Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems
Part II: Prismatic Reactor Cross Section Generation

Vincent Descotes
Javier Ortensi

March 2011

Idaho National Laboratory

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems
Part II: Prismatic Reactor Cross Section Generation

Vincent Descotes
Javier Ortensi

March 2011

Idaho National Laboratory
Fuel Cycle Research & Development
Idaho Falls, Idaho 83415

http://www.inl.gov

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

March 2011   iii

Co-Authors:

Javier Ortensi   3/30/2011

Javier Ortensi for Vincent   Date
Descotes and Javier Ortensi

Reviewed by:

3/30/2011

Michael A. Pope   Date

Concurred by:

A.M. Ougouag   03/30/2011

Work Package Manager   Abderrafi M. Ougouag   Date

**Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor
Cross Section Generation**

iv   March 2011

Page Intentionally Left Blank

**Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor
Cross Section Generation**

iv   March 2011

EXECUTIVE SUMMARY

In Very High Temperature Reactors (VHTRs), the long mean-free-path and large migration area of neutrons leads to spectral influences between fuel and reflector zones over long distances. This presents significant challenges to the validity of the classic two-step approach of cross section preparation wherein infinite lattice transport calculations are performed on relatively small physical domains (e.g. single assembly) in order to compute homogenized few-group cross sections for whole core analysis.

Effects of the inner and outer reflectors challenge the classical two-step approach, while burnable poison locations affect neighboring assemblies as well. Use of transuranics-only (TRU) Deep Burn fuel in a prismatic VHTR (DB-VHTR) presents the additional challenge of producing vastly different neutron spectra between fresh and burned fuel.

Modeling the VHTR also requires the ability to treat in the code the hexagonal geometry, the presence of TRISO particles, whose distribution in the fuel compacts is stochastic, and large lattice domains. Currently, DRAGON can treat these features: the hexagonal geometry is in place, the large domains may be treated by the method of characteristics (MOC) or collision probability (CP), and the double-heterogeneity treatment can be used for TRISO particles.

Therefore, in order to complement the analysis and design activities for Deep Burn in prismatic High Temperature Reactors (HTRs), this study was performed in order to provide some initial guidance on the necessary procedures to model the DB-VHTR with sufficient accuracy. The purpose of this study is to provide evaluation of the tradeoffs in accuracy between domain size in the lattice calculation and number of energy groups used in the whole core calculation. The lattice calculations were performed using the DRAGON-4 code and the whole core calculations using the Idaho National Laboratory code INSTANT, both of which are described in the following sections. The accuracy of the whole-core calculations was judged by comparison of Eigenvalues and fission rate shapes to results from MCNP calculations. The number of groups used in the lattice calculations was fixed at 295 and two different domain sizes were used, a single block and a supercell. Primarily of interest in this study were the effects of the reflector on the spectrum and, in turn, the adequacy of the cross section sets.

Also included in this report are results from a preliminary study aimed at determining the effect of the presence of reflector on the burnup characteristics in a peripheral block. Depletion was performed in a supercell in order to observe the differences across a block of interest.

The results showed that using the Method of Characteristics (MOC) in DRAGON, both a single block and a supercell leads to good predictions of the Eigenvalue and of the relative fission rates of each fuel block in the core if enough groups are kept in INSTANT (e.g., 26 groups). The supercell path produced the best results compared to the equivalent calculations using the single-block path. A k_{eff} within ±20 pcm from MCNP was obtained in both cases. The fission rate errors for fuel blocks were within ±1% with the supercells model, and within [-1%, +1.9%] with the single-block path, compared to the values from MCNP. Calculations with the collision probability (CP) and interface current coupling method proved to be less accurate, especially for supercells, than the MOC. However, considering the good agreement reached by the single-block path with MCNP and considering the significant calculation durations for the supercells, the use of supercells may not appear worthwhile for this simplified core.

Several concerns remain and may play against a calculation using only the single-block model. First, the introduction of burnable poisons and control rods may create geometric difficulties and spectral effects that would not be carried to the core model appropriately using a single-block model, even with many groups in the core calculation. It is especially important to correctly model the burnable poison effects because they strongly affect the local flux.

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

vi   March 2011

Another major challenge remains the depletion calculation in the code. Trials on a supercell have shown that the number densities across a block partially surrounded by reflector vary in significant proportions, up to 45% between two sides of the block for certain isotopes. In addition, the neutron spectrum also changes significantly depending on whether the reflector is close or not to the considered fuel cell. Thus, at least two important phenomena are present in those blocks: a high variation of the fission rates, which tends to create a heterogeneous burning in the block; and a spectral variation. Determination of an adequate depletion method should be pursued in future work.

The material in this report was also prepared as a master's thesis by Vincent Descotes at the École Polytechnique de Montréal under the supervision of Alain Hébert, and Guy Marleau based largely on work performed during an internship at the Idaho National Laboratory.

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

March 2011   vii

CONTENTS

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

March 2011 ix

FIGURES

TABLES

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

xiv

March 2011

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

March 2011   xv

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

xvi   March 2011

LIST OF ACRONYMS AND ABBREVIATIONS

ASCII	American Standard Code for Information Interchange
CEA	Commissariat à l'énergie Atomique
CP	Collision Probability. Solution method of the transport equation.
CPU	Central Processing Unit
DH295	"Double heterogeneity", 295 groups. Describes one calculation path where the geometry is completely described with TRSIO particles
DOE	Department of Energy
DRAGON	Complete lattice code developed at the "École Polytechnique de Montréal"
EXCELT	Tracking module in DRAGON. Used for the method of characteristics
FEM	Finite Element Method
HOM295	"Homogeneous", 295 groups. Calculation path in 295 groups starting with homogeneous cross-sections for the different cells (fuel, graphite, coolant). HOM26 "Homogeneous", 26 groups. Calculation path in 26 groups starting with homogeneous cross-sections for the different cells (fuel, graphite, coolant)
INL	Idaho National Laboratory
INSTANT	Full-core P_N calculation code developed at the INL
LANL	Los Alamos National Laboratory
LWR	Light-Water Reactor
MCNP	Monte-Carlo Neutral Particle reference code developed at the LANL
MOC	Method of Characteristics. Solution method of the transport equation.
NGNP	Next Generation Nuclear Plant
P_N	Solution method of the transport equation using a polynomial expansion of the streaming operator
SERPENT	Monte-Carlo lattice code developed at VTT (Finland) by Jaakko Leppänen
SB	Single block model
SC	Supercell model
S_N	Solution method of the transport equation using an angular discretization of the streaming operator. Also known as the discrete ordinates method
SPH	Super Homogenization
SYB	Abbreviation for SYBILT
SYBILT	Tracking module in DRAGON using the collision probability method
TRISO	Tristructural-Isotropic
VHTR	Very high temperature reactor
VTT	Valtion Teknillinen Tutkimuskeskus

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

March 2011   xvii

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

xviii

March 2011

FINAL REPORT ON UTILIZATION OF TRU TRISO FUEL AS APPLIED TO HTR SYSTEMS PART II: PRISMATIC REACTOR CROSS SECTION GENERATION

1.   INTRODUCTION

1.1   Background

In Very High Temperature Reactors (VHTRs), the long mean-free-path and large migration area of neutrons leads to spectral influences between fuel and reflector zones over long distances. This presents significant challenges to the validity of the classic two-step approach of cross section preparation wherein infinite lattice transport calculations are performed on relatively small physical domains (e.g. single assembly) in order to compute homogenized few-group cross sections for whole core analysis.

Effects of the inner and outer reflectors challenge the classical two-step approach, while burnable poison locations affect neighboring assemblies as well. Use of transuranics-only (TRU) Deep Burn fuel in a prismatic VHTR (DB-VHTR) presents the additional challenge of producing vastly different neutron spectra between fresh and burned fuel.

Modeling the VHTR also requires the ability to treat in the code the hexagonal geometry, the presence of TRISO particles, whose distribution in the fuel compacts are stochastic, and large lattice domains. Currently, DRAGON can treat these features: the hexagonal geometry is in place, the large domains may be treated by the method of characteristics (MOC) or collision probability (CP), and the double-heterogeneity treatment can be used for TRISO particles.

The purpose of this study is to provide evaluation of the tradeoffs in accuracy between domain size in the lattice calculation and number of energy groups used in the whole core calculation. The lattice calculations were performed using the DRAGON-4 code and the whole core calculations using the Idaho National Laboratory code INSTANT, both of which are described in the following section. The accuracy of the whole-core calculations was judged by comparison to MCNP. The number of groups used in the lattice calculations was fixed at 295 and two different domain sizes were used, a single block and a supercell. Primarily of interest in this study was the effects of the reflector on the spectrum and, in turn, the adequacy of the cross section sets.

