that using the diffusion theory near very strong neutron sources or material boundaries can result in discrepancies compared with transport codes. On the other hand the simplicity and the mathematical methods to solve these equations make it a very useful and fast computer code, that results in very good approximations for most reactors.

4.2 The FX2 code

The FX2 code is a two dimensional diffusion code. It was written to simulate Liquid Metal Fast Breeder Reactors (LMFBR). This is a different type of reactor than the one considered here. The programmers of this computer code, however, ensure that it can be used for other types as well. More important, the code has some very useful properties for transient analysis. One of the most useful properties for our case is the fact that FX2 interpolates in cross-section libraries at different temperatures. This means that, if we supply cross section libraries spanning the typical range of temperatures encountered by a transient, we can simulate either dynamically or steady state using the same code and the same libraries. Because one of the goals of this research was to replace the point kinetic model of ACM by a two dimensional diffusion code, this last property becomes very valuable. Next to this the FX2 code can simulate RZ-geometries. This configuration is ideal for simulating a cylinder shaped reactor in 2 dimensions.

The goal is to obtain a model, as simple as possible, which will show good comparison with more detailed simulations as well as with other codes. The $k_{\text{effective}}$ will be calculated for different energy groups and different models. The $k_{\text{effective}}$ is an eigenvalue of the diffusion code when forced in steady state. This eigenvalue is also known by the name of multiplication factor. This name explains the physical meaning of this parameter. It is the ratio of the number of neutrons on two consecutive generations.

FX2 will be compared with BOLD VENTURE, another diffusion code, and KENO Va, a neutron transport code that uses the Monte Carlo method. Because we have to find out if FX2 generates good results and to find out how detailed the model should be, both in number of energy groups and geometrically, we are not yet interested in dynamic simulations. In the forced steady state all time derivatives in previous paragraph become zero and the $k_{\text{effective}}$ is added. So (4.10) can be rewritten as

$$-\nabla \cdot D_g(r) \nabla \mathbf{f}_g(r) + \sum_{g'=1}^G \Sigma_{gg'}(r) \mathbf{f}_{g'}(r) = \frac{1}{k_{effective}} \mathbf{c}_g \sum_{g'=1}^G \mathbf{n} \Sigma_{f_{g'}} \mathbf{f}_{g'}(r) + Q_g(r) \quad (4.12)$$

This can be rewritten as

$$\int_k \underline{J}_g(r) \cdot dS + \sum_{g'=1}^G V_k \Sigma_{gg',k} \bar{\mathbf{f}}_{g',k} = \frac{1}{k_{effective}} \mathbf{c}_g \sum_{g'=1}^G V_k \mathbf{n} \Sigma_{f_{g',k}} \bar{\mathbf{f}}_{g',k} + \bar{Q}_g(r) \quad (4.13)$$

In this equation k refers to the mesh cell k and

$$\bar{\mathbf{f}}_{g,k} = \frac{1}{V_k} \int_k \mathbf{f}_g(r) dV \quad (4.14)$$

$$\bar{Q}_{g,k} = \frac{1}{V_k} \int_k Q_g(r) dV \quad (4.15)$$

$$V_k = \int_k dV \quad (4.16)$$

In our case we do not have any external source, so $Q = 0$. This leads to the following equation that is solved by FX2.

$$[L_g][\bar{\mathbf{f}}_g] + [\Sigma_{r_g}][\bar{\mathbf{f}}_g] - \sum_{\substack{g'=1 \\ g' \neq g}}^G [\Sigma_{s_{gg'}}][\bar{\mathbf{f}}_{g'}] = \frac{1}{k_{effective}} \mathbf{c}_g \sum_{g'=1}^G [\mathbf{n} \Sigma_{f_{g'}}]^T [\bar{\mathbf{f}}_{g'}] \quad (4.17)$$

This is a homogeneous matrix eigenvalue problem in which

K is the total number of mesh cells.

$[\bar{\mathbf{f}}_g]$ is a column vector of length K consisting of the neutron fluxes at every mesh cell for group g .

$[\Sigma_{r_g}]$ is a diagonal $K \times K$ matrix consisting of the macroscopic removal cross sections at every mesh cell for group g times the volume of the mesh cell.

$[\mathbf{n} \Sigma_{f_{g'}}]$ is a column vector of length K consisting of the average number of neutron per fission times the macroscopic fission cross section at every mesh cell for group g times the volume of the mesh cell.

$[\Sigma_{s_{gg'}}]$ is a diagonal $K \times K$ matrix consisting of the macroscopic scattering cross sections at every mesh cell for scattering from group g' to group g times the volume of the mesh cell.

$[L_g]$ is a banded $K \times K$ matrix consisting of the finite difference coefficients at all mesh cells for group g .

Solving this equation is an iterative process in which FX2 first calculates the right hand side of the equation for all energy groups g . Next it solves the matrix equation after which it will make a new estimate of the eigenvalue $k_{effective}$. FX2 has a wide variety of options for simulating many different situations. It is however a code written to simulate fast breeder reactors and one of the assumptions made in this code was that it does not take any up scattering into account. Up

scattering is the effect that a neutron gains energy when it collides with a nuclide. This happens only when the energy of the neutron is below 3.3 eV.

4.3 Different calculations

In 4.1 the diffusion equations is derived. In this derivation some approximation were made that have some consequences for the type of reactor we want to simulate here. Equation 4.11 states that the flux may not vary much over a distance of $1/\Sigma_t$. But in the cavity the density of the helium is low compared to other densities. Helium has also very small cross sections. These two properties of the cavity make that requirement stated in equation 4.11 cannot be fulfilled. Gerwin and Scherer(H.Gerwin & W.Scherer 1987)came up with a method to limit the diffusion constant in the cavity above the pebble bed. They derived an expression for the value of the diffusion constant in the cavity as a function of height and radius of the cavity.

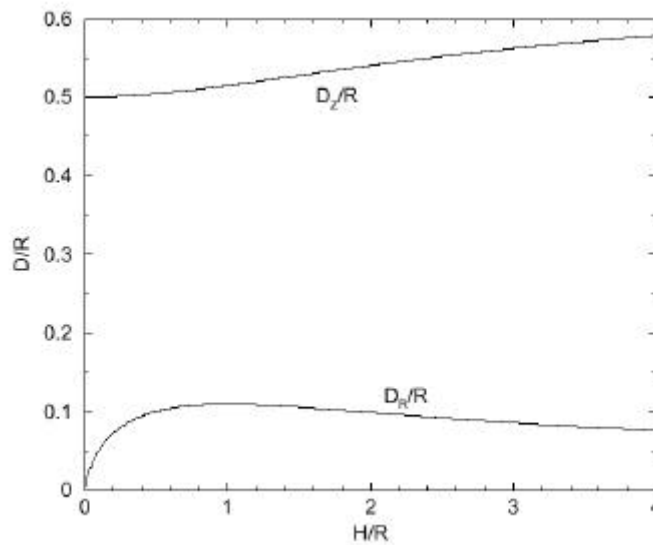


Figure 4.1; Normalised diffusion constants as a function of the cavity height (H) and radius(R) according to Gerwin and Scherer(H.Gerwin & W.Scherer 1987).

Gerwin and Scherer make use of an anisotropic diffusion constants in two directions. In the ISOTXS library we could change only one microscopic transport cross section. Since the diffusion constant is a measurement for the distance as the crow flies between the origin of the neutron and the point where it is absorbed, we squared both the radial and the axial diffusion constant and added them together. In this way we got the square of the new diffusion constant we were going to implement.

$$D' = \sqrt{D_r^2 + D_z^2} = \frac{1}{3N\sigma_{tr}} \quad (4.18)$$

A second weakness of the diffusion theory is near borders of different regions. In order to obtain better results and reduce the influence of this weakness of the diffusion theory a heterogeneous mesh was applied. In border regions between core and reflectors the mesh was narrowed to 2.5 cm and even 1 cm. On the other hand in the outside border regions the mesh was enlarged to 10 cm. In appendix A an overview of the meshes is presented.

A third type of modification to the models was that we modified the buckling factor in XSDRNPM. The buckling factor represents the leakage of neutrons in transverse direction.

XSDRNPM is a one dimensional code and assumes that in the other direction(s) the model stretches to infinity. In this way there is no leakage in transverse direction. The buckling factor was set to such a value that a $k_{\text{effective}}$ of 1, or a critical reactor, was obtained. When this was reached, the cross sections will be collapsed using a critical flux. And in this way an appropriate set of cross sections for a critical reactor is generated.

4.4 BOLD-VENTURE and Keno Va

Results from one code can seem very likely or show good comparison with reality. In our case the comparison with reality is not possible, so a code can only seem very likely. This however can be improved when we compare the results with those of other codes and they both agree reasonably well. This is called inter-code comparison and is widely used throughout the scientific community. FX2 is a diffusion code that has some limitations. One of the codes we compared with is a Monte Carlo transport code. Keno Va(L.M.Petrie & N.F.Landers 1993) was selected for this job. This is a 3 dimensional Monte Carlo transport code. It is a time consuming code that no approximations in the calculation algorithm. It places a user defined number of neutrons in the reactor and tracks their path until they are absorbed or they leak out of the reactor. In the case of fission their location is used as a seed for the next generation of neutrons. In our case we started every generation with 10000 neutrons and tracked them for 500 generations.

FX2 was also compared with BOLD VENTURE. This is also a diffusion code but this code cannot calculate transients and it cannot interpolate between different input libraries made at different temperatures. This is however not a problem for static calculations at one temperature. BOLD VENTURE does however take up scattering into account.

5. The Aspen Custom Modeller program

The simulation code written by Kikstra (J.F.Kikstra 2001) can actually be divided into two parts. These are the highly detailed energy conversion system, with approximately 33.000 variables, and the model of the nuclear reactor. A detailed description of this model can be found in the thesis of Kikstra. The first part is very detailed and suits our purpose well. The second part can have some improvements, which will be discussed below.

The reactor model can be divided into three parts, the solid elements, these describe the pebble bed and the reflector elements, the flow elements and the neutronics. All these elements are described by different sub-routines. Next to these three types of elements there is the main model that connects all elements with each other and with the input and output of the energy conversion system.

5.1 Aspen Custom Modeller

Aspen Custom Modeller is a solver. This means that ACM solves all the equations in the model by placing them in one matrix and solves this matrix. This is very different from a model written in a programming language like Fortran or C. In the latter one can write $n=n+1$ in a loop for example. In ACM this would lead to a singularity because this equation cannot be solved. ACM generates a matrix containing all the equations and solves in this way the set of equations.

The total number of equations should be enough to solve the number of unknown parameters. If this is not the case and you do not have enough equations your model is 'under defined' or if there are too many 'over defined'. This exposes one of the weak points of ACM, there is almost no constructive help for debugging. The program simply states that there are not enough or too many degrees of freedom, equations, and it refuses to run. The option, built in ACM to help find where the equations are lacking or redundant, gives in the best case only a direction where to look for the errors. Because of this limitation, progress could be made only very slowly. Each step, that we wanted to take as small as possible, had to be checked for steady state simulations as well as dynamic simulations and of course for the correct number of degrees of freedom.

A second point that gave troubles was in the listing of the code. It was not clear where what parameters were used and where they were being calculated. Because this model was ill commented this turned out to be a problem.

It might look that ACM only has disadvantages, but that is not true. If the model in ACM is written for calculating the outlet temperature as a function of a change in input pressure, we can easily switch to the outlet pressure as a function of inlet mass flow. You can keep on changing parameters from fixed to free and visa versa as long as the total number of equations and unknown parameters remains balanced. This is the main advantage of ACM: it is very easy to switch the parameters you want to simulate the behaviour of.

In figure 5.1 an overview can be seen of the original model made by Kikstra. The blocks represent the different sub routines, each representing a part of the reactor. The arrows between different sub routines represent connections for exchanging parameters. These connections can also represent physical connections between different routines, for example they can represent heat exchange or the flow of the helium. In figure 5.1 one can also see two routines that are

separated of the other routines, the point kinetics and the geometry routines. These two routines are actually connected to all the routines but the arrows were left out for reasons of clarity. In this last case the connections do not represent a physical connection.