1.2   Simplified Design Used for this Study

This report will focus on the neutronic simulation of a Deep Burn VHTR. A simplified reactor with only blocks of reflector and blocks of fuel was considered. The reactor arrangement consisted of a prismatic geometry with a 5-ring annular core surrounded by blocks of graphite reflectors.

The fuel blocks are comprised of three types of hexagonal cells:

- Fuel compacts surrounded by graphite and filled with transuranic elements. The fuel is located inside tristructural-isotropic (TRISO) particles diluted in a graphite matrix.

- Cylindrical tubes of coolant with two different diameters filled with helium and surrounded by graphite.

- Pure graphite hexagonal cells in the center and at the periphery. The usual places for burnable poisons in the corners of the fuel block are also filled with pure graphite. A drawing is provided in Figure 1-1(a).

Figure 1-1(b) shows the two-dimensional (2-D) core map, with a central reflector block, two rings of reflector, then five rings of fuel blocks and three rings of reflector blocks before the vessel. The vessel's influence is not considered in this analysis as its influence on the neutronics should be low compared to the outer graphite. Therefore, this study focuses on the interaction between the fuel and the graphite reflector. Likewise, this study assumes that all fuel blocks are equivalent, made up of fresh fuel, without burnable poisons or control rods.

Table 1-1 gives the parameters used for all our calculations.
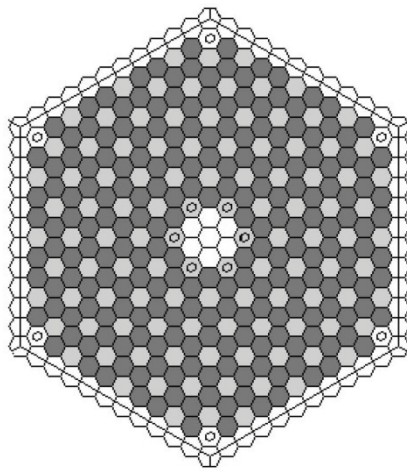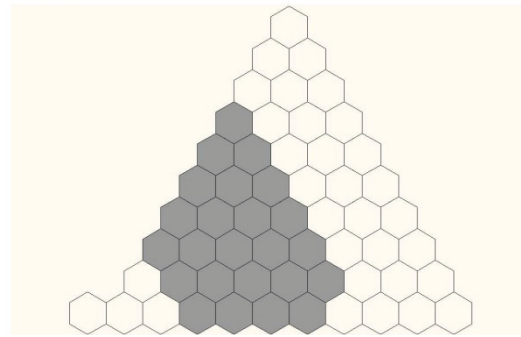
| (a) Geometry of a fuel block. | (b) Core map for INSTANT model. |

Figure 1-1. Core map and detailed geometry of a fuel block.

Table 1-1. Core parameters.

Parameter	Value
Block pitch (cm)	36.0
Block side (cm)	20.78461
Maximum number of blocks in a direction (N)	21
Total number of blocks (T)	331

The following relation, given in the INSTANT manual[1] gives us the correspondence between the number of hexagons in a direction and the total number of hexagons in the core for a 2-D "complete" geometry:

$$T = \frac{3}{4}(N^2 - 1) + 1 \tag{1-1}$$

The fuel blocks exhibit a 1/12th symetry, so the model in the simulation can be simplified. Figure 1-2 shows the disposition of the different cells in the 1/12th model and Table 1-2(a) gives the geometric parameters. The reflector blocks are filled with graphite whose density is the same as the graphite cells in

the fuel block: $8.774.10^{-2}.10^{24}$ atoms.cm^{-3} (see Table 1-4). Tables 1-3(a) and 1-3(b) give the composition of the TRISO particles, and Table 1-2(b) their dimensions.
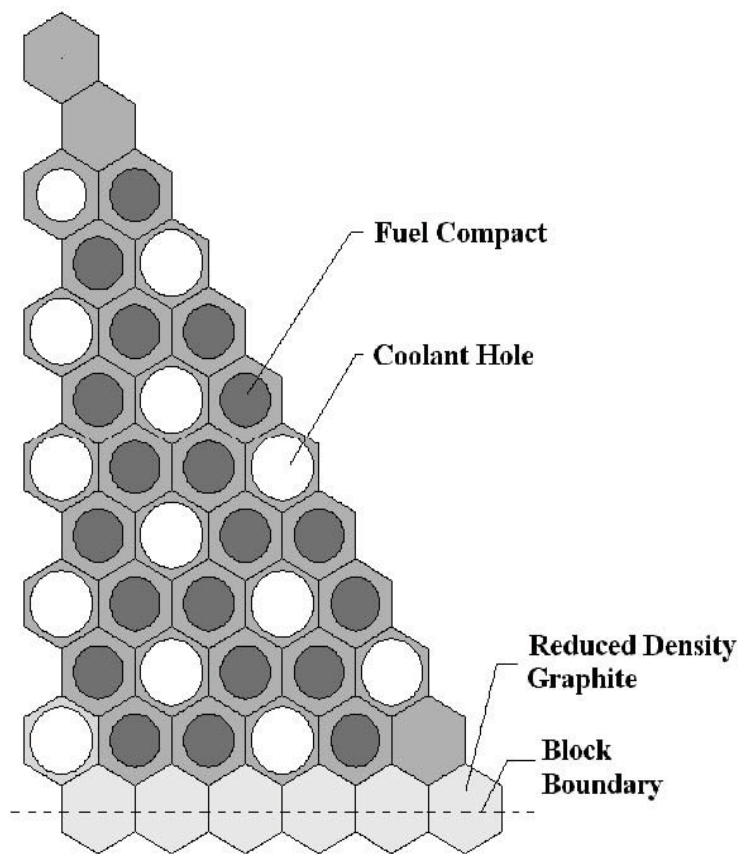
Figure 1-2. Fuel block with 1/12th symetry.

Table 1-2. Fuel block parameters.

(a) Cell dimensions.		(b) TRISO particle layer dimensions.	
Parameter	Value	Layer	Thickness (μm)
Large coolant channel radius (cm)	0.794	Fuel kernel	200
Small coolant channel radius (cm)	0.635	Buffer layer	120
Fuel compact radius (cm)	0.6225	IPyC	40
Fuel hole radius (cm)	0.6350	SiC	35
Coolant/Fuel cell pitch (cm)	1.88	OPyC	40
Coolant/Fuel cell side (cm)	1.08542		
TRISO packing fraction (%)	17.5016		

Table 1-3. TRISO composition.

(a) Fuel kernel.		(b) Buffers compositions.	
Nuclide	Number density ($.10^{24}$ atoms.cm^{-3})	Nuclide	Number density ($.10^{24}$ atoms.cm^{-3})
^{16}O	$4,436.10^{-2}$	Porous Graphite Buffer Layer	
^{237}Np	$1,522.10^{-3}$	Graphite	$5,265.10^{-2}$
^{238}Pu	$6,464.10^{-4}$	IPyC Layer	
^{239}Pu	$1,099.10^{-2}$	Graphite	$9,526.10^{-2}$
^{240}Pu	$5,084.10^{-3}$	SiC Coating Layer	
^{241}Pu	$1,937.10^{-3}$	^{28}Si	$4,402.10^{-2}$
^{242}Pu	$1,074.10^{-3}$	^{29}Si	$2,235.10^{-3}$
^{241}Am	$6,163.10^{-4}$	^{30}Si	$1,473.10^{-3}$
^{242}Am	$4,385.10^{-6}$	^{12}C	$4,772.10^{-2}$
^{243}Am	$3,057.10^{-4}$	OPyC Layer	
		Graphite	$9,526.10^{-2}$

Table 1-4. Other mixtures.

Nuclide	Number density ($x10^{24}$ atoms.cm^{-3})
Graphite matrix in the fuel pellet for TRISOs	$8.524.10^{-2}$
Graphite surrounding the fuel pellet and the coolant tubes	$8.774.10^{-2}$
^{4}He in the coolant tubes and in an annular tube around the fuel pellet	$7.000.10^{-4}$

1.3   Objectives of the Report

This report investigates the different ways to prepare cross sections for the core calculation. It especially focuses on two parameters:

- The number of groups to which the cross-sections are condensed before the whole core calculation.

- The presence or absence of the blocks surrounding the block of interest at the lattice level.

The code DRAGON Version 4 was used for the lattice level and INSTANT for the whole core calculations. Note that INSTANT is not a diffusion but a 3-D transport code, which uses the PN method.

1.4   Organization of the Report

In this report, neutronics equations are first presented in Section 2. Nothing new has been added in this section compared to text books, but it may be useful for the reader to review the physical equations underlying each numerical method used in the simulations. Section 3 develops with details the different calculation schemes considered—single-block paths versus supercells paths using a larger domain at the lattice level. Details are given about the options used in DRAGON and INSTANT. Section 4 contains all the results to analyze the two main paths and quantify their accuracy. In Section 4.6, a depletion study provides more data for a future complete calculation scheme. It emphasizes the problems raised by this reactor. A final conclusion summarizes the results of this work and suggest some points, which should be further studied. Bibliography and appendixes are provided at the end of this report.

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

March 2011   5

2.   NEUTRONICS EQUATIONS

The neutronics deals with interactions between neutrons and matter in a nuclear reactor. This discipline may also be called reactor physics, and is of high importance for reactor design, operation, and safety. It aims to determine an accurate representation of the neutron distribution inside the core, which leads to the determination of the power shape in the reactor, the reactivity of the core, and the depletion of the different isotopes. In this section, the equations are presented that underlie all neutronics codes, particularly the codes DRAGON and INSTANT, which are used in this work. All of this section is based on the graduate-level course of Alain Hébert at the École Polytechnique de Montréal. An exhaustive presentation of this thema can be found in reference,[2] which inspired this section, but also in Reuss and Massimo's documents,[3,4] this last one being more focused on high-temperature reactors.

Interactions between neutrons and nuclei are described by functions of neutron energy and nuclide characteristics, which are called cross sections. Their establishment results both from quantum mechanical models of the neutron-nuclei interactions and from direct or indirect measuring during experiments. Thus, neutronics only deals with the statistical behavior of neutrons at a mesoscopic level, so that quantum effects vanish. We start with five assumptions, which are well verified[2]:

- Relativistic effects are neglected

- Neutron-neutron interactions are neglected; in fact, the neutron density in the reactor is always much lower than the density of the nuclei

- Neutrons are neutral particles; therefore, neutron mean free paths are straight lines

- The materials are isotropic in space—all considered mixtures are perfectly mixed

- The nuclides are in thermal equilibrium with their neighbors.

The neutron distribution in the core will depend of the position in the reactor (3 variables), the velocity of neutrons (3 variables), and the time (1 variable). This distribution is the solution of the transport equation that models the behavior of neutrons over the core.

2.1   Fundamental Nuclear Parameters

Depending on the energy of the incident neutron, a collision between a neutron and a nucleus can produce several reactions. Two basic phenomena can happen:

- The neutron is scattered by the nucleus without penetrating it, which is called a potential scattering reaction. This reaction is described as elastic because both the momentum and kinetic energy of the neutron-nucleus pair are conserved. A classical analogy is the billiard-ball collision.

- The collision produces a compound nucleus where the incident neutron penetrates the nucleus and mixes with other nucleons. The compound nucleus therefore gains a lot of internal energy and is most of the time highly unstable. After a life-time of between 10^{-22} s to 10^{-14} s, the compound nucleus loses its excitation energy by emitting particles and/or electromagnetic rays. The compound nucleus can undergo several evolutions: fission, particle emission (proton, neutron, alpha particle) and/or gamma rays emission. Notice that if a single neutron is emitted, the reaction is called resonant scattering reaction. The result is comparable to the potential scattering reaction. However, this reaction can be inelastic, which means that the momentum and kinetic energy are not conserved. In this case, the reaction comes with gamma ray emission. If the compound nucleus is only emitting gamma rays, then the reaction is called radiative capture.[2]

In neutronics, we are only interested in the effect of the reaction upon the neutron population. Therefore, we introduce cross-sections and collision laws.

Cross sections are related to the probability that each nuclear reaction has to occur. They lead to the calculation of corresponding reaction rates, that is to say the number of nuclear reactions of this type per unit of time.

The collision laws describe the dynamics of a collision, which leads to the calculation of the velocity and direction characteristics of the emitted particle. All reactions involving the formation of a compound nucleus are usually meant to be isotropic in the laboratory reference system (LAB). Collision laws for potential scattering reactions are usually simplified by using the approximation of isotropic scattering or of linear anisotropic scattering. Those approximations will be detailed while introducing the neutron scattering source term.

2.1.1   Definition of the Cross Sections

Cross sections describe the probability of each type of nuclear reaction and are based on a fundamental property of nuclear reactions[2]:

> *The probability for a neutron located at $\vec{r}$ and moving in a material at velocity*
> *Vn to undergo a nuclear reaction in a differential element of trajectory ds is*
> *independent of the past history of the neutron and is proportional to ds.*

For example, consider a monokinetic and monodirectional beam of neutron hitting perpendicularly a target of width ds and area S at a velocity V_n. I is the intensity of the beam, that is to say the number of neutrons hitting the target per unit of area per unit of time. Mathematically, this intensity is defined by:

$$I = nV_R \tag{2-1}$$

where V_R is the relative velocity of the neutrons with respect to the target, and n is the volumetric concentration of the neutrons (neutrons.cm^{-3}) in the beam. To simplify, consider the target nuclei at rest (0 K); then, V_R is equal to V_n.

The volumetric concentration of nuclei in the target is given by:

$$N = \frac{\rho A_0}{M} \tag{2-2}$$

where ρ is the mass density of nuclides (g.cm^{-3}), M is the atomic mass of a nuclide (g.mol^{-1}) and A_0 is the Avogadro number defined as $6.022094.10^{23}$ atoms.mol^{-1}.

The surfacic reaction rate dR_x is defined as the number of nuclear reactions of type x per unit of time and per unit of area of the target. Experiment has shown that this reaction rate can be expressed by:

$$dR_x = \sigma_x NI \, ds \tag{2-3}$$

The microscopic cross section σ_x is a proportionality factor. Quantum mechanics is now able to provide a more complete definition of the microscopic cross section so that it is possible to explain part of their behavior[5]; however, this is not the purpose of this report.

A dimensional analysis of Equation 2-3 shows that the microscopic cross section must have the dimension of an area. It is generally expressed in barns (*b*), with *1b = 10^{-24} cm^2*. In a classical analogy, imagine the microscopic cross section as the surface on which a neutron heading toward the nuclide will interact with it. Of course, the concept of area at those scales is not really relevant and can be misleading, but it gives a helpful representation of the phenomenon. Some nuclides will have a high microscopic cross section, which reflects a high affinity of the neutrons with those nuclides. On the contrary, other nuclides will not react as readily with neutrons, and therefore have a small microscopic cross section.

It can be useful to group all the characteristics of the target into a single value. Therefore, we define the macroscopic cross section Σ_x:

$$\Sigma_x = N\sigma_x \tag{2-4}$$

Due to its definition, the macroscopic cross section has the dimension of the inverse of a length. It is usually expressed in cm^{-1}.

The microscopic and macroscopic cross sections are additive. For example, if the material of the target is a homogeneous mixture of different types i of nuclides, the resulting macroscopic cross section is:

$$\Sigma_x = \sum_i N_i \sigma_{x,i} \tag{2-5}$$

Nuclear reactions are independent from each other. This allows us to define the total macroscopic cross section as the sum of the cross sections from all nuclear reactions:

$$\Sigma = \sum_x \Sigma_x \tag{2-6}$$

The mean free path λ of neutrons is defined by the average distance that the neutrons can travel in an infinite and homogeneous material before undergoing a reaction.[3] In an infinite homogeneous slab, it is equal to:

$$\lambda = \frac{1}{\Sigma} \tag{2-7}$$

where Σ is the total macroscopic cross section.[2] The mean free path gives a rough estimate of the traveling distance of the neutrons.

Nuclear reactions are usually classified into two categories: scattering reactions and absorption.[2] The scattering cross-section is defined as:

$$\sigma_s = \sigma_e + \sigma_{in} + \sum_{x \geqslant 2} \sigma_{n,xn} \tag{2-8}$$

where σ_e is the elastic (or potential) scattering cross-section, σ_{in} is the inelastic cross-section and $\sigma_{n,xn\ x>2}$ the reactions that emit more than 2 neutrons, but without undergoing a fission.

The absorption cross-section σa is defined as the sum of all other types of reactions: fission reactions (σ_f), radiative capture (σ_γ), and transmutation reactions (σ_α: an α particle is emitted; σ_p: a proton is emitted, etc.).

Except the potential scattering, all nuclear reactions involve the formation of a compound nucleus and their cross sections may exhibit high variations with neutron energy.[2,3] Therefore, all other quantities depending on cross sections like the mean free path will also exhibit high variations with the neutron energy.

Figure 2-1 shows for example the evolution of the total cross section of Pu^{239} with the incident neutron energy. This figure was created using the online cross-section plotter of http://atom.kaeri.re.kr and the JEFF2.2 library.