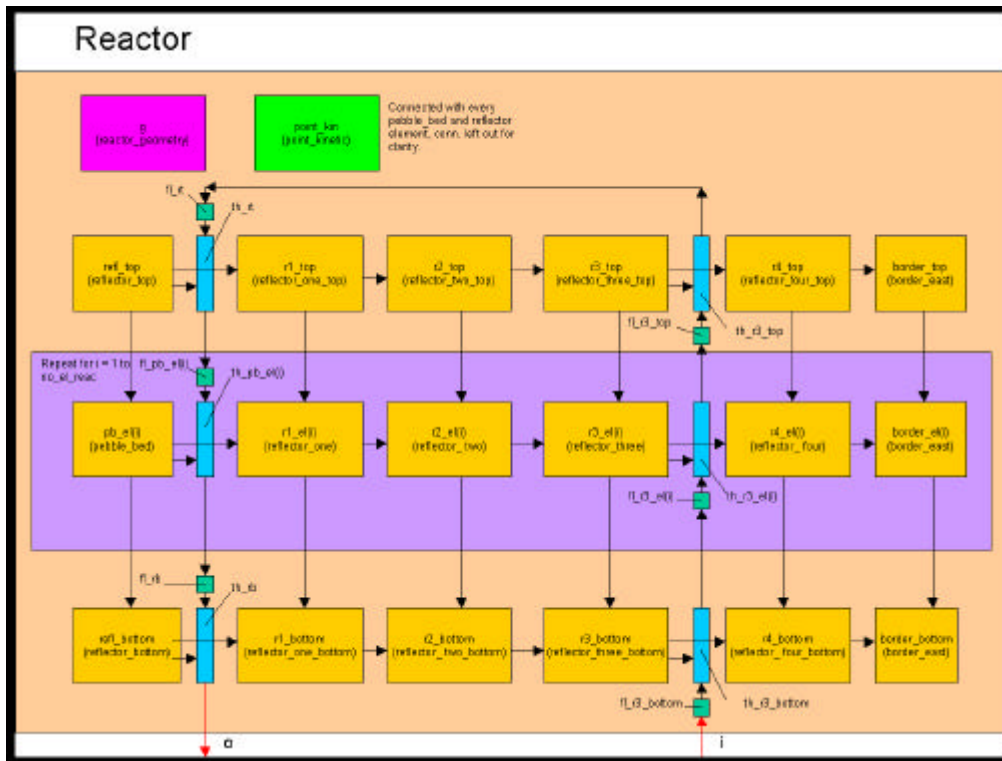


Figure 5.1; Schematic overview of the model of the reactor created by Kikstra. Arrows indicate a connection for heat transfer between the different routines.

5.2 The helium flow

In the ACM model the helium flow is divided into flow nodes and thermal nodes. These nodes overlap each other as can be seen in figure 5.2.

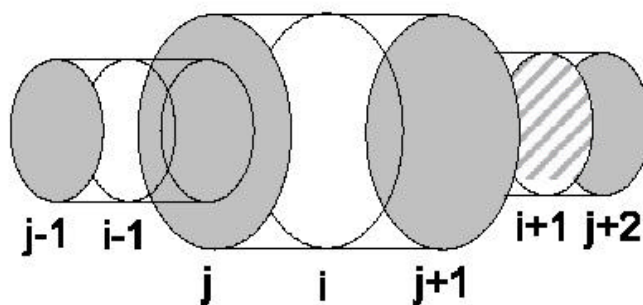


Figure 5.2; Schematic view of overlapping flow and thermal nodes.

A flow node, consisting of two half thermal nodes, extends from $i-1$ to i . A thermal node, consisting of two half flow nodes extends from j to $j+1$. The flow node has the function of determining the mass flow. The thermal node will determine the new material properties due to a change in internal energy because of the heat transfer from the element the helium is flowing through. These properties include internal energy, density, pressure etc. In the flow node the mass flow is calculated from the momentum balance

$$\begin{aligned} \frac{d(\Delta x_j \mathbf{f}_j)}{dt} = & \mathbf{f}_{i-1} v_{i-1} - \mathbf{f}_i v_i + A_{i-1} p_{i-1} - A_i p_i - (A_{i-1} - A_i) p_j \\ & + F_{friction_j_{i-1}} - F_{friction_j_i} + \frac{F_{rot_{i-1}} - F_{rot_i}}{2} + \frac{F_{grav_{i-1}} - F_{grav_i}}{2} + F_{comp_j} \end{aligned} \quad (5.1)$$

ϕ stands for the mass flow, v for the velocity, p for the pressure and A for the area the flow is passing through. The mass flow is taken on the borders $i-1$ and i of the flow node j . The energy balance for the thermal element can be written as

$$V_i \frac{d(\mathbf{r}_i E_i)}{dt} = \mathbf{f}_j (H + K + \Phi)_j - \mathbf{f}_{j+1} (H + K + \Phi)_{j+1} + Q_i + W_i \quad (5.2)$$

In which E_i is the total specific energy, H the specific enthalpy, K the specific kinetic energy and Φ the specific potential energy. The balance is taken between borders j and $j+1$ of element i . The thermal node needs one more equation: the mass balance

$$V_i \frac{d\mathbf{r}_i}{dt} = \mathbf{j}_j - \mathbf{j}_{j+1} \quad (5.3)$$

By solving the mass balance and the energy balance, helium thermal nodes calculate the specific energies and density. With these parameters a range of other material properties of the helium flow as well as pressure and temperature of the helium in node i can be calculated. The model has the ability to take potential forces into account. But all simulations were done by neglecting this part of the energy balance. The gravitation forces on the other hand, were taken into account in the momentum balance.

The friction force is calculated by

$$F_{friction_j_i} = - \frac{e_{vi} |\mathbf{f}_j| \mathbf{f}_j}{4A_i \mathbf{r}_i} \quad (5.4)$$

In which e_v is the friction loss factor. It is a function of

$$e_{vi} = \frac{4f_i \Delta x_i}{D_{ei}} \quad (5.5)$$

In this function f_i is the Fanning friction factor, a function of the Reynolds number and the geometry of the flowpath.

5.3 Solid elements

The solid elements are a bit more straightforward. These types of element are used to describe the pebble bed elements and the reflector elements. This is a two dimensional model, so the four sides will be referred to as north, east south and west or simply n, e, s and w. For each solid element an energy balance has to be solved.

$$V \frac{d(\mathbf{r}H)}{dt} = \mathbf{a}_{n_to_c} S_{n_to_c} (T_n - T_c) + \mathbf{a}_{e_to_c} S_{e_to_c} (T_e - T_c) + \mathbf{a}_{w_to_c} S_{w_to_c} (T_w - T_c) + \mathbf{a}_{s_to_c} S_{s_to_c} (T_s - T_c) + Q_{fluid} + Q_{production} \quad (5.6)$$

In which

$$\mathbf{a}_{n_to_c} = \frac{1}{\frac{\Delta x_n}{I_n} + \frac{\Delta x_c}{I_c}} \quad (5.7)$$

In these equations $\alpha_{n_to_c}$ is the heat transfer coefficient for transfer from the centre of this element to the centre of the north element. The thermal conductivity, λ , is determined in all elements, Δx_c is the distance from the centre of an element to the side we are calculating the heat transfer for and Δx_n the distance from the centre of the north element to its south side. $S_{n_to_c}$ is the area for heat conduction from the elements we are interested in to its north side. The thermal conductivity is a function of temperature and depends on the type of material and the geometry. For the thermal conductivity empirical relations were obtained for each different type of geometry and material as a function of the temperature (VSOP 1994; Yan 1990). This approximation includes heat transfer by radiation.

The difference between the reflector and the pebble bed elements can be found in the last term of (5.6). The reflector elements do not produce any heat, so $Q_{production}$ is equal to zero. Pebble bed elements do not have any heat transfer to the west side due to symmetry. There are also some differences between the reflector elements. Not all reflector elements contain helium flowing through pipes, so in most reflector elements Q_{fluid} is also equal to zero. When a helium flow is connected to the solid element, this routine also calculates the friction loss factor as a function of diameter, flow length and geometry.

The heat capacity of the graphite, an important property of the HTR, is not independent of density or temperature. It is calculated by an empirical relation (Yan 1990) as a function of temperature.

$$(\mathbf{r}C_p)_{PB} = (1 - \mathbf{x}) \sum_{i=0}^3 C_i (T_{PB} - 273.15)^i \quad (5.8)$$

In this equation ρ stands for the density, ξ for the porosity and C_i for 4 empirical constants determined by Yan.

5.4 Neutronics

The original ACM model uses a rather simple model for the neutronics. A fixed relation between the temperature of the different elements and their reactivity contribution calculates the reactivity change in the reactor. Each solid element in the reactor sends its temperature to the point kinetic model. This model compares this temperature to a reference temperature, that is the temperature of this element at nominal operating conditions. The change in temperature will result in a change in reactivity according to an empirical relation determined by Verkerk.(E.C.Verkerk 2000)

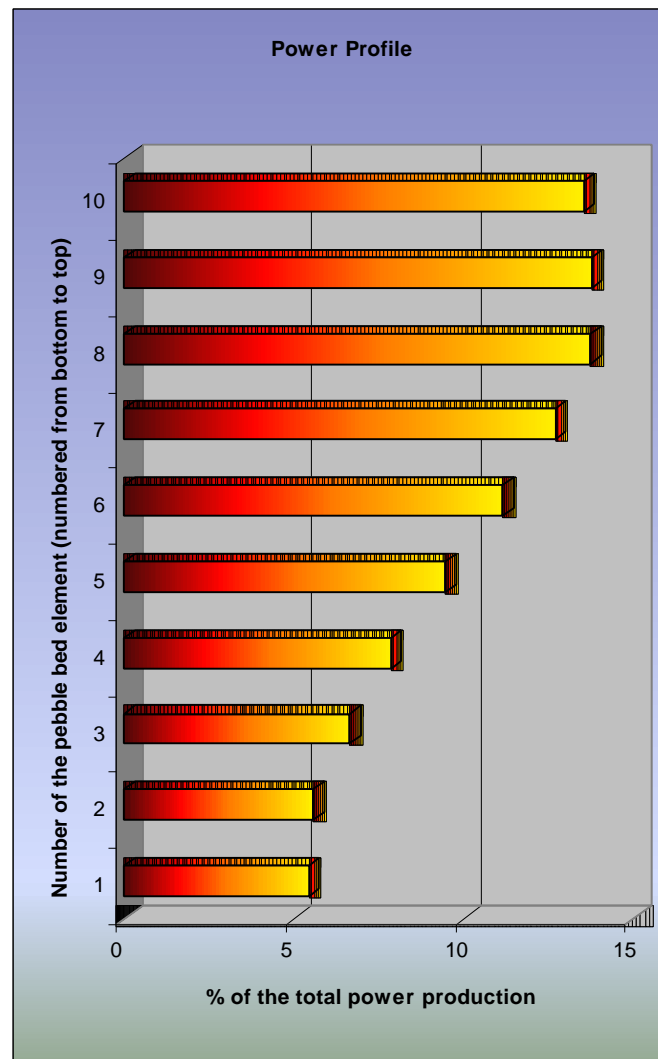


Figure 5.3; The fixed power profile

This change in reactivity results in a new total power production. This power production is then in its turn redistributed to the pebble bed elements according to a fixed power profile, see figure 5.3. These assumptions lead to a simple neutronics model, which works well for nominal operation conditions. One of the reasons for the differences in power production between the slabs is the burnup in fuel pebbles. The power profile is that of a core after 1055 days of operation. Since this reactor uses a pue-à-pue fuelling strategy, the top layers consist of relatively new pebbles. Therefore it isn't surprising that the top layers produce most of the power.