Three regions are observed: the thermal region at low energies where the cross section is regular, the epithermal region that exhibits several resonances with important variations, and the high energies where resonances are no more resolved (here we are only able to give an average value of the cross sections, but it does not mean that there are no more resonances).
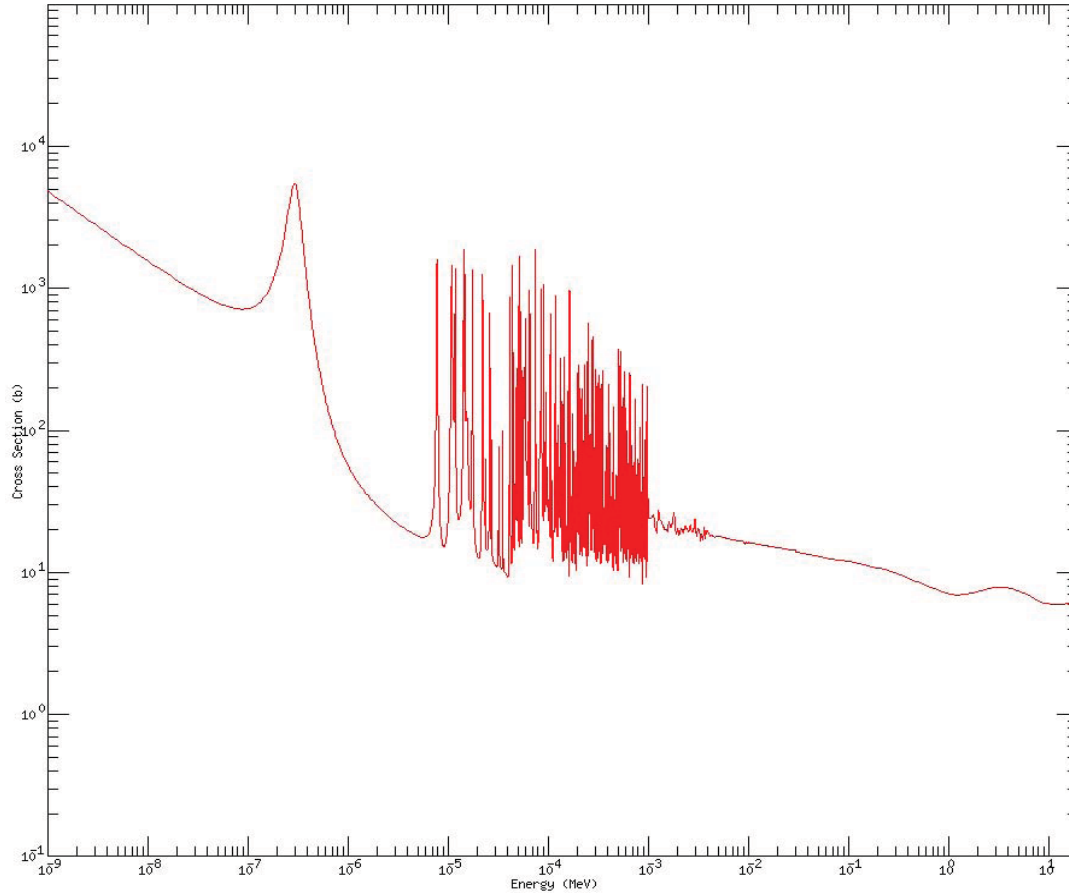
Figure 2-1. Total cross section of Pu^{239} in barn, function of the incident neutron energy (MeV).

2.2   The Particle Flux

In the reactor, each particle is described in Cartesian coordinates by:

- Three position coordinates $\vec{r} = x\vec{\imath} + y\vec{\jmath} + z\vec{k}$

- Three velocity coordinates $\vec{v} = v_x\vec{\imath} + v_y\vec{\jmath} + v_z\vec{k}$ and $\vec{v} = \frac{d\vec{r}}{dt}$. Practically, we use the

  velocity module $V_n = \|\vec{v_n}\| = \sqrt{v_x^2 + v_y^2 + v_z^2}$ and the solid angle $\vec{\Omega}$, which is defined by

  $\vec{\Omega} = \frac{1}{\|\vec{v_n}\|} \cdot \vec{v} = \mu\vec{\imath} + \eta\vec{\jmath} + \epsilon\vec{k}$ and represents the unit vector of direction of movement. This

  normalization leads to the fact that only two components of $\vec{\Omega}$ are independent: for example, the
  colatitude or *polar angle* ψ and the azimuth ϕ which determines all componets using:

$$\begin{cases} \mu^2 + \eta^2 + \epsilon^2 = 1 \\ \mu = \cos\psi \\ \eta = \cos\phi \\ \epsilon = \sin\phi \end{cases} \Longleftrightarrow \begin{cases} \eta^2 + \epsilon^2 = 1 - \mu^2 = \sin^2\psi \\ \mu = \cos\psi \\ \eta = \sqrt{1 - \mu^2}\cos\phi \\ \epsilon = \sqrt{1 - \mu^2}\sin\phi \end{cases} \qquad (2\text{-}9)$$

Consider an elemental volume d^3r. A population of particles is represented by a distribution called the
population density $n(\vec{r}, V_n, \vec{\Omega}, t)$ such that $n(\vec{r}, V_n, \vec{\Omega}, t)\, d^3r\, dV_n d^2\Omega$ is the number of particles at time t,

in the volume element $\mathrm{d}^3 r$ surrounding position $\vec{r}$, in the velocity element $\mathrm{d}V_n$ surrounding V_n and in the solid angle element $\mathrm{d}^2\Omega$ surrounding $\vec{\Omega}$.

We define the *angular flux* as a distribution related to the population density and its velocity:

$$\phi(\vec{r}, V_n, \vec{\Omega}, t) = n(\vec{r}, V_n, \vec{\Omega}, t) \cdot V_n \tag{2-10}$$

We may want to sum the flux on all angular directions. By performing a distribution reduction on variable $\vec{\Omega}$, we define the *integrated flux* $\phi(\vec{r}, V_n, t)$:

$$\phi(\vec{r}, V_n, t) = \int_{4\pi} \phi(\vec{r}, V_n, \vec{\Omega}, t)\, \mathrm{d}^2\Omega \tag{2-11}$$

To spare notations, we use the same symbol to represent the angular flux and the integrated flux. The presence or absence of $\vec{\Omega}$ in the arguments will tell us which quantity we are considering. The same convention was adopted in Hébert's 2009 article.[2]

We commonly use the following change of variable for V_n:

$$\mathrm{E} = \frac{1}{2} m V_n^2 \ \ with \ \ \mathrm{d}E = m V_n \, dV_n$$

$$u = 1n\left(\frac{E_0}{E}\right) \ \ with \ \ \mathrm{d}u = \left| \frac{E}{E_0} E_0 \left(-\frac{1}{E^2}\right) \right| \mathrm{d}E = \frac{1}{E}\, \mathrm{d}E \tag{2-12}$$

where E is the energy of the particle, m its mass and V_n the norm of its velocity vector; u is a quantity called the lethargy, with E_0 the maximum energy of a particle so that u remains positive. It gives:

$$\phi(\vec{r}, E, t) = \frac{1}{m V_n} \phi(\vec{r}, V_n, t), \qquad E_0 \geqslant E > 0$$

$$\phi(\vec{r}, u, t) = E\phi(\vec{r}, E, t), u \geqslant 0. \tag{2-13}$$

The angular flux and the integrated flux do not have a precise physical meaning.[2] In fact, the physical quantity that makes sense is the angular current of Particle $\vec{J}$. It represents the number $\frac{\mathrm{d}N}{\mathrm{d}t}$ of particles of Velocity V_n passing through an Elemental Surface $\mathrm{d}^2 S$ per unit of time.[3]

$$\frac{\mathrm{d}N}{\mathrm{d}t} = \underbrace{n(\vec{r}, V_n, \vec{\Omega}, t) V_n \vec{\Omega}}_{\vec{J}(\vec{r}, V_n, \vec{\Omega}, t)} \cdot \vec{N} \, \mathrm{d}^2 S$$

$$\Rightarrow \vec{J}(\vec{r}, V_n, \vec{\Omega}, t) = \phi(\vec{r}, V_n, \vec{\Omega}, t)\, \vec{\Omega} \tag{2-14}$$

We can define the integrated current:

$$\vec{J}(\vec{r}, V_n, t) = \int_{4\pi} \vec{J}(\vec{r}, V_n, \vec{\Omega}, t)\, \mathrm{d}^2\Omega = \int_{4\pi} \phi(\vec{r}, V_n, \vec{\Omega}, t)\, \vec{\Omega}\, \mathrm{d}^2\Omega \tag{2-15}$$

2.3 The Transport Equation

The transport equation is the mathematical translation of the principle of conservation of the particles.

2.3.1 The Differential Form of the Transport Equation

Let us define a *control volume* $\mathcal{V}$ surrounded by a *controlled surface* $\partial\mathcal{V}$. We consider particles located in $\mathcal{V}$, traveling in direction $\vec{\Omega}$ within a $\mathrm{d}^2\Omega$ interval, with a velocity equal to $\mathcal{V}_n$ within a $\mathrm{d}\mathcal{V}_n$ interval. The initial number of particle is equal to:

$$\int_{\mathcal{v}} n(\vec{r}, V_n, \vec{\Omega}, t)\, \mathrm{d}V_n \mathrm{d}^2\Omega \mathrm{d}^3 r. \tag{2-16}$$

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

March 2011   11

This number is going to change during an infinitesimal time dt according to following balance relation:

(Number of particles at t + dt in $\mathcal{V}$) – (Number of particles at t in $\mathcal{V}$)

= (Number of particle created in $\mathcal{V}$) – (Number of particles lost by collision in $\mathcal{V}$)

   + (Particles entering $\mathcal{V}$ through $\partial\mathcal{V}$) – (Particles come out of $\mathcal{V}$ through $\partial\mathcal{V}$).   (2-17)

   Mathematically, the terms of the previous equation are written:

- The variation of the particle number:

$$\mathrm{d}^3\mathcal{A} = \int_v \left[n(\vec{r}, V_n, \vec{\Omega}, t + \mathrm{d}t) - n(\vec{r}, V_n, \vec{\Omega}, t) \right] \mathrm{d}V_n \mathrm{d}^2\Omega \mathrm{d}^3 r \qquad (2\text{-}18)$$

- The net number of particles streaming out of $\mathcal{V}$ during dt:

$$\mathrm{d}^3\mathcal{B} = \int_{\partial v} \phi\left(\vec{r}, V_n, \vec{\Omega}, t\right) \vec{\Omega} \cdot \vec{N} \, \mathrm{d}V_n \, \mathrm{d}^2\Omega \, \mathrm{d}t \, \mathrm{d}^2 r \qquad (2\text{-}19)$$

where $\vec{N}$ is the unit vector normal to the surface $\partial\mathcal{V}$, pointing outside $\partial\mathcal{V}$. The surface $\partial\mathcal{V}$ is located at position $\vec{r}$. A negative value would mean that in reality, more particles are coming in than streaming out of $\mathcal{V}$. With the divergence theorem,[6] this expression is equal to:

$$\mathrm{d}^3\mathcal{B} = \int_v \, div\left[\phi(\vec{r}, V_n, \vec{\Omega}, t)\vec{\Omega}\right] \mathrm{d}V_n \, \mathrm{d}^2\Omega \, \mathrm{d}t \, \mathrm{d}^3 r \qquad (2\text{-}20)$$

$$= \int_v \vec{\nabla} \cdot \left[\phi(\vec{r}, V_n, \vec{\Omega}, t) \, \vec{\Omega}\right] \mathrm{d}V_n \, \mathrm{d}^2\Omega \, \mathrm{d}t \, \mathrm{d}^3 r \qquad (2\text{-}21)$$

- The number of collisions in d^{3}r during dt is equal to:

$$\mathrm{d}^3\mathcal{C} = \int_v \Sigma(\vec{r}, V_n) \left[n(\vec{r}, V_n, \vec{\Omega}, t) \cdot V_n \right] \mathrm{d}V_n \, \mathrm{d}^2\Omega \, \mathrm{d}t \, \mathrm{d}^3 r$$

$$= \int_v \Sigma(\vec{r}, V_n) \left[\phi(\vec{r}, V_n, \vec{\Omega}, t) \right] \mathrm{d}V_n \, \mathrm{d}^2\Omega \, \mathrm{d}t \, \mathrm{d}^3 r \qquad (2\text{-}22)$$

where we assume that the total macroscopic cross section Σ is independent of $\vec{\Omega}$ and t. This hypothesis simplifies the notations, but it is usually not valid. When a collision occurs, we consider that the particle will always gain or lose some speed, or take another direction. Therefore, it vanishes from our integration element because it streams out of dV_n.

- The number of new particles created in $\mathcal{V}$ during dt is equal to:

$$\mathrm{d}^3\mathcal{D} = \int_v Q(\vec{r}, V_n, \vec{\Omega}, t)\mathrm{d}V_n \, \mathrm{d}^2\Omega \, \mathrm{d}t \, \mathrm{d}^3 r \qquad (2\text{-}23)$$

where Q is the distribution representing the source of neutrons, which we will define more precisely later.

The transport equation is then written:

$$\mathrm{d}^3\mathcal{A} = -\mathrm{d}^3\mathcal{B} - \mathrm{d}^3\mathcal{C} + \mathrm{d}^3\mathcal{D} \qquad (2\text{-}24)$$

Replacing the terms by their expressions, we obtain:

$$\int_v \left[n(\vec{r}, V_n, \vec{\Omega}, t + \mathrm{d}t) - n(\vec{r}, V_n, \vec{\Omega}, t) \right] \mathrm{d}V_n \, \mathrm{d}^2\Omega \, \mathrm{d}^3 r$$

$$= -\int_v \vec{\nabla} \cdot \left[\phi(\vec{r}, V_n, \vec{\Omega}, t)\vec{\Omega}\right] \mathrm{d}V_n \, \mathrm{d}^2\Omega \, \mathrm{d}t \, \mathrm{d}^3 r$$

$$- \int_v \Sigma(\vec{r}, V_n) \, \phi(\vec{r}, V_n, \vec{\Omega}, t) \, dV_n \, d^2\Omega \, dt \, d^3r$$

$$+ \int_v Q(\vec{r}, V_n, \vec{\Omega}, t) \, dV_n \, d^2\Omega \, dt \, d^3r \tag{2-25}$$

The transport equation is valid for all control Volumes $\mathcal{V}$, so that we can discard it in the four terms and obtain the differential form of this equation:

$$\left[n(\vec{r}, V_n, \vec{\Omega}, t + dt) - n(\vec{r}, V_n, \vec{\Omega}, t) \right] dV_n \, d^2\Omega$$

$$= -\vec{\nabla} \cdot \left[\phi(\vec{r} \, V_n, \vec{\Omega}, t) \vec{\Omega} \right] dV_n \, d^2\Omega dt$$

$$- \Sigma(\vec{r}, V_n) \, \phi(\vec{r}, V_n, \vec{\Omega}, t) \, dV_n \, d^2\Omega dt$$

$$+ Q(\vec{r}, V_n, \vec{\Omega}, t) \, dV_n d^2\Omega dt \tag{2-26}$$

Finally, we divide by dt, dV_n, $d^2\Omega$ and take the limit as $dt \to 0$ so that we recognize a partial derivation of $n(\vec{r}, V_n, \vec{\Omega}, t)$. We obtain:

$$\frac{\partial n(\vec{r}, V_n, \vec{\Omega}, t)}{\partial t} = -\vec{\nabla} \cdot \left[\phi(\vec{r}, V_n, \vec{\Omega}, t) \vec{\Omega} \right] - \Sigma(\vec{r}, V_n) \, \phi(\vec{r}, V_n, \vec{\Omega}, t) + Q(\vec{r}, V_n, \vec{\Omega}, t) \tag{2-27}$$

We then use a formula to expand the divergence term:

$$\text{div} \left(\phi\vec{\Omega} \right) = \underbrace{\phi \, div \left(\vec{\Omega} \right)}_{=0} + \vec{\Omega} \cdot \overrightarrow{\text{grad}} \left(\phi \right) \tag{2-28}$$

This leads to the final formulation, introducing the flux in the first term:

$$\frac{1}{V_n} \frac{\partial \phi(\vec{r}, V_n, \vec{\Omega}, t)}{\partial t} = -\vec{\Omega} \cdot \vec{\nabla} \left[\phi(\vec{r}, V_n, \vec{\Omega}, t) \right] - \Sigma(\vec{r}, V_n) \, \phi(\vec{r}, V_n, \vec{\Omega}, t) + Q(\vec{r}, V_n, \vec{\Omega}, t) \tag{2-29}$$

Notice that the term V_n does not depend on t, which explains why it can be moved out of the partial derivation.

In steady-state conditions, all partial time derivatives upon t vanish. The equation becomes:

$$\vec{\Omega} \cdot \vec{\nabla} \left[\phi \left(\vec{r}, V_n, \vec{\Omega} \right) \right] + \Sigma(\vec{r}, V_n) \, \phi(\vec{r}, V_n, \vec{\Omega}) = Q(\vec{r}, V_n, \vec{\Omega}) \tag{2-30}$$

This form is suitable for certain methods of resolution such as the P_N or S_N methods. The streaming operator $\vec{\Omega} \cdot \vec{\nabla}$ must be explicitly written depending on the coordinate system and then discretized. However, we can also transform this equation in two other equivalent forms, leading to two other methods of solution.