The power production in each axial slab is calculated by

$$Q_{generated} = P_{normalised} Q_{nominal} \quad (5.9)$$

In which $P_{normalised}$ is the dimensionless normalised thermal power produced in the reactor according to the point kinetics model and $Q_{nominal}$ the nominal power production at normal operating conditions. The reactor dynamics are modelled with the point kinetic approximation using six delayed neutron precursor groups and 23 decay heat groups. But from this 23 decay heat groups the last 10 were merged into one by

$$\frac{\mathbf{b}_{eff}}{\mathbf{l}_{eff}} = \sum_{j=14..23} \frac{\mathbf{b}_j}{\mathbf{l}_j} \quad (5.10)$$

β_{eff} is the sum of all β 's in the summation. The prompt power production is calculated in a normalised form by

$$\frac{dP_{pr}}{dt} = \frac{\mathbf{r} - \mathbf{b}}{\Lambda} P_{pr} + \sum_{i=1}^6 \mathbf{l}_i C_i \quad (5.11)$$

The concentration of precursor group i in normalised form is

$$\frac{dC_i}{dt} = \frac{\mathbf{b}_i}{\Lambda} P_{pr} - \mathbf{l}_i C_i, i = 1..6 \quad (5.12)$$

And finally the heat of decay group j

$$\frac{dP_j}{dt} = \frac{\mathbf{b}_j}{\Lambda} P_{pr} - \mathbf{l}_j P_j, j = 1..14 \quad (5.13)$$

With these equations we can calculate the normalised power production by

$$P_{norm} = \frac{P_{pr} + \sum_{j=1}^{14} \mathbf{l}_j P_j}{1 + \sum_{j=1}^{14} \mathbf{b}_j} \quad (5.14)$$

In the calculation for the reactivity the xenon poisoning and external factors are taken into account by

$$\mathbf{r} = \mathbf{r}_{temperature} + \mathbf{r}_{Xenon} + \mathbf{r}_{external} \quad (5.15)$$

The $\rho_{temperature}$ is the sum of all reactivity contributions of all the elements in the reactor. This contribution is a function of the nominal contribution and a change of the nominal temperature calculated by Verkerk.

$$\mathbf{r}_{temperature} = \sum_{k=1}^{N_{elements}} \mathbf{r}_k \Delta T_k \quad (5.16)$$

With ΔT_k the difference between the actual and the nominal temperature. The summation is done over all solid elements except for the border elements. The xenon concentration is an important factor in neutronics. This is due to its extremely high neutron absorption cross section. The reactivity contribution of xenon is calculated by

$$\mathbf{r}_{Xe} = \mathbf{r}_{Xe,st,st} (1 - C_{Xe}) \quad (5.17)$$

With $\rho_{Xe,st,st}$ the reactivity contribution of xenon in a steady state situation. C_{Xe} is the normalised concentration of xenon to a steady state situation in the reactor. So one can see that with increasing xenon concentration the reactivity drops and becomes negative. Xenon is created in a nuclear reactor in two ways. It is created directly by the fission process and it is a decay product of iodine, which in its turn is a product of the fission process. This makes the xenon concentration a function of power, this specifies the amount of xenon directly produced and destroyed by

neutron absorption, and a function of the iodine concentration, responsible for the production of xenon by β -decay. So the time dependent iodine concentration can be calculated by

$$\frac{dC_I}{dt} = g_I \Sigma_f f - I_I C_I \quad (5.18)$$

And the xenon concentration can be calculated by

$$\frac{dC_{Xe}}{dt} = g_{Xe} \Sigma_f f + I_I C_I - s_{a,Xe} C_{Xe} f - I_{Xe} C_{Xe} \quad (5.19)$$

In these equations the γ_I and the γ_{Xe} are the fractions of fissions that result in the creation of respectively iodine and xenon. In this model this equation looks a bit different since it uses normalised power production instead of flux to determine the amount of xenon and iodine produced. The decay constants of iodine and xenon are respectively 6.6 and 9.2 hours. This implies that the xenon poisoning is dominant in the hours-domain.

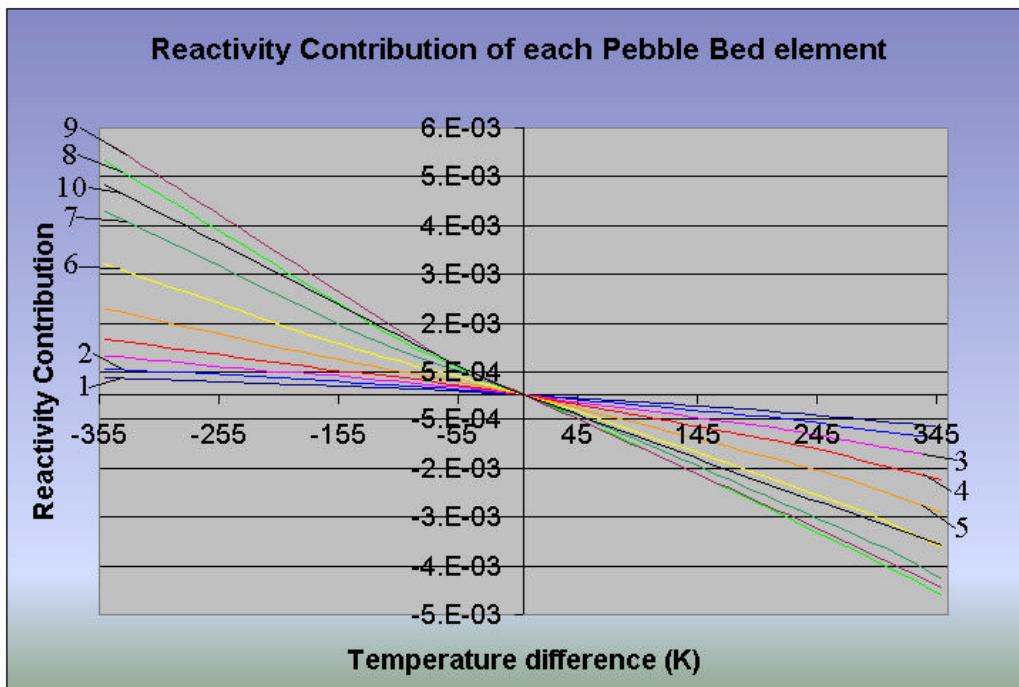


Figure 5.4; Reactivity contributions of the different pebble bed elements. The numbers represent the appropriate pebble bed elements.

Figure 5.4 shows the reactivity contributions calculated by Verkerk and form the basis for the ACM neutronics model. The x-axis shows the difference in temperature from the operating temperature. One can spot some properties of this model. The most important is the negative temperature coefficient. If the temperature difference is negative the reactivity contribution of all the elements is positive. The point kinetic model calculates the total power production in the reactor as a function of reactivity. This new total power production is then redistributed to the pebble bed elements according to the fixed power profile, figure 5.3. Since some pebble bed elements calculate more power than others, it is logical that a change in temperature in those elements has a larger contribution to the total reactivity than the elements, which produce less power. So similarities can be seen between figure 5.3 and 5.4. The elements that produce more power have the larger contributions to the reactivity at the same temperature difference.

5.5 Modifications

As mentioned before, the energy conversion system has been modelled very precisely. The reactor model itself could have some improvements. First of all the point kinetics model describes the behaviour well at operating conditions, but what if we want to simulate loss of flow accidents and loss of cooling accidents? A point of concern is the fixed power profile. This profile is assumed to be the same under all circumstances. Will this be valid during a loss of flow accident or loss of cooling accidents when the temperature changes more than 100 degrees?

The department of Reactor Physics has several neutronics codes at its disposal. In chapter four we described the two dimensional diffusion code, FX2. Combining this code and the ACM model would make the model more versatile. In order to do so, the ACM model should be modified to a two dimensional flow. A second modification is to insert a helium cavity between the top of the pebble bed and the top reflector. The last modification we will discuss is the model in which we corrected an error made by Kikstra. All these will be discussed in the next paragraphs, while results will be presented in chapter 6.

5.5.1 Two dimensional flow

The most radical modification was to enable two-dimensional flow in the ACM model. In order to achieve this we first tried to obtain a parallel vertical flow. This was the first step before we could try to model transverse flow. The modifications made to the model can be seen in figure 5.5. First the two plenums were modelled in the reactor. These plenums were necessary because the flow had to be redistributed in multiple parallel flows. The plenums were placed above the top reflector and below the bottom reflector. This corresponds to a helium cavity in the original model of the ACACIA reactor. Previously, these volumes were included in the volume of the helium space in the top reflector and the bottom reflector. This was obtained relatively easy.

The point kinetics model works with a fixed power profile for a fixed number of axial slices. In order to simplify the model and to be able to work with a parallel flow without having to calculate a new normalised power production, the point kinetics model was replaced by a model, that simply returned to each element that they were working at operation conditions. This means that no reactivity contributions of any element at any temperature were taken into account.

As seen in figure 5.5 the pebble bed ring elements, as well as reflector top ring and reflector bottom ring elements, differ from the original or centre elements by one connection. The ring elements conduct heat with both their eastern and their western neighbours.

The main modification, the parallel flow, turned out to be a time consuming action. The first problem to be encountered was to get the correct number of degrees of freedom. Finally, after several months of debugging and a third blank start, we obtained a model that had the required amount of degrees of freedom. Now phase two of the parallelisation started, obtaining a converging set of starting values for all parameters. This proved to be an even more challenging task. Several more simplifications were made to the model to make it easier to converge. The temperature dependent heat capacity, viscosity etc. were replaced by a fixed value. The Fanning friction factor was replaced by a fixed value. The temperature of all pebble bed elements was fixed and even the flow in the ring elements was fixed. But even with all these simplifications the model proved to be too complex to converge. After months of work, no results could be

obtained from the modifications, because convergence could not be obtained from the simulation with the parallel flow. We did obtain a working and converging model of a reactor with two helium gaps modelled in the simulation. Results of this modification will be shown in chapter 6.

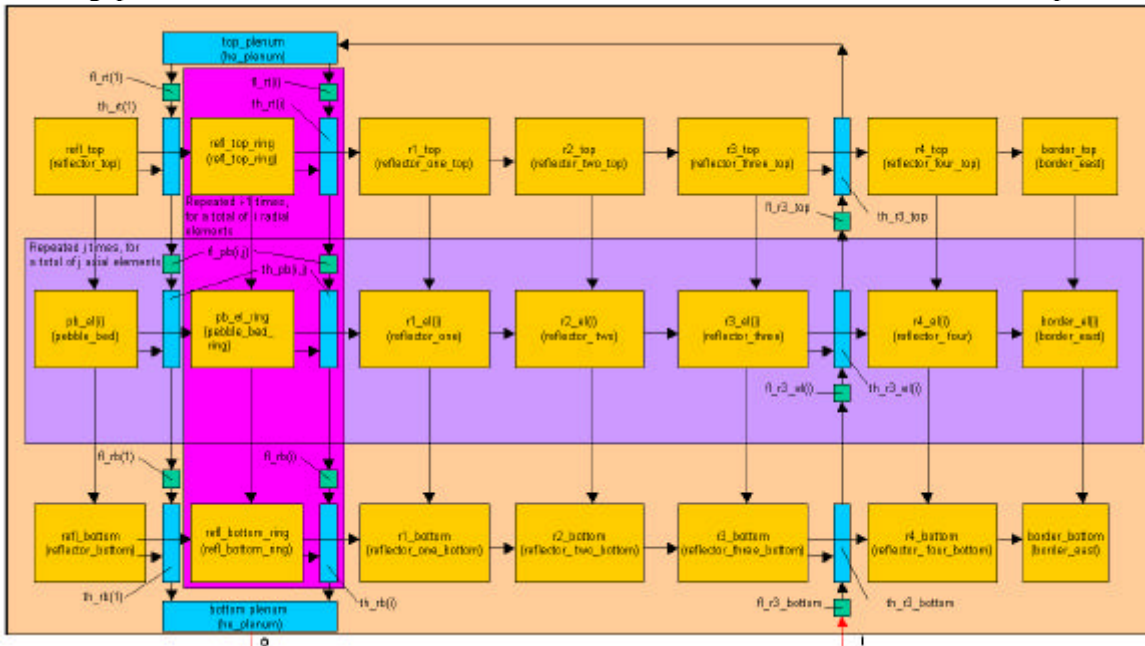


Figure 5.5; Schematic overview of the ACM model with parallel flow in the core

5.5.2 The helium cavity

Another modification was to include the helium cavity in the model. HTR reactors are not completely filled with pebbles, above the pebble bed the open space is filled with helium as described in chapter 2. Including this in the model is more difficult than one might imagine at first sight. The main difficulty is the fixed reactivity contribution of all elements in the reactor. As described in paragraph 5.4 the temperature of each element is a part of the function that calculates the reactivity. So an approximation was made. The reactivity contribution of the newly added helium cavity and the reflectors surrounding it was ignored. The helium cavity could not be modelled by using any of the existing models in ACM. We therefore had to modify one of the other models. A helium thermal node was modified to include heat transfer to the north and to the south.

To simulate any heat transfer between the top pebble bed element and the helium cavity, we used a fixed heat transmission coefficient. This was a fixed small value, which implies that the heat that is transmitted to the north will enter the pebble bed again because of the forced convection. For the heat transfer to the first reflector we used the same routine as the third reflector, which simulates a reflector with a helium flow passing through it.

5.5.3 Friction correction

One more modification and the most simple one was to correct an error in the original model. In paragraph 5.2 we discussed the friction factor. This factor is a function of the geometry of the flow path and the Reynolds number. In the original model the same friction factor was used for the flow through the pipes as well as for the flow through the pebbles. Both flow routines used the friction factor for flow through the pebble bed. This resulted in a too high friction for the flow through the reactor, approximately 80 times too high. The friction has a large influence on the pressure drop over the reactor and on the speed with which the reactor reacts to changes on the inlet flow.