2.3.2   The Characteristic Form of the Transport Equation

We define a *characteristic* as a straight line of direction $\vec{\Omega}$ corresponding to a neutron trajectory. We then define the curvilinear abscissa s as the distance s of the neutron on a characteristic from a reference position $\vec{r}$ on this characteristic. Therefore, we can operate a change of variables:

$$\begin{cases} \vec{r}(t) = \vec{r}(t_0) + s\vec{\Omega} \\ \quad t = t_0 + \frac{s}{V_n} \end{cases} \Rightarrow \begin{cases} \frac{d}{d_s} = \vec{\Omega} \cdot \vec{\nabla} + \frac{1}{V_n} \frac{\partial}{\partial t} \end{cases} \tag{2-31}$$

The previous implication is relatively easy to demonstrate with Cartesian coordinates:

$$\begin{cases} ds\,\vec{\Omega} = d\vec{r} = dx\vec{\imath} + dy\vec{\jmath} + dz\vec{k} \\ \dfrac{d}{ds} = \dfrac{dx}{ds}\dfrac{\partial}{\partial x} + \dfrac{dy}{ds}\dfrac{\partial}{\partial y} + \dfrac{dz}{ds}\dfrac{\partial}{\partial z} + \dfrac{1}{V_n}\dfrac{\partial}{\partial t} \end{cases}$$

$$\Rightarrow \frac{d}{ds} = (\vec{\Omega}\cdot\vec{\imath})\frac{\partial}{\partial x} + (\vec{\Omega}\cdot\vec{\jmath})\frac{\partial}{\partial y} + (\vec{\Omega}\cdot\vec{k})\frac{\partial}{\partial z} + \frac{1}{V_n}\frac{\partial}{\partial t}$$

$$\Leftrightarrow \frac{d}{ds} = \vec{\Omega}\cdot\vec{\nabla} + \frac{1}{V_n}\frac{\partial}{\partial t} \tag{2-32}$$

It is more complicated with other coordinate systems, because we need to explicitly calculate the Jacobian matrix.

Substituting Equation 2-31 in 2-29 gives the backward characteristic form of the transport equation:

$$\frac{d\phi(\vec{r}+ s\vec{\Omega},V_n,\vec{\Omega},t+ s/v_n)}{ds} + \Sigma(\vec{r}+ s\vec{\Omega},V_n)\,\phi(\vec{r},V_n,\,\vec{\Omega},t+ s/V_n) = Q(\vec{r}+ s\vec{\Omega},V_n,\,\vec{\Omega},t+ s/V_n) \tag{2-33}$$

It can also be written in the forward form as:

$$\frac{d\phi(\vec{r}+ s\vec{\Omega},V_n,\vec{\Omega},t+ s/v_n)}{ds} + \Sigma(\vec{r}- s\vec{\Omega},V_n)\,\phi(\vec{r},V_n,\,\vec{\Omega},t- s/V_n) = Q(\vec{r}- s\vec{\Omega},V_n,\vec{\Omega},t- s/V_n) \tag{2-34}$$

This equation will be solved by the MOC.[7] But we can continue to manipulate this equation to provide an integral form.

2.3.3   The Integral Form of the Transport Equation

Let us define the *optical path* as:

$$\tau(s,V_n) = \int_0^s \Sigma(\vec{r}- s'\,\vec{\Omega},V_n)\,ds' \tag{2-35}$$

We start by computing the following expression:

$$\frac{d}{ds}\left[e^{-\tau(s,V_n)}\times \phi(\vec{r}- s\vec{\Omega},V_n,\vec{\Omega},t- s/V_n)\right]$$

$$= e^{-\tau(s,V_n)}\frac{d}{ds}\left[\phi(\vec{r}- s\vec{\Omega},V_n,\vec{\Omega},t- s/V_n)\right]$$

$$\qquad + \phi(\vec{r}- s\vec{\Omega},V_n,\vec{\Omega},t- s/V_n)\frac{d}{ds}\left[e^{-\tau(s,V_n)}\right]$$

$$= e^{-\tau(s,V_n)}\frac{d}{ds}\left[\phi(\vec{r}- s\vec{\Omega},V_n,\vec{\Omega},t- s/V_n)\right]$$

$$\qquad + \phi(\vec{r}- s\vec{\Omega},V_n,\vec{\Omega},t- s/V_n)\times e^{-\tau(s,V_n)}\left(-\Sigma(\vec{r}- s\vec{\Omega})\right)$$

$$= e^{-\tau(s,V_n)}\left(\frac{d}{ds}\left[\phi(\vec{r}- s\vec{\Omega},V_n,\vec{\Omega},t- s/V_n)\right]\right.$$

$$\qquad \left. + \phi(\vec{r}- s\vec{\Omega},V_n,\vec{\Omega},t- s/V_n)\left(-\Sigma(\vec{r}- s\vec{\Omega})\right)\right) \tag{2-36}$$

where we used the identity:

$$\frac{\mathrm{d}}{\mathrm{d}s} \int_0^s f(s')\, \mathrm{d}s' = f(s). \tag{2-37}$$

We recognize part of the characteristic forward form of the transport equation. Substitution of Equation 2-34 into 2-36 gives:

$$-\frac{\mathrm{d}}{\mathrm{d}s}\left[e^{-\tau(s,V_n)} \times \phi\big(\vec{r} - s\vec{\Omega}, V_n, \vec{\Omega}, t - s/V_n\big)\right] = e^{-\tau(s,V_n)}\left[Q\big(\vec{r} - s\vec{\Omega}, V_n, \vec{\Omega},\ t - s/V_n\big)\right] \tag{2-38}$$

We then perform an integration between 0 and $+\infty$ over ds:

$$-\int_0^{+\infty}\frac{\mathrm{d}}{\mathrm{d}s}\left[e^{-\tau(s,V_n)} \times \phi\big(\vec{r} - s\vec{\Omega}, V_n, \vec{\Omega}, t - s/V_n\big)\right]\mathrm{d}s$$

$$= \int_0^{+\infty} e^{-\tau(s,V_n)}\left[Q\big(\vec{r} - s\vec{\Omega}, V_n, \vec{\Omega}, t - s/V_n\big)\right]\mathrm{d}s \tag{2-39}$$

$$\Leftrightarrow\ \phi\big(\vec{r}, V_n, \vec{\Omega}, t\big) = \int_0^{+\infty} e^{-\tau(s,V_n)}\left[Q\big(\vec{r} - s\vec{\Omega}, V_n, \vec{\Omega}, t - s/V_n\big)\right]\mathrm{d}s \tag{2-40}$$

Equation 2-40 is the integral form of the transport equation for an infinite domain.[2] Physically, it represents the neutron flux created by a source at distance s from a position $\vec{r}$. The exponential term represents the attenuation of the sources with the distance s—the further is the source, the fewer neutrons can reach the position $\vec{r}$. The attenuation coefficient is the total cross section of each mixture laying between the source and the studied position.

This form of the transport equation is the basis for the CP method of resolution.[2]

In the case of a finite domain, the infinite integral must be replaced by a finite one to take into account the fact that the characteristics are finite. We obtain:

$$\phi\big(\vec{r}, V_n, \vec{\Omega}, t\big) = e^{-\tau(b,V_n)}\, \phi\big(\vec{r} - b\vec{\Omega}, V_n, \vec{\Omega}, t - b/V_n\big)$$

$$+ \int_0^b e^{-\tau(s,V_n)}\left[Q\big(\vec{r} - s\vec{\Omega}, V_n, \vec{\Omega}, t - s/V_n\big)\right]\mathrm{d}s \tag{2-41}$$

where b is the maximal distance that a neutron can cover on one characteristic.[2] An integration has then to be performed for all characteristics to get the flux in one position. The term $e^{-\tau(b,V_n)}\, \phi\big(\vec{r} - b\vec{\Omega}, V_n, \vec{\Omega}, t - b/V_n\big)$ can be seen as the boundary flux, and is therefore related to the boundary conditions.

2.4   The Boundary Conditions

Several boundary conditions can be defined depending on our models. Let us call $\partial \mathcal{V}$ the boundary surface of volume $\mathcal{V}$, $\vec{r}_s$ a position on this surface, $\vec{N}(\vec{r}_s)$ the outward normal to this surface at $\vec{r}_s$. Setting a boundary condition means to clarify the behavior of the incoming flux $\phi\big(\vec{r}_s, V_n, \vec{\Omega}, t\big)$ for all incoming directions: $\vec{\Omega} \cdot \vec{N}\,(\vec{r}_s) < 0$.[2] We can use:

- The albedo boundary condition:

$$\phi\big(\vec{r}_s, V_n, \vec{\Omega}, t\big) = \beta\phi\big(\vec{r}_s, V_n, \vec{\Omega}', t\big)\ with\ \vec{\Omega} \cdot \vec{N}\,(\vec{r}_s) < 0 \tag{2-42}$$

where $\vec{\Omega}'$ is the direction of the outgoing particle. $\beta = 1$ corresponds to a reflective condition whereas $\beta = 0$ corresponds to a vacuum condition.

- The specular reflection corresponds to the case where:

$$\vec{\Omega} \cdot \vec{N}(\vec{r}_s) = -\vec{\Omega}' \cdot \vec{N}(\vec{r}_s) \; and \; (\vec{\Omega} \wedge \vec{\Omega}) \cdot \vec{N}(\vec{r}_s) = 0. \tag{2-43}$$

This necessitates a tracking of the geometry where tracks are undergoing a mirror reflection when they reach edges. The implementation is quite difficult and requires a lot of computational resources.

- The white boundary condition is easier to implement; therefore, it is often used because it is a good approximation of the mirror reflection. The white reflective condition considers that all particles reaching the edge turn back to the volume with an isotropic angular distribution: for $\vec{\Omega} \cdot \vec{N}(\vec{r}_s) < 0$ we have:

$$\phi(\vec{r}_s, V_n, \vec{\Omega}, t) = \beta \cdot \frac{\int_{\vec{\Omega} \cdot \vec{N}(\vec{r}_s) > 0} \left[\vec{\Omega}' \cdot \vec{N}(\vec{r}_s) \right] \phi(\vec{r}_s, V_n, \vec{\Omega}', t) \, d^2\Omega'}{\int_{\vec{\Omega} \cdot \vec{N}(\vec{r}_s) > 0} \left[\vec{\Omega}' \cdot \vec{N}(\vec{r}_s) \right] d^2\Omega'}$$

$$\Leftrightarrow \phi(\vec{r}_s, V_n, \vec{\Omega}, t) = \frac{\beta}{\pi} \cdot \int_{\vec{\Omega} \cdot \vec{N}(\vec{r}_s) > 0} \left[\vec{\Omega}' \cdot \vec{N}(\vec{r}_s) \right] \phi(\vec{r}_s, V_n, \vec{\Omega}', t) \, d^2\Omega' \tag{2-44}$$

- The periodic boundary condition expresses the fact that the lattice is infinite and periodic—the flux on one boundary is equal to the flux on another parallel boundary:

$$\phi(\vec{r}_s, V_n, \vec{\Omega}, t) = \phi(\vec{r}_s, + \delta\vec{r}, V_n, \vec{\Omega}, t) \tag{2-45}$$

where $\delta\vec{r}$ is the lattice pitch.

Finally, we can add that inside the domain $\mathcal{V}$, the angular flux $\phi(\vec{r}_s, V_n, \vec{\Omega}, t)$ must be continuous across all internal interfaces in the direction $\vec{\Omega}$ of the moving neutron. On the contrary, continuity is not required along directions that are not parallel to the path of travel.

2.5 The Steady-state Source Density

In a reactor, the sources of neutrons with an energy E are mainly:

- The neutrons created by the fissions
- The neutrons coming from the scattering reactions
- The neutrons created by n, xn reactions.

This section will focus on the term $Q\left(\vec{r}, E, \vec{\Omega}\right)$ of the transport equation. We use the variable of energy E instead of V_n in the equations. As this study is based on steady-state conditions, the time dependence of the equations is not considered. More information about transient analysis can be found in Hébert's 2009 article and Ronzon's 1992 article.[2,8]

2.5.1 The Scattering Source

The scattering reactions can be considered as a source of neutrons; indeed, an incident neutron of energy E' which is undergoing such a reaction can produce a secondary neutron with an energy E. We call this source $Q^{scat}\left(\vec{r}, E, \vec{\Omega}\right)$ and we write:

$$Q^{scat}\left(\vec{r}, E, \vec{\Omega}\right) = \int_{4\pi} \int_0^{+\infty} \Sigma_s\left(\vec{r}, E \leftarrow E', \vec{\Omega} \leftarrow \vec{\Omega}'\right) \phi\left(\vec{r}, E', \vec{\Omega}'\right) dE' d^2\Omega' \tag{2-46}$$

where $\Sigma_s\left(\vec{r}, E \leftarrow E', \vec{\Omega} \leftarrow \vec{\Omega}'\right)$ is the macroscopic differential scattering cross section taking into account diffusion and (n, xn) reactions. It means that for each type of reaction, we have to know the

probability of a neutron having a collision at a certain energy E' with a collision direction $\vec{\Omega}'$, and exiting at energy E in the direction $\vec{\Omega}$.

Most media can be considered as isotropic, because the neutron mass is by far inferior to the mass of the collided nuclei. A well-known exception is water,[3] which contains hydrogen nuclei whose mass is close to that of a neutron. But this assumption is quite reasonable for graphite reflectors[4] and for the fuel mixtures. In this case, the scattering cross section is only a function of the scattering angle:

$$\Sigma_s\left(\vec{r}, E \leftarrow E', \vec{\Omega} \leftarrow \vec{\Omega}'\right) = \frac{1}{2\pi}\Sigma_s\left(\vec{r}, E \leftarrow E', \vec{\Omega} \cdot \vec{\Omega}'\right) \tag{2-47}$$

It is then convenient to write an expansion with Legendre polynomials:

$$\Sigma_s\left(\vec{r}, E \leftarrow E', \vec{\Omega} \cdot \vec{\Omega}'\right) = \sum_{l=0}^{L} \frac{2l+1}{2} \Sigma_{s,l}(\vec{r}, E \leftarrow E')\, P_l\left(\vec{\Omega} \cdot \vec{\Omega}'\right) \tag{2-48}$$

where L is the scattering order of the medium considered. $L = 0$ and $L = 1$ correspond to isotropic scattering and to linearly anisotropic scattering in the LAB, respectively.[2] The Legendre coefficients $\Sigma_{s,l}(E \leftarrow E',)$ are defined as:

$$\Sigma_{s,l}(E \leftarrow E',) = \int_{-1}^{1} \Sigma_s(E \leftarrow E', \mu)\, P_l(\mu)\mathrm{d}\mu \tag{2-49}$$

We can also expand the flux with spherical harmonics to approximate the integral over $\mathrm{d}^2\Omega$, so that we obtain[2]:

$$Q^{\mathrm{scat}}\left(\vec{r}, E, \vec{\Omega}\right) = \int_0^{+\infty} \sum_{l=0}^{L} \frac{2l+1}{4\pi} \Sigma_{s,l}(\vec{r}, E \leftarrow E') \sum_{m=-l}^{l} R_l^m\left(\vec{\Omega}\right) \phi_l^m(\vec{r}, E')\, \mathrm{d}E' \tag{2-50}$$

where

$$\phi_l^m(\vec{r}, E') = \int_{4\pi} R_l^m\left(\vec{\Omega}\right) \phi\left(\vec{r}, E, \vec{\Omega}\right) \mathrm{d}^2\Omega. \tag{2-51}$$

2.5.2   The Fission Source

The fission source is due to the fission reactions, which produce some secondary neutrons. It is usually isotropic in the LAB because they involve a compound nucleus, which disintegrates much later after its formation. Therefore, it does not conserve any memory of the incoming direction of the neutron.[2]

Thus, we write:

$$Q^{fiss}\left(\vec{r}, E, \vec{\Omega}\right) = \frac{1}{4\pi K_{eff}} Q^{fiss}(\vec{r}, E) \tag{2-52}$$

K_{eff} is called the effective multiplication factor. It enables a steady-state calculation even if the reactor is not at the equilibrium, which would mean that the sum of absorption and leakage equals the production rate of new fission neutrons.[2] The K_{eff} adjusts the fission source so that we go back to steady-state conditions. If $K_{eff} > 1$ the reactor is supercritical: the production rate of neutrons is higher than the loss rate of neutrons, and the population of neutrons is increasing. On the contrary, when $K_{eff} < 1$, losses are more important than production of neutrons, and the population is decreasing. The reactor is said to be subcritical. With $K_{eff} = 1$, equilibrium is reached: the reactor is critical. Therefore, the determination of this constant is very interesting in the reactor analysis for safety, operation, and design.[3]

We assume that the isotropic fission source is independent of the energy of the incident neutron. However, the emitted neutrons do not have always the same energy. This information is given by the fission spectrum $\chi_i(E)$, different for each fissile nuclide i. $\chi_i(E)$ is the probability for an emitted neutron to have an energy equal to E within a $\mathrm{d}E$ interval in the LAB.[2] As it is a probability, it is normalized to 1:

$$\int_0^{+\infty} \chi_i E \, dE = 1 \tag{2-53}$$

Thus, the isotropic fission source is written:

$$Q^{fiss}(\vec{r}, E) = \sum_{j=1}^{J^{fiss}} \chi_j(E) \int_0^{+\infty} \nu \Sigma_{f,j}(\vec{r}, E') \, \phi(\vec{r}, E') \, dE' \tag{2-54}$$

where J^{fiss} is the total number of fissile isotopes; ν is the number of emitted neutrons per fission, taken as constant here; and $\Sigma_{f,j}(\vec{r}, E)$ is the macroscopic fission cross section of the jth fissile isotope.