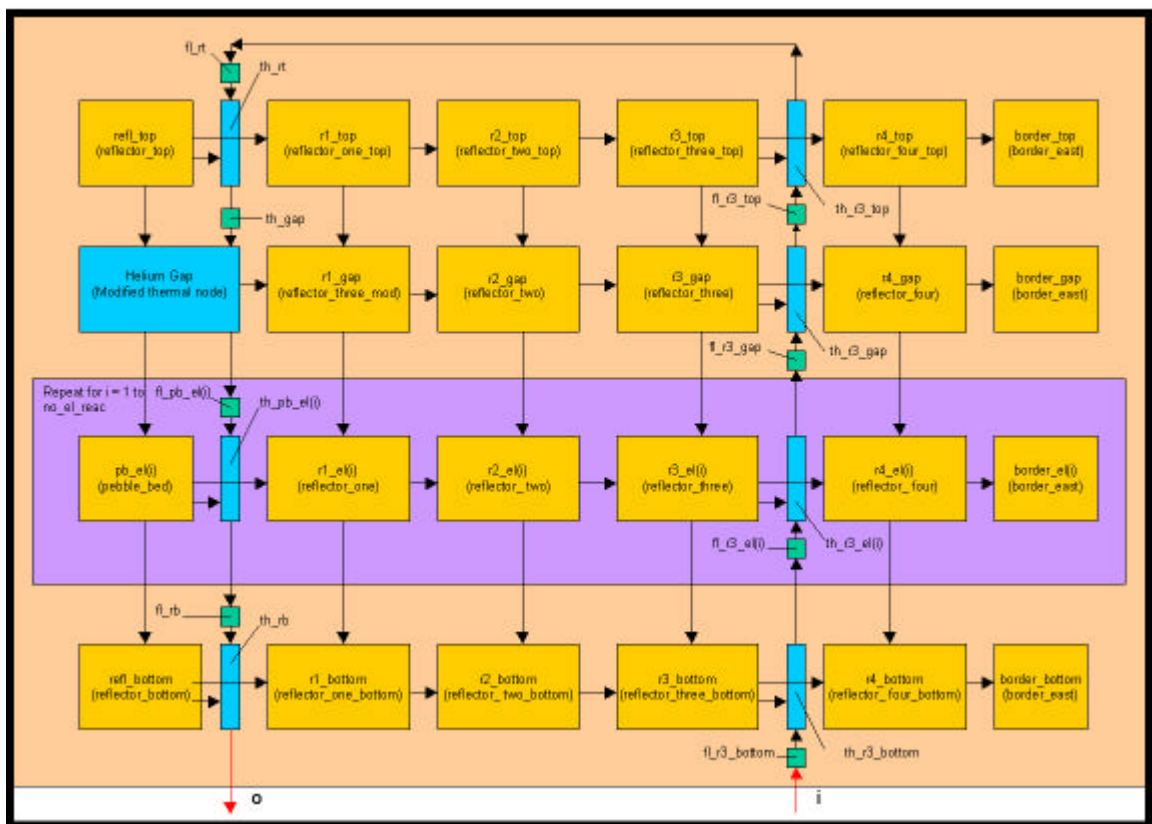


Figure 5.6; Schematic overview of the ACM model of the reactor with a helium cavity modelled

6. Results

6.1 FX2 results

In paragraph 4.2 we mentioned that we would investigate the influence of the number of energy groups on the $k_{\text{effective}}$. Therefore we started by calculating the $k_{\text{effective}}$ for a large variety of energy groups for a simple model. For a total of 21 different number of energy group structures, the $k_{\text{effective}}$ was calculated².

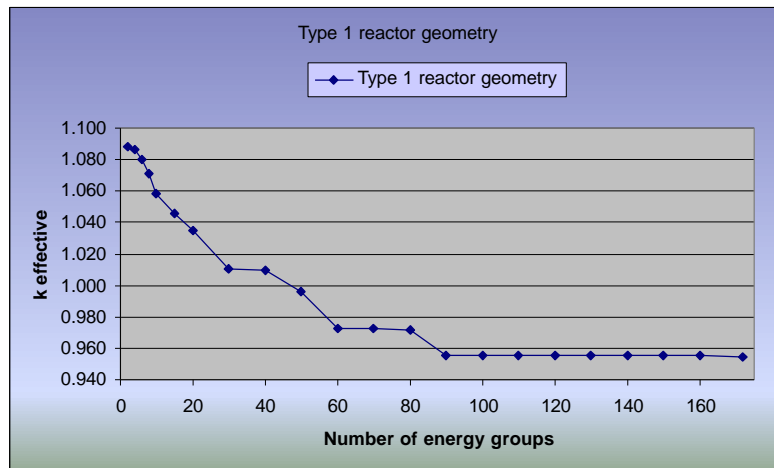


Figure 6.1; The influence of the number of energy groups on the $k_{\text{effective}}$ calculated by FX2

A strong energy group dependency can be seen in figure 6.1. The intention of these calculations is to find an energy group structure with as few energy groups as possible that produces the same results as a very detailed simulation. From this calculation we selected a number of energy group structures to continue with. All calculations have been done with 2,4,8,15,30, 70 and 172 groups. The energy boundaries for the different calculations can be seen in table 6.1. For 15 and more energy groups the energy boundaries were selected in a straightforward way. Simply divide the 172 energy groups by the number of energy groups wanted in the collapsed library. Any rest is divided over the fast range of the energy spectrum.

Unfortunately problems with the selected energy group structures were immediately encountered. When the 172-group calculation was done for the type 3 reactor model it turned out to be a problem for FX2. With a total of 25 different zones and 104 different nuclides the amount of memory needed by FX2 was too large to handle for this program. Therefore the 172-group calculation was dropped from the list. With the energy group structures selected the two reactor models could be compared.

² Some of these calculations were done by M. Flaska (June 2001)

Table 6.1; Upper energy boundaries for different number of energy groups.

Number of Energy groups	Upper boundaries (eV)
2	1, 1.96e7
4	1, 1e4, 1e6, 1.96e7
8	0.1, 1, 10, 100, 1e4, 1e5, 1e6, 1.96e7
15	number of energy groups divided into equal parts
30	number of energy groups divided into equal parts
70	number of energy groups divided into equal parts

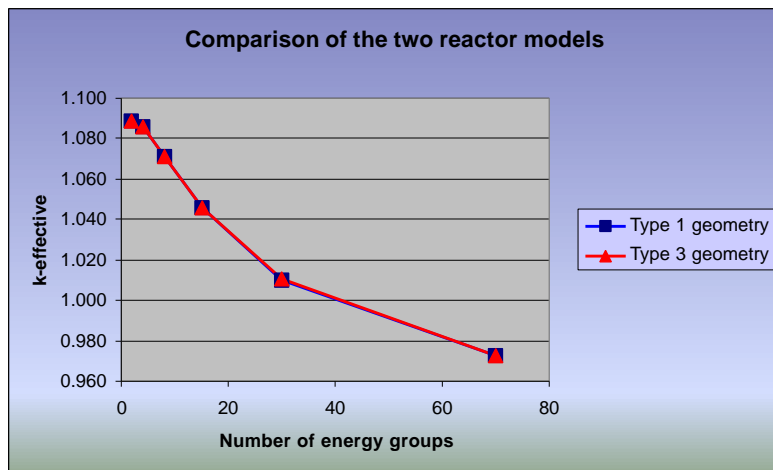


Figure 6.2; The $k_{effective}$ off the two reactor models for different number of energy groups

The impact of the more precise reactor model appears to be negligible. Therefore other modifications were made to the model. The first modification we made was to modify the microscopic transport cross section of the helium according to equation (4.18).

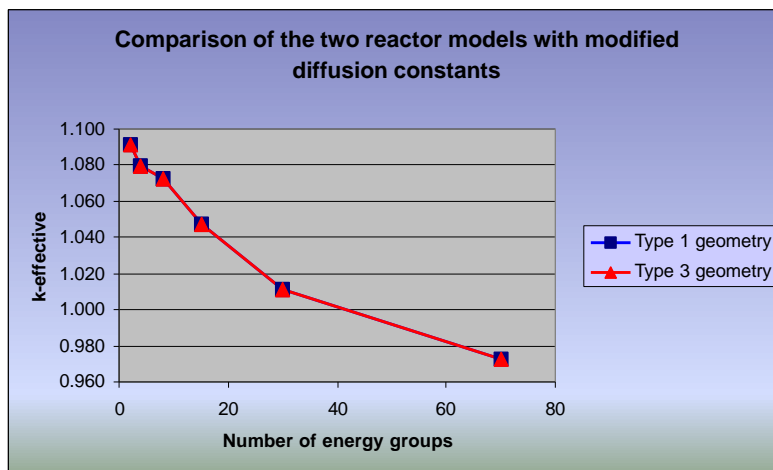


Figure 6.3; The $k_{effective}$ of the two reactor models with limited diffusion constants

Both figures 6.2 and 6.3 show negligible differences between the two reactor models. This is why only one line is visible. Because the results were almost uninfluenced by the changes, further

modifications were made to compensate for the weak points in the diffusion theory. Two heterogeneous meshes were tried with a finer mesh near the border of different regions. The results can be found in figure 6.4. The heterogeneous mesh was only applied to the type three reactor model. The main reason to do this was because the type three reactor model has more zones. For each of these zones a mesh can be defined. So we can specify finer meshes in areas where the flux gradient is large and coarser meshes where the gradient is small. In appendix A, a schematic view of the heterogeneous meshes can be found.

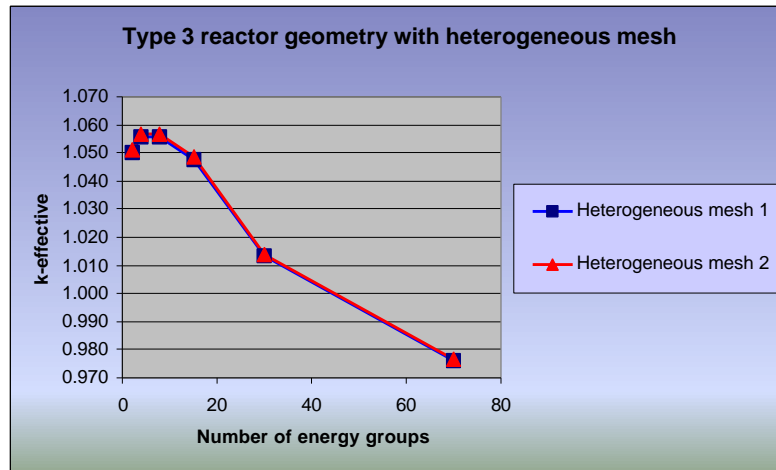


Figure 6.4; The k -effective for the type 3 reactor model on a heterogeneous mesh.

One remark must be made to explain the difference in shape between these results and previous ones. A different energy group structure was used in order to achieve faster convergence, see table 6.2. These changes in energy boundaries were made to emphasise more on the thermal neutron spectrum. In figure 6.4 one can see that the result obtained with 2 energy groups is closer to the result of the detailed model than the 4 or 8 energy group model. So except for the different shape in the calculations with few energy groups the shape of the graph is the same. The shape of the graph is still strongly dependent on the number of energy groups used.

Table 6.2; Upper energy boundaries for different number of energy groups used to obtain the results printed in figure 6.4 and 6.5

Number of Energy groups	Upper boundaries (eV)
2	7.52, 1.96e7
4	1.84, 15.9, 748, 1.96e7
8	0.030, 0.391, 1.93, 15.9, 67.9, 749, 6.74e4, 1.96e7
15	number of energy groups divided into equal parts
30	number of energy groups divided into equal parts
70	number of energy groups divided into equal parts

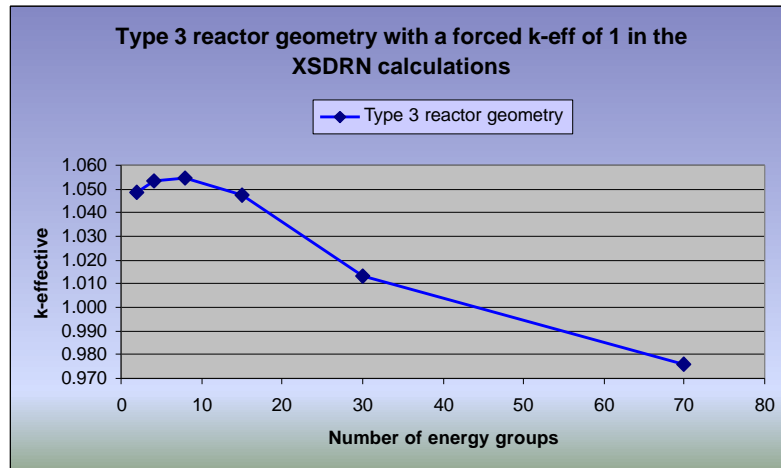


Figure 6.5; The results of the calculations done with a library generated with a forced $k_{\text{effective}}$ of 1 in the XSDRNPM calculations

The last attempt to obtain better results was to compensate for the approximations made by XSDRNPM as explained in paragraph 4.3. These results can be seen in figure 6.5. Since all simulations showed the same behaviour and approximately the same values for the $k_{\text{effective}}$, there might have been something wrong with the cross section library. Therefore we modelled the leakage in the system in XSDRNPM in such a way that the $k_{\text{effective}}$ was set to 1.0.