2.5.3 Complete Source Expression

When we add the two source terms, we obtain[2]:

$$Q(\vec{r}, E, \vec{\Omega}) = \int_0^{+\infty} \sum_{l=0}^{L} \frac{2l+1}{4\pi} \Sigma_{s,l}(\vec{r}, E \leftarrow E') \sum_{m=-1}^{l} R_l^m(\vec{\Omega}) \, \phi_l^m(\vec{r}, E') \, dE' \tag{2-55}$$

$$+ \frac{1}{4\pi K_{eff}} \sum_{j=1}^{J^{fiss}} \chi_j(E) \int_0^{+\infty} \nu \Sigma_{f,j}(\vec{r}, E') \, \phi(\vec{r}, E') \, dE'$$

We can remark that:

$$\phi(\vec{r}, E) = \phi_0^0(\vec{r}, E). \tag{2-56}$$

The substitution of this term in the transport Equation 2-30 leads to the singular form of the transport equation. As stated in Reference 2 it is an eigenvalue problem:

- The neutron flux appears in each term. $\phi = 0$ is a trivial solution. A set of nontrivial solutions (eigenvectors or eigensolutions) exists for some discrete values of K_{eff}. The possible K_{eff} values are the eigenvalues and are associated with eigenvectors ϕ. The fundamental solution corresponds to the maximum possible value of K_{eff} and is the only eigensolution with a physical meaning.[2]

- The flux distribution corresponding to the fundamental solution is positive or null everywhere. All other solutions are harmonics and become strictly negative in some regions. A linear combination of the fundamental eigensolution and of the different harmonics can lead to a solution that is positive or null everywhere, which would have a physical meaning. However, we always use the approximation that the flux is well represented by the fundamental solution, without corrections by the harmonics.[2]

- The flux distribution of each eigensolution can be arbitrarily normalized. The value of the normalization constant is usually computed from the thermal power P of the reactor that is measured:

$$P = \int_0^{+\infty} \int_v H(\vec{r}, E) \, \phi(\vec{r}, E) \, d^3r \, dE. \tag{2-57}$$

where $\mathcal{V}$ is the volume of the reactor and $H(\vec{r}, E)$ is the power factor giving the recoverable energy in terms of the flux.

We recall here that in theory, from a neutronics point of view, any power can be achieved by any reactor if it is able to become supercritical. The neutronic limitation only occurs, for example, if the increase of the temperature changes the cross sections and provides a negative feed-back strong enough to decrease K_{eff} under 1. But the power does not depend directly on the resolution of the steady-state transport equation.

2.6 The Transport Correction

Solving the transport equation with linear anisotropy in the scattering source is difficult. The method of collision probabilities shown later is not even able to deal with anisotropic sources. Therefore, it has been imagined to "trick" the cross sections, so that a calculation in isotropic condition would take into account some linear anisotropic effects. The principle is to add a forward-peaked component in the Legendre

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor
Cross Section Generation
18   March 2011

expansion of the differential scattering cross sections.[2] This special treatment of the cross sections is not strictly mathematically correct; however, it is widely used. This additional component takes the form of a Dirac delta term:

$$\Sigma_s(\vec{r}, E \leftarrow E', \mu) = \sum_{l=0}^{L} \frac{2l+1}{2} \bar{\Sigma}_{s,l}(\vec{r}, E \leftarrow E') P_l(\mu) + \Delta\Sigma_{tr}(\vec{r}, E \leftarrow E')\delta(\mu - 1) \tag{2-58}$$

where $\bar{\Sigma}_{s,l}(\vec{r}, E \leftarrow E') P_l(\mu)$ is a modified Legendre coefficient and $\Delta\Sigma_{tr}(\vec{r}, E \leftarrow E')$ is the additional coefficient multiplying the Dirac delta term. They are computed so as to preserve the Legendre moments:

$$\forall l \in [\![0; L + 1]\!], \ \Sigma_{s,l}(\vec{r}, E \leftarrow E') = \int_{-1}^{1} \Sigma_s(\vec{r}, E \leftarrow E', \mu) P_l(\mu) \, d\mu. \tag{2-59}$$

Using $P_l(1) = 1$, we obtain:

$$\begin{cases} \forall l \in [\![0; L]\!], \ \bar{\Sigma}_{s,l}(\vec{r}, E \leftarrow E') + \Delta\Sigma_{tr}(\vec{r}, E' \leftarrow E) = \Sigma_{s,l}\left(E \leftarrow E'\right) \\ \Delta\Sigma_{tr}(\vec{r}, E' \leftarrow E) = \Sigma_{s,l}\left(E \leftarrow E'\right) \end{cases} \tag{2-60}$$

Then, writing $L = 0$, we have:

$$\Sigma_s\left(\vec{r}, E \leftarrow E', \mu\right) = \frac{1}{2}\left[\Sigma_{s,0}(\vec{r}, E \leftarrow E') - \Sigma_{s,1}(\vec{r}, E \leftarrow E')\right] + \Sigma_{s,1}(\vec{r}, E \leftarrow E')\delta(\mu - 1) \tag{2-61}$$

We then substitute in the transport Equation 2-30:

$$\vec{\Omega} \cdot \vec{\nabla} \phi(\vec{r}, E, \vec{\Omega}) + \Sigma(\vec{r}, E, \vec{\Omega}) \phi(\vec{r}, E, \vec{\Omega}) - \int_0^{+\infty} \Sigma_{s,1}(\vec{r}, E \leftarrow E', \vec{\Omega}) \phi(\vec{r}, E', \vec{\Omega}) = Q(\vec{r}, E', \vec{\Omega}) \tag{2-62}$$

where the transport-corrected steady-state source is now written:

$$\bar{Q}(\vec{r}, E', \vec{\Omega}) = \frac{1}{4\pi} \int_0^{+\infty} \left[\Sigma_{s,0}(\vec{r}, E \leftarrow E') - \Sigma_{s,1}(\vec{r}, E \leftarrow E')\right] \phi(\vec{r}, E') \, dE'$$

$$+ \frac{1}{4\pi K_{eff}} \sum_{j=1}^{J^{fiss}} \chi_j(E) \int_0^{+\infty} \nu\Sigma_{f,j}(\vec{r}, E')\phi(\vec{r}, E') \, dE' \tag{2-63}$$

Another approximation is required to simplify the left-hand side of the equation. In DRAGON, we have always been using the micro-reversibility approximation, which is usually valid in the thermal equilibrium energy domain where all neutrons are in quasi equilibrium with the nuclei.[2] It is written:

$$\Sigma_{s,1}(\vec{r}, E \leftarrow E')\phi(\vec{r}, E', \vec{\Omega}) = \Sigma_{s,1}(\vec{r}, E' \leftarrow E)\phi(\vec{r}, E, \vec{\Omega}) \tag{2-64}$$

We do this approximation over the complete energy spectrum, even if it is quite optimistic. It leads to:

$$\vec{\Omega} \cdot \vec{\nabla} \phi(\vec{r}, E, \vec{\Omega}) + \bar{\Sigma}(\vec{r}, E) \phi(\vec{r}, E, \vec{\Omega}) = \bar{Q}(\vec{r}, E, \vec{\Omega}) \tag{2-65}$$

with:

$$\begin{cases} \bar{\Sigma}(\vec{r}, E) = \Sigma(\vec{r}, E) - \Delta\Sigma_{tr}(\vec{r}, E) \\ \Delta\Sigma_{tr}(\vec{r}, E) = \int_0^{+\infty} \Sigma_{s,1}(\vec{r}, E' \leftarrow E) \, dE' \end{cases} \tag{2-66}$$

The source $\bar{Q}(\vec{r}, E, \vec{\Omega})$ can also be written with a transport correction:

$$\bar{Q}(\vec{r}, E, \vec{\Omega}) = \frac{1}{4\pi} \int_0^{+\infty} \left[\Sigma_{s,0}(\vec{r}, E \leftarrow E')\right] \phi(\vec{r}, E') \, dE'$$

$$+ \frac{1}{4\pi K_{eff}} \sum_{j=1}^{J^{fiss}} \chi_j(E) \int_0^{+\infty} \nu\Sigma_{f,j}(\vec{r}, E')\phi(\vec{r}, E') \, dE' \tag{2-67}$$

with:

$$\bar{\Sigma}_{s,0}(\vec{r}, E \leftarrow E') = \Sigma_{s,0}(\vec{r}, E \leftarrow E') - \delta\big(E' - E\big)\Delta\Sigma_{tr}(\vec{r}, E). \tag{2-68}$$

2.7 The Multi-group Steady-state Transport Equation

The variation of the cross sections as a function of the energy of the incoming neutron is analytically not known. Therefore, it is necessary to discretize the energy domain in intervals in which we will consider that the cross sections are constant and independent of energy.[3] Those intervals are called groups. They can be of various widths. The more groups we take, the better we will be able to approach the real behavior of the cross sections. However, a compromise has to be struck between the accuracy of this representation and the computational limitations (time, memory). Optimizing the number of groups for a calculation and the position of the boundaries is a huge task in itself, outside the scope of this work, and several propositions can be made. Usually, deterministic codes use between 150 and 400 groups for the lattice calculations, and less than 50 groups for the whole core calculations.[2] An energy condensation is performed between the two calculations to reduce the number of groups. We also assume a separation