Again one can see the strong dependency of the FX2 results on the number of energy groups. And secondly one can see that the values obtained for the $k_{\text{effective}}$ are nearly the same. This is in contrast to the XSDRNPM calculations that showed a large drop of the values obtained for the $k_{\text{effective}}$ in both radial and axial directions. The $k_{\text{effective}}$ dropped from 1.10 and 1.07 to 1.0.

From all previous calculations the same conclusions can be made. The value of the $k_{\text{effective}}$ is strongly dependent on the number of energy groups used and there is almost no difference between the type 1 geometry and the type 3 geometry. One might wonder if other codes give the same results. Therefore FX2 was compared with the codes KENO Va and BOLD VENTURE as written in paragraph 4.4. These comparisons are done for the type 1 reactor geometry as well as for the type 3 reactor geometry. These calculations have an extra calculation step in which the unweighted isotopes of carbon, boron and helium are added as explained in paragraph 3.3. The energy group boundaries are placed according to Massimo (L. Massimo 1976), see table 6.3. A complete overview of the values of the calculation can be found in appendix B.

Table 6.3; Upper energy boundaries for different number of energy groups used to obtain the results of the inter-code comparison

Number of Energy groups	Upper boundaries (eV)
2	1.90, 1.96e7
4	0.37, 17.6, 748, 1.96e7
8	0.03, 0.35, 1.93, 15.9, 67.9, 748, 6.74e4, 1.96e7
15	0.03, 0.08, 0.18, 0.39, 1.07, 0.78, 0.986, 1.15, 1.93, 3.38, 15.9, 67.9, 748, 2.93e4, 1.11e6, 1.96e7
30	number of energy groups divided into equal parts
70	number of energy groups divided into equal parts

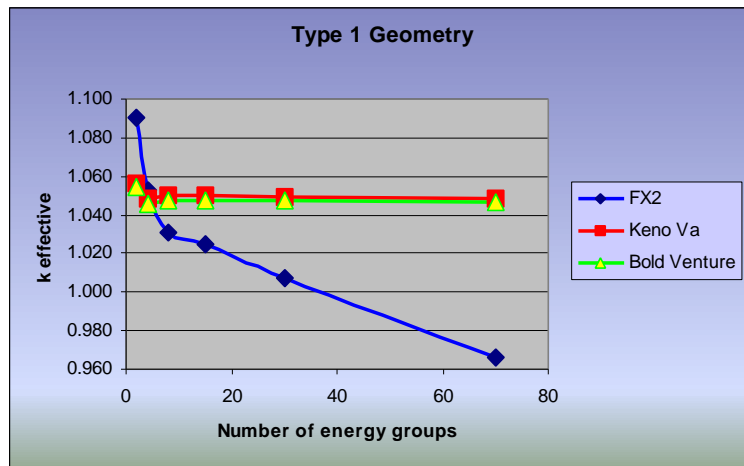


Figure 6.6; The comparison of the different codes for the type 1 reactor geometry

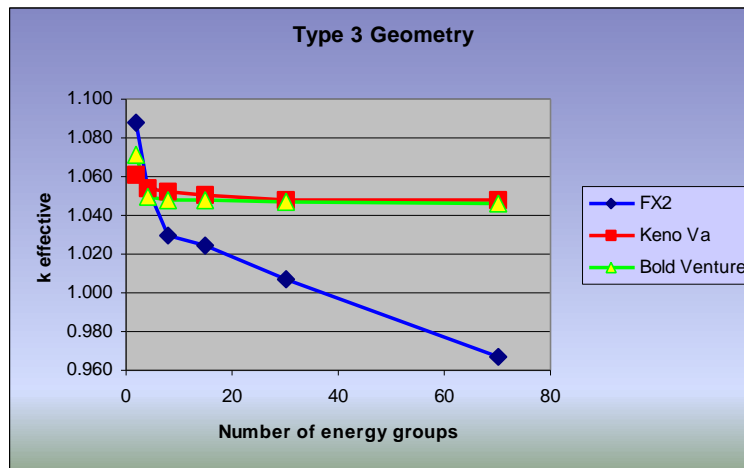


Figure 6.7; The comparison of the different codes for the type 3 reactor geometry

In figures 6.6 and 6.7 one can see clearly the similar results of the KENO Va and the BOLD VENTURE calculations. Results of simulations with 8 or 15 energy groups do not differ much from highly detailed simulations with 70 energy groups. One can say that convergence is reached with 8 or 15 energy groups using BOLD VENTURE or KENO Va. FX2 only reached convergence using 90 energy groups and the result differs much from the results obtained by BOLD VENTURE and KENO Va. With the results of the latter so close together, it is safe to say that a neutron diffusion code can be sufficiently accurate to simulate the ACACIA reactor.

A probable reason for this difference can be found in the manual of the FX2 code. It states that the code was written for liquid metal fast breeder reactors (LMFBR), but that the code can also be used for other type of reactors. It even compares itself with neutronics codes like Venture, but this is only for the relation between the neutron current and the neutron flux that is used internally. But there is one significant difference. FX2 does not take up-scattering into account. Because FX2 was written for LMFBR, this is a suitable approximation because of the fast neutron spectrum.

Some more research was done to find out if the up scattering was the only reason for these differences and the strongly energy group dependent behaviour. Two more series of calculation

were done. The first was to simulate the type one reactor geometry for different sets of energy groups in such a way that no up scattering appeared in the simulation. By taking the lowest energy boundary at 3.38 eV, just above the up scattering limit above which no up scattering occurs, a system was obtained in which no up scattering took place. These results would be compared to the BOLD VENTURE results to see if they would show the same behaviour and obtain the same values for their $k_{\text{effective}}$. If this was the case then all the differences could be explained by the fact that FX2 does not take scattering into account.

The second set of calculations was just the other way around. The energy boundaries would be selected in such a way that as much up scattering as possible would take place. By keeping all the energy group boundaries below the up scattering limit this was obtained. In this case, if the differences between the codes are the result of neglecting the up scattering, FX2 should show again the strong energy group dependency.

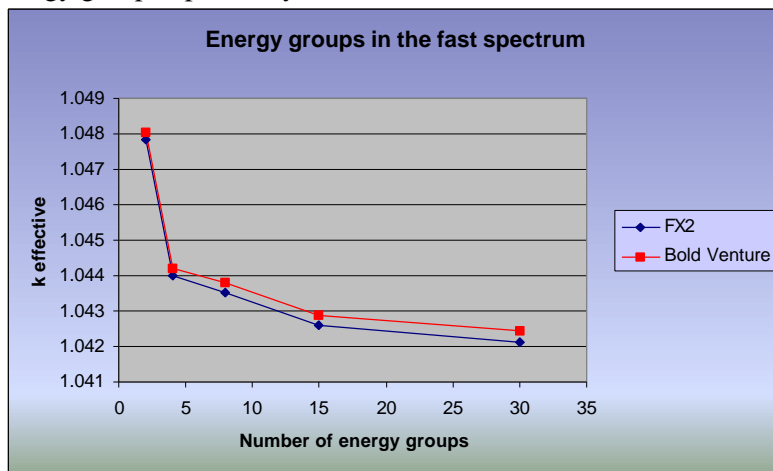


Figure 6.8; Inter-code comparison of the simulations without up scattering

From figures 6.8 and 6.9 it can be concluded that the strong energy group dependent behaviour and the different results calculated by FX2 are the result of neglecting the up-scattering. Future research should determine whether using only energy boundaries above the 3.3 eV can generate good results.

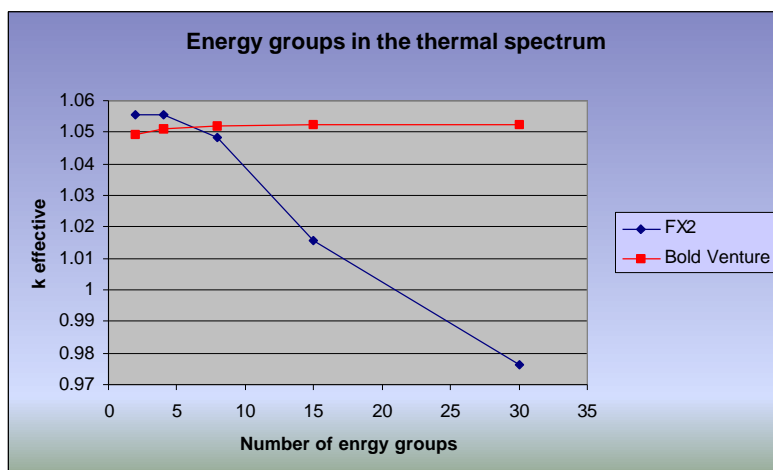


Figure 6.9; Inter-code comparison of the simulation with all but one energy group below the up scatter limit.

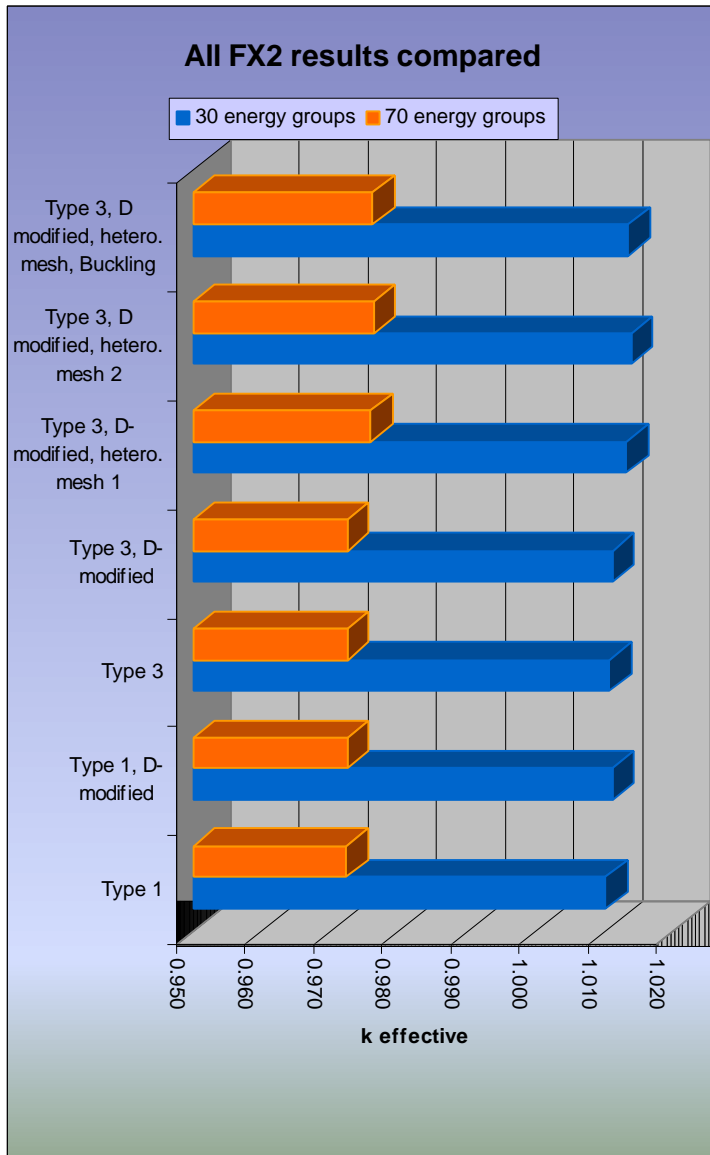


Figure 6.10; Comparison of all FX2 results for 30 and 70 energy groups.

In figure 6.10 the $k_{\text{effective}}$ is shown for all the FX2 calculations. The results are only plotted for the 30 and 70 energy group configuration because of the differences in energy group boundaries for the other configurations. These two energy group structures are the only ones that have the same energy boundaries in all calculations. The only noticeable difference between these calculations is the heterogeneous mesh. The values of the $k_{\text{effective}}$ calculated with the heterogeneous mesh applied are higher than those obtained by the other calculations. But one should pay attention to the x-axis and conclude that the difference is very small for the same number of energy groups. Next to these small differences some similarities can be seen. The most important one is the strong energy group dependency. This is noticeable for all different setups.

Although the heterogeneous mesh did not have the required effect in these calculations, it can be used for speed up of the calculations by applying a coarser mesh in the regions where the flux and the flux gradient are small.

6.2 Aspen Custom Modeller results

The ACM model was built to simulate transients. We are interested in the behaviour of the reactor the first few minutes after a transient initiating event and in the long run of around 50 hours. These simulation times were used to be able to compare results with Verkerk (E.C. Verkerk 2000). We have a total of four models to compare: The original model as used by Kikstra (J.F. Kikstra 2001), we shall refer to as the Reactor Transients Model. The second model and closest to this one, is the model what we will call the RT model, fixed friction. This model has only one difference with the original model. The correct friction factor is used for the helium flow through the pipes as discussed in paragraph 5.5.3. The last two models have some physical modifications.

The Helium Cavity model is the model as described in figure 5.4 and the plenum model is the model as described in figure 5.3 for $i=1$. So this model is the same as the reactor transients model except for two plenums which are placed above the top and below the bottom reflector. We are interested in two transients. The first one is a sudden increase of the helium inlet temperature by 100 K. The second one is a sudden decrease of the helium inlet temperature by 100 K. For each of these two transients three parameters are tracked:

- The normalised power, the total power production normalised to the operating condition of 40 MW_{th}
- The outlet temperature, the temperature of the helium leaving the reactor
- The temperature of the hottest pebble bed element, to see if the critical temperature is reached

The increase in temperature starts at $t = 1$ second and takes 1 second to reach its new temperature. When this rise time is very low it is very hard for ACM to converge. A more important reason however is that an even steeper increase is unlikely to happen in reality.

In figure 6.11 the results of the simulations of a temperature increase for the first 10 minutes can be seen. All models show a decline in normalised power production. This decrease of power production is a result of the behaviour of the reactor, described by the negative temperature coefficient. So an increase in temperature leads to a decrease in power production. A glitch can be seen at 150 seconds in the transient in the results of all simulations. This is due to decrease in power production of the top pebble bed elements. This decrease in power production leads to a decrease in temperature, which in their turn results in an increased power production.

The results of the RT model, fixed friction, are almost indistinguishable from the results of the plenum model. This is the reason that only three lines are visible in figure 6.11. In figure 6.12 and 6.13 one can see that the starting values at $t = 0$ seconds are not the same. This is due to the fact that changes in the model have some consequences at the steady state condition. We configured the reactor in such a way that the nominal power production is 1.0 at the starting point for all the models. Figure 6.12 and 6.13 show both a drop in temperature. This is a direct result of the drop in power production.

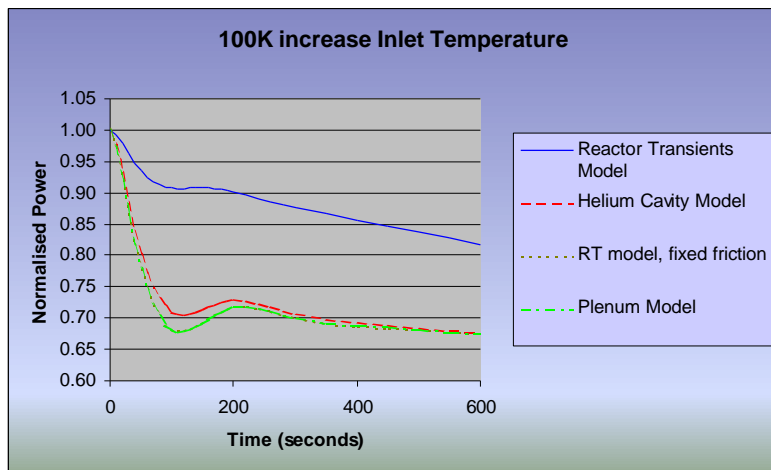


Figure 6.11; The change in normalised power due to an increase in the inlet temperature

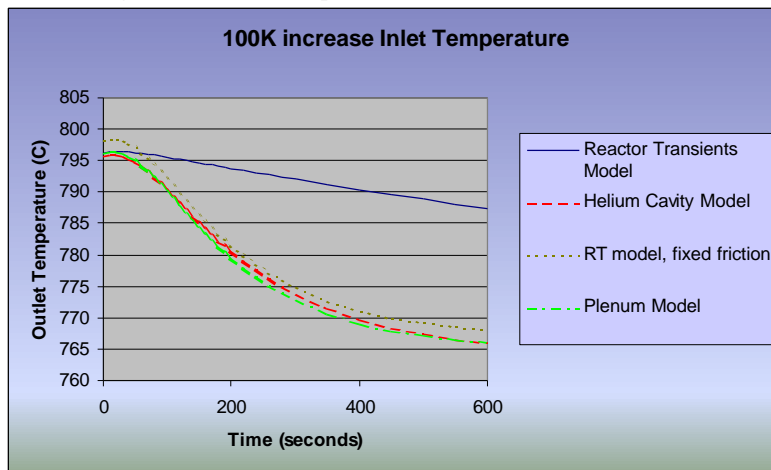


Figure 6.12; The change in outlet temperature due to an increase in inlet temperature

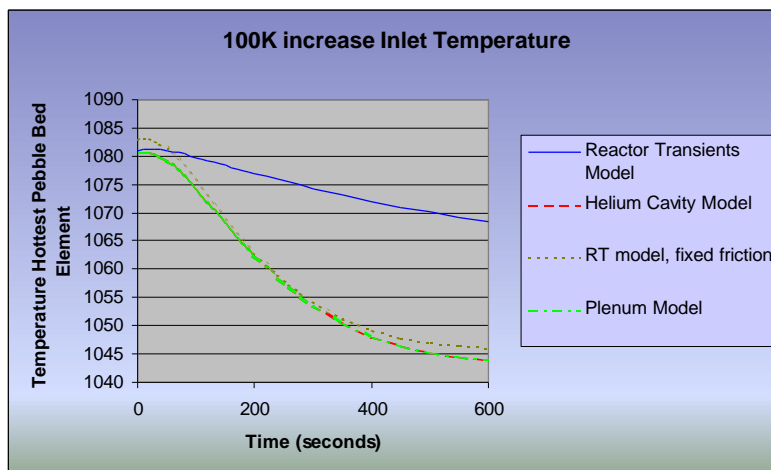


Figure 6.13; The temperature of the hottest pebble bed element due to an increase in inlet temperature

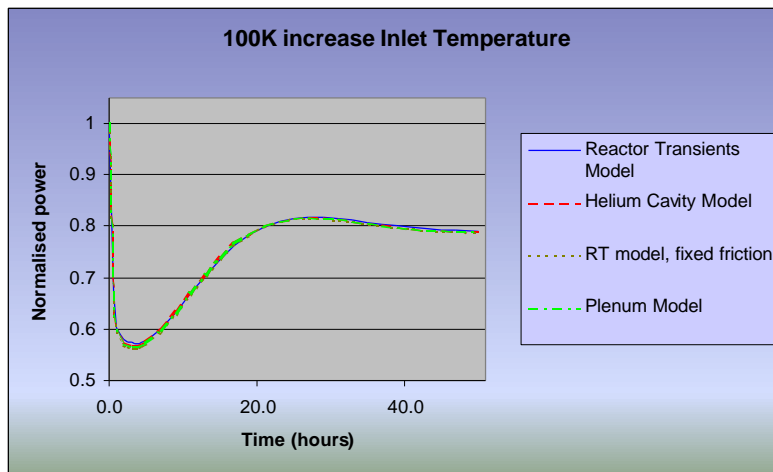


Figure 6.14; The influence of the inlet temperature increase on the power production. Note the time scale up to 50 hours

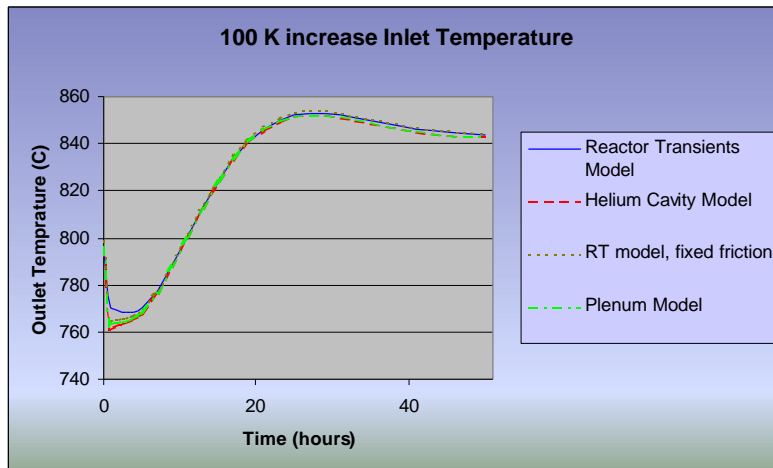


Figure 6.15; The influence of the inlet temperature increase on the outlet temperature. Note the time scale up to 50 hours

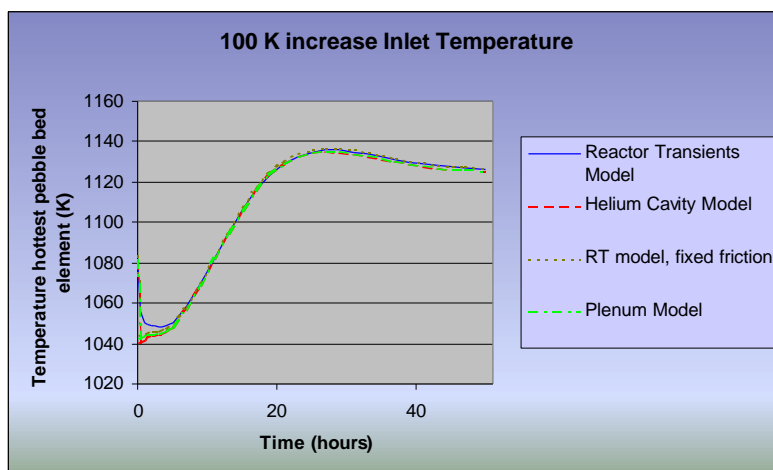


Figure 6.16; The temperature increase in the hottest pebble bed element due to the inlet temperature increase. Note the time scale up to 50 hours

Figure 6.14, 6.15 and 6.16 show the behaviour of the ACACIA reactor in the long run. The power production, figure 6.14, drops fast in the first few hours. This is a direct result of the increase of the inlet temperature. After the first few hours the power production slowly increases again to reach its new steady state value of 0.79. This rise is a result of the drop in xenon concentration. Xenon is a decay product of iodine and a product of nuclear fission described by equation 5.19. This first term in this equation does not react directly to changes in power production, but it reacts with a certain delay. The xenon poisoning is the reason why this graph is shaped this way. The outlet temperature and the temperature of the hottest pebble bed element show very similar behaviour. First a sharp decrease followed by an increase of the temperature to reach finally a higher temperature than the one started with. Although the input temperature is increased by a 100 K, the output temperature and the temperature of the hottest pebble bed element both only increased with 50 K. This is a result of the drop in power production. Between the models there is almost no difference except for the behaviour in the first few hours. Since all models used the same formulas to calculate the reactivity contribution of the xenon poisoning, this similar behaviour can be explained.

For the inherently safe behaviour of the HTR it is very important that the temperature of the pebbles does not exceed the 1600 °C limit (E.C.Verkerk 2000). This is generally taken for the upper limit below which the pebbles will not release any fission products. We can see that in previous simulations the temperature of the pebble bed elements stayed well below this limit. The second type of transient we are interested in is the decrease of the inlet temperature by 100 K.

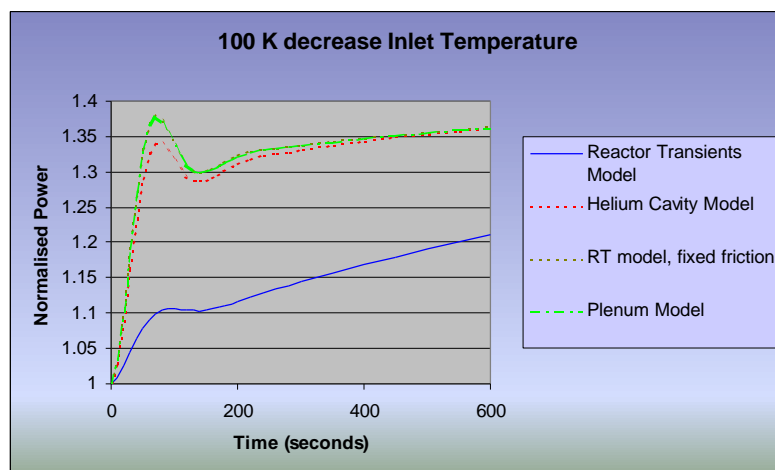


Figure 6.17; The influence of the temperature decrease in inlet temperature to the normalised power

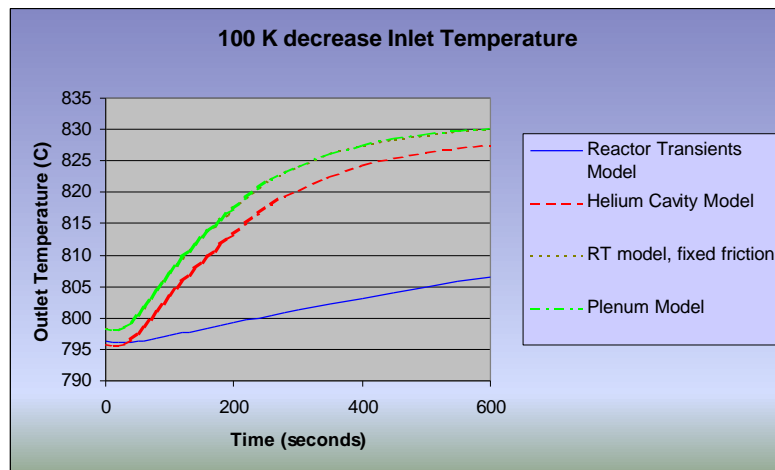


Figure 6.18; The influence of the temperature decrease in inlet temperature to the outlet temperature

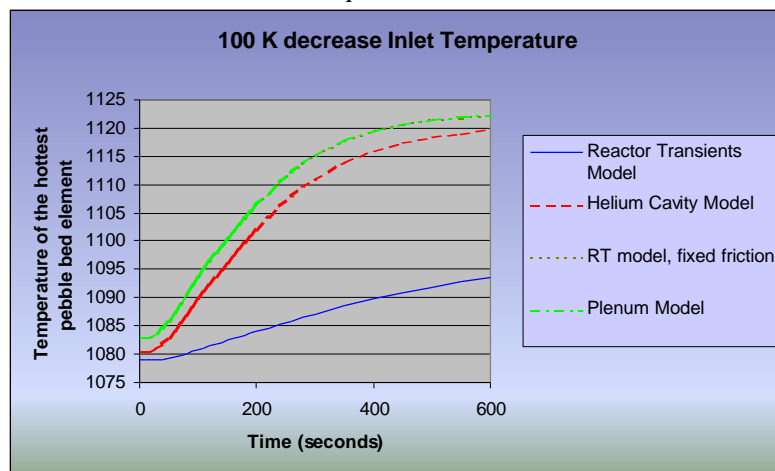


Figure 6.19; The influence of the temperature decrease in inlet temperature to the hottest pebble bed element

In these calculations the same approximation was applied. All models were tuned in such a way that they all started with a normalised power production of 1.0. This resulted in some small differences in the starting values as can be seen in 6.18 and 6.19. The normalised power production, figure 6.17, is nearly exactly the opposite of the normalised power in figure 6.11. First the normalised power increases because of the drop in temperature. Due to the increased power the reactor warms up and with this increased temperature the power production drops again. Exactly the opposite behaviour of the inlet temperature increase. The models with the corrected friction factor show good similarity. But there is a large difference when compared with the model without the corrected friction factor. This applies as well for the outlet temperature as the temperature of the hottest pebble bed element. A decrease of inlet temperature leads to an increase of the outlet and pebble bed temperature due to the increased power production for the first few minutes.

We simulated the behaviour of the reactor in the long run as well (figures 6.20 till and including 6.22).

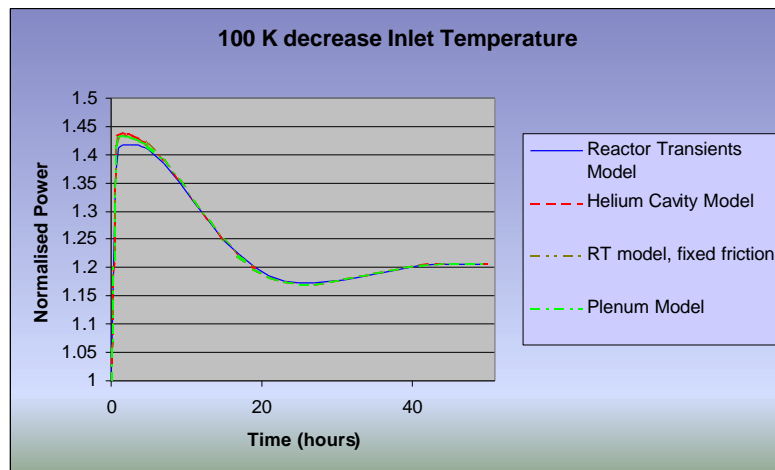


Figure 6.20; Normalised power production after a sudden decrease of the inlet temperature. Note the time scale up to 50 hours

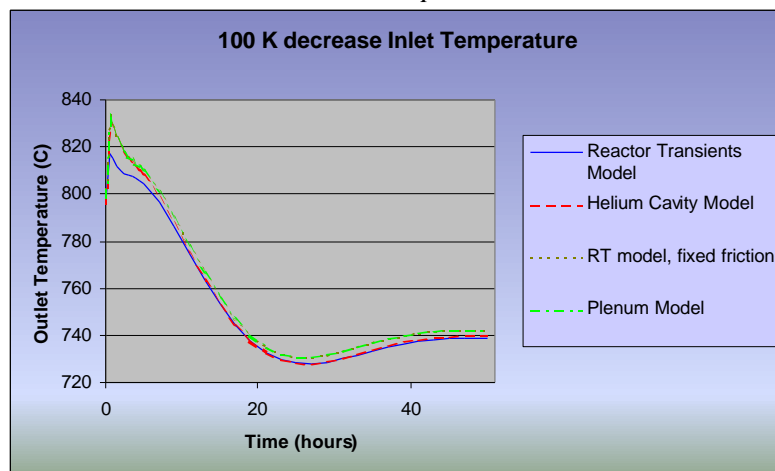


Figure 6.21; Outlet temperature after a sudden decrease of the inlet temperature. Note the time scale up to 50 hours

One can immediately notice that in the long run there are only small differences between the models. The behaviour is opposite to that after an inlet temperature increase. The power production increases due to the inlet temperature drop. The increased power production leads to an increase in xenon concentration with a delay of several hours. The xenon poisoning reduces the reactivity and thereby the power production. This drop results in an decrease in xenon production, that results after the several hours delay in an slight increase in power production. The xenon poisoning of the reactor is the most important factor to explain the shapes of the graphs. Since there were no differences between algorithms of the models to calculate the reactivity effect of the xenon poisoning the results are very similar. In figure 6.23 the reactivity curves are shown of various reactor elements. Remember that the reactivity function took three contributions into account: temperature, xenon and external. This last one was fixed to 0, so only two contributions remain.

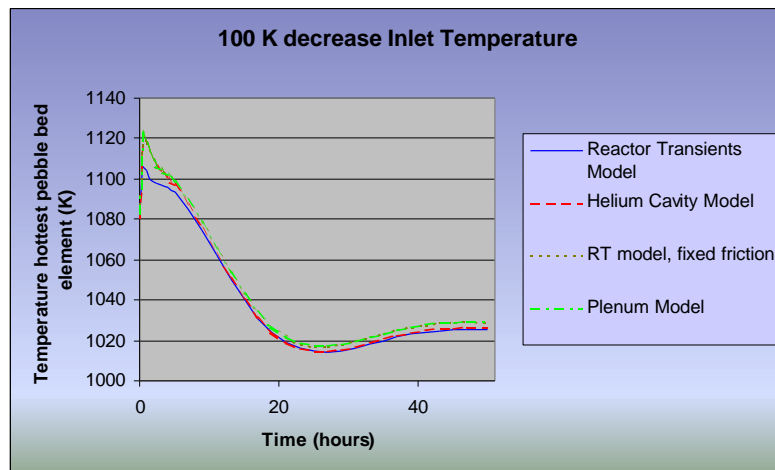


Figure 6.22; The temperature of the hottest pebble bed element after a sudden decrease of the inlet temperature.

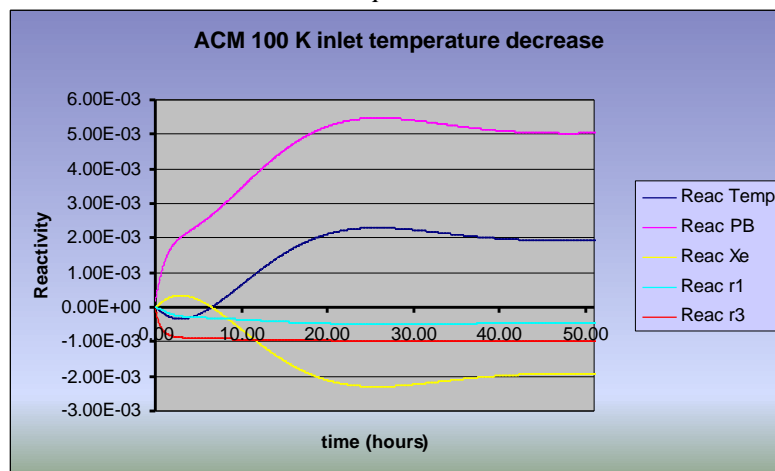


Figure 6.23; Reactivity contribution of different parts of the reactor in the long run after a sudden decrease in the inlet temperature.

Figure 6.23 shows the reactivity curves for the long run. Reac_Temp stands for the reactivity contribution of all elements in the reactor due to a change in temperature. Reac_PB stands for the contribution of the pebble bed elements, Reac_r1 and Reac_r3 for the contributions of respectively the first radial reflector and the third. One can see that the contribution of the xenon and that of the temperature cancel each other almost out after a few hours. This means that there is only a relatively slow change of normalised power in time as can be seen in figure 6.20. The positive contribution of the xenon poisoning comes because the neutron flux increases due to the inlet temperature drop. From (5.18) one can see that a sudden increase in flux results in a direct decrease of the xenon concentration. After a few hours the xenon concentration increases again due to the decay of iodine, the most important source of xenon. In this figure the reactivity contribution of some reflectors is plotted. These reflectors show the opposite behaviour of the pebble bed elements. Therefore their behaviour can be described by a positive temperature coefficient.

7 Conclusions and recommendations

7.1 The FX2 model

In the manual of the FX2 code, the writers claim that this code is also suitable for simulating reactors different from the LMFBR. This might be true, but we can now say that it is not suitable for simulating the ACACIA reactor with an energy group boundary below 3.3 eV. From figures 6.8 and 6.9 one can see that the energy group dependency is directly related to the number of up scatter groups. It becomes difficult now to interpret the other results. However, one thing is clear: the influence of neglecting the up scattering is dominant over all other changes. The most important question that needs to be answered is: “Is a 3.3 eV upper boundary for the thermal group sufficiently accurate?” Further research should be focused on this question.

The most noticeable, but still very small, difference in results is in the simulations with the heterogeneous mesh. The fine mesh near material boundaries is successful. On the other hand a heterogeneous mesh can also be used for speed up by applying a coarser mesh in zones where the flux and the flux gradient are small.

The two reactor geometry models have in steady state almost no impact on the results obtained. One might conclude that the type 1 geometry is sufficiently accurate, but this is only true in steady state situations. For dynamic situations more research is needed. If one would apply the type 3 geometry model to any code, one would be able to define the temperature in more zones. Since this is one of the most important properties our model should have, one might favour the type 3 geometry model. It might even be necessary to simulate the core in more zones.

Nevertheless BOLD VENTURE has shown that a neutron diffusion code is sufficiently accurate to simulate the behaviour of the ACACIA reactor. The BOLD VENTURE code could even be improved when one would create a routine that could interpolate between cross-section libraries obtained at different temperatures. In this case one of the most favourable abilities of FX2 could be integrated in BOLD VENTURE.

7.2 The ACM model

Experience with ACM has taught us a lot about its strength and its weaknesses. The main disadvantage is its often incorrect help debugging; Either fixing the number of degrees of freedom or getting the simulation to converge. One can try to avoid this first problem by taking very small steps when making modifications and very careful programming. This does however take a lot of time and requires more programming to create routines necessary for intermediate steps. Even then the problem with the number of degrees of freedom cannot always be avoided.

The second problem is more a property of the model itself. Experience has taught us that the convergence problems are most of the time related to the flow. In order to avoid problems in the future and to make the model more versatile, a new flow model should be developed. The requirements for this model are primarily stability and simplicity so that modifications can be made more easily. These requirements are made because at first we wanted to achieve a model that is fully capable of simulating 2 dimensional flow, not to be confused with parallel flow. Calculations done by Verkerk (E.C. Verkerk 2000) show that the fuel temperature reaches a maximum when a loss of flow accident cuts the flow rate to 5 –10 % of its original value. This proves the importance of simulating the flow at low velocities. When this occurs it is also more

likely that there is a flow from one radial element to the other. Therefore the model should be capable of modelling the transverse flow due to natural convection. In conclusion I would say that ACM is a very difficult and obscure code to work with. In my opinion the time necessary to make a simple modification is too long. Therefore replacing the ACM core model with a new Fortran routine is a serious option.

The four models presented here can be divided into two groups, the models with the corrected friction factor, and the one without. The impact of the corrected friction factor is very large in the short run. We can see that the reactor reacts faster to changes at the helium inlet. In the long run the differences are negligible. The friction factor is used to calculate the force the wall has on the helium flow, which is the most important part of the equation that calculates the mass flow. The friction factor of the helium flowing through pipes is around 80 times lower than that of helium flowing through a pebble bed. This results in a higher velocity of the helium and explains why the reactor reacts faster to inlet temperature changes.

Next to this change the models with the corrected friction factor reach higher normalised power and higher temperatures on an inlet temperature decrease than the uncorrected model. It also reaches lower values on an inlet temperature increase. Due to the lower friction factors the helium flow passes through the reactor faster. So if we take a look at a decrease of the inlet temperature, the amount of heat transferred to the helium will be less before it enters the core. The helium will cool the top pebble bed element more, resulting in a higher normalised power. The same applies, but the other way around of course, for the increase in helium inlet temperature.

Between the three models with a fixed friction factor the differences are a lot smaller. The difference between the plenum model and the reactor transients model with corrected friction coefficients is the existence of two plenums above the top and under the bottom reflector. Since these plenums have no influence on the heat conduction and do not contribute anything to the reactivity, there is no reason to expect any major difference between these two models. On the other hand, the helium cavity model shows a different behaviour. As can be seen in figure 5.4 a total of nine extra routines have been added. These routines have consequences on the heat transfer but not directly on the reactivity calculation. The latter is due to the fact that no reactivity curve for these elements was available. There is some difference in the total reactivity. Since the reflector 1 top elements, reflector two top etc. are now farther away from the core, they will receive less heat from the core. The model, however, ignores heat loss to the top and the bottom and this is why in steady state the differences are very small. During transients the differences can grow from a few degrees up to 50 degrees. This is not negligible. The temperature of the helium cavity model reflectors should be included in the reactivity calculations. This explains part of the difference with the other models.

The reactor transients model with fixed friction factors is sufficiently detailed to describe the behaviour of the ACACIA reactor in the long run. The calculation time necessary to simulate the transients was nearly the same for all models. Therefore it is not necessary to make as much simplifications as possible since their impact on the calculation time is negligible. It ranges from a minute to calculate the behaviour of the reactor during the first 600 seconds to several minutes for the 50 hours transients. All results obtained from these simulations support the proposition that the high temperature gas cooled pebble bed nuclear reactor is inherently safe.

References

1. A.I.van Heek 1997, *INCOGEN Pre-Feasibility Study, Nuclear Cogeneration*, ECN Petten.
2. B.G.Carlson 1953, *Solutions of the Transport Equation by S_n Approximations*, Los Alamos Scientific Laboratory, LA-1599.
3. D.R.Vondy, T.B.Fowler, & G.W.Cunningham III. BOLD VENTURE: A Reactor Analysis Code System, Version IV. 1981. Oak Ridge (TN), VS, Oak Ridge National Laboratory.
Ref Type: Computer Program
4. E.C.Verkerk 2000, *Dynamics of the Pebble-Bed Nuclear Reactor in the Direct Brayton Cycle*, Delft University of Technology.
5. E.E.Bende 2000, *Plutonium burning in a pebble bed type High Temperature Nuclear Reactor*, Delft University of Technology.
6. F.Daniels 1944, *Suggestions for a High Temperature Pebble Pile.*, Oak Ridge National Laboratory, Oak Ridge (TN), VS.
7. H.Gerwin & W.Scherer 1987, "Treatment of upper cavity in a pebble-bed high temperature gas cooled reactor by diffusion theory", *Nuclear Science and Engineering* no. 97, pp. 9-19.
8. I.I.Bondarenko 1964, *Group Constants for Nuclear Reactor Calculations*, Authorised Translation from the Russian, Consultants Bureau, New York.
9. J.E.Hoogenboom & H.van Dam 1998, *Kernreactorkunde* Delft University of Technology, Delft.
10. J.F.Kikstra 2001, *Modelling, Design and Control of a Cogenerating Nuclear Gas Turbine Plant*, Delft University of Technology.
11. K.Kugeler & R.Schulten 1989, *Hochtemperaturreaktortechnik* Springer.
12. L.M.Petrie & N.F.Landers. KENO Va: An improved Monte Carlo program with supergrouping. 1-10-1993. Oak Ridge (TN), VS, Oak Ridge National Laboratory.
Ref Type: Computer Program
13. L.Massimo 1976, *Physics of hightemperature reactors* Pergamon Press, Exeter, UK.
14. L.W.Nordheim, G.Birkhoff, & E.P.Wigner 1961, "The Theory of Resonance Absorption", *Am. Math. Soc.*, p. 58.
15. N.F.Landers & L.M.Petrie. CSAS, Control module for enhanced Criticality Safety Analysis Sequences. 9-1-1998. Oak Ridge (TN), VS, Oak Ridge National Laboratory.
Ref Type: Computer Program
16. N.M.Greene. BONAMI, Resonance Self Shielding by the Bondarenko method. 9-1-1998. Oak Ridge (TN), VS, Oak Ridge National Laboratory.
Ref Type: Computer Program
17. N.M.Greene & L.M.Petrie. XSDRNPM: A one-dimensional discrete-ordinates code for transport analysis. 9-1-1998. Oak Ridge (TN), VS, Oak Ridge National Laboratory.
Ref Type: Computer Program

18. N.M.Greene, L.M.Petrie, & R.M.Westfall. NITAWL-II: Scale system module for performing resonance shielding and working library production. 9-1-1998. Oak Ridge (TN), VS, Oak Ridge National Laboratory.
Ref Type: Computer Program
19. N.M.Greene, W.E.Ford III, L.M.Petrie, & J.W.Arwood. AMPX-77: A modular code system for generating coupled multigroup neutron-gamma cross section libraries from ENDF/B-IV or ENDF/B-V. 1-10-1992. Oak Ridge (TN), VS, Oak Ridge National Laboratory.
Ref Type: Computer Program
20. R.A.Shober, T.A.Daly, & D.R.Ferguson. FX2-TH: A two-dimensional nuclear kinetics code with thermal-hydraulic feedback. 1-10-1978. Argonne (IL), VS, Argonne National Laboratory.
Ref Type: Computer Program
21. VSOP. Computer Code System for Reactor Physics and Fuel Cycle Simulations, Input Manual and Comments. 1994. Julich, Germany, Forschungszentrum Julich.
Ref Type: Computer Program
22. Yan, X. L. 1990, *Dynamic Analysis and Control System Design for an Advanced Nuclear Gas Turbine Power Plant*, MIT.

Appendix A, Heterogeneous meshes

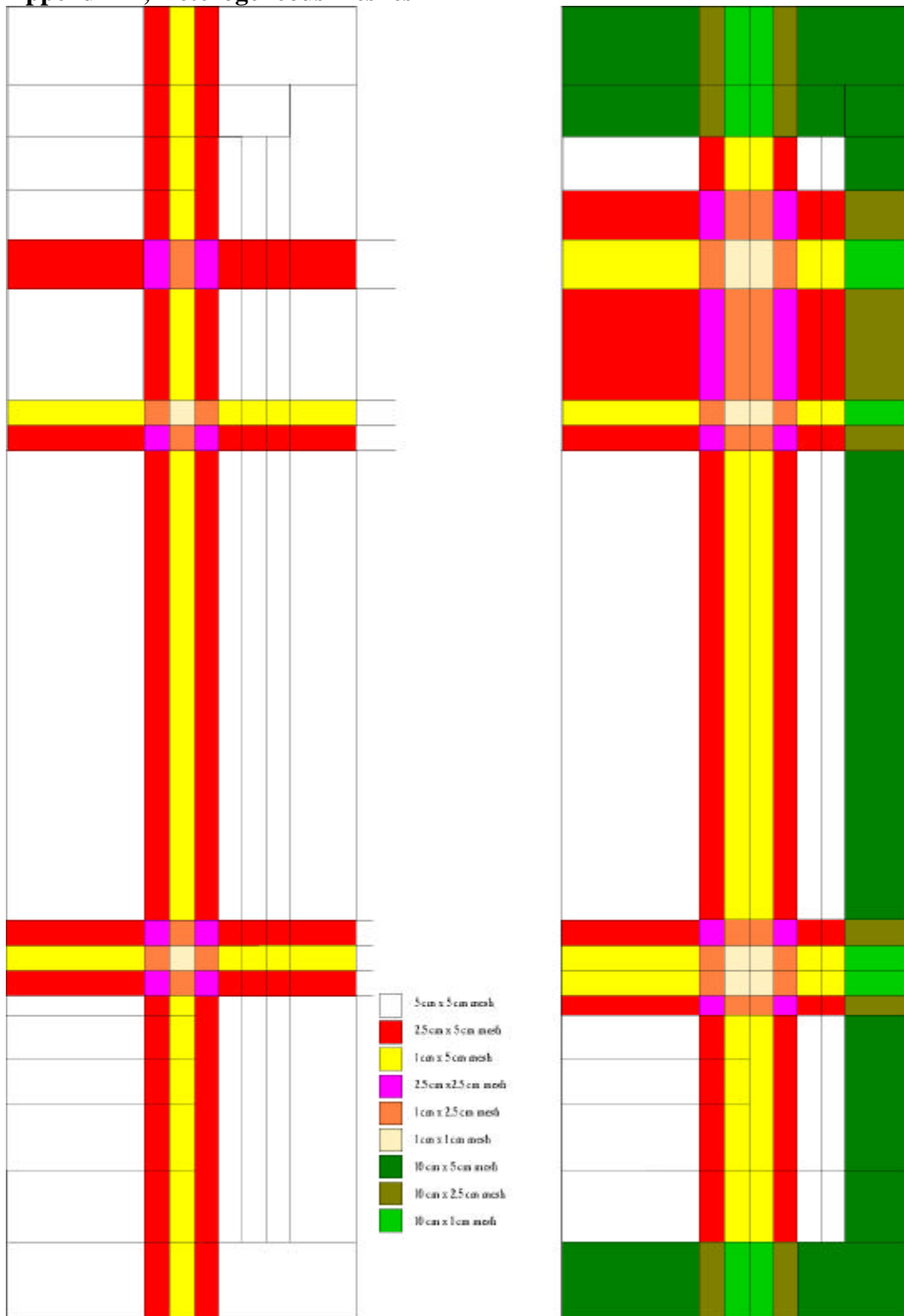


Figure A.1; Overview of the two heterogeneous meshes, Mesh 1 on the left and two on the right.

Appendix B, Tables with the FX2 results

Table B.1; The numerical results of the FX2 calculations

Number of Energy groups	Type 1	Type 1, D-modified	Type 3	Type 3, D-modified	Type 3, D-modified, hetero. mesh 1	Type 3, D modified, hetero. mesh 2	Type 3, D modified, hetero. mesh, Buckling
2	1.08849	1.09100	1.08864	1.09100	1.05025	1.05101	1.04861
4	1.08607	1.07951	1.08626	1.07951	1.05596	1.05661	1.05336
8	1.07131	1.07253	1.07148	1.07253	1.05589	1.05657	1.05488
15	1.04592	1.04701	1.04616	1.04701	1.04772	1.04842	1.04746
30	1.01024	1.01121	1.01075	1.01121	1.01325	1.01394	1.01346
70	0.97234	0.97259	0.97246	0.97259	0.97591	0.97642	0.97616

Table B.2; The numerical results of the inter-code comparison calculations

Number of Energy groups	K-eff, type 1	K-eff, type 3	Keno, type 1, mixedlib	Keno, type 3, mixedlib	Bold Venture, type 1, 5cm	Bold Venture, type 3, 5cm
2	1.09055	1.08820	1.05590	1.06090	1.05409	1.07156
4	1.05252	1.05424	1.04860	1.05390	1.04576	1.04925
8	1.03088	1.02968	1.05010	1.05230	1.04727	1.04781
15	1.02518	1.02405	1.05020	1.05010	1.04754	1.04743
30	1.00711	1.00678	1.04960	1.04800	1.04715	1.04655
70	0.96655	0.96661	1.04880	1.04750	1.04677	1.04606

Table B.3; the numerical results of the simulations with all energy group boundaries either in the thermal or the fast energy range.

Energy groups	FX2, no up scattering	Bold Venture, no up scattering	FX2, thermal energy boundaries only	Bold Venture, thermal energy boundaries only
2	1.04784353	1.0480521	1.0556430	1.0492163
4	1.04400237	1.0442032	1.0554413	1.0511556
8	1.04351545	1.0438014	1.0485413	1.0519258
15	1.04259395	1.0428909	1.0157794	1.0523106
30	1.04211572	1.0424222	0.9762167	1.0524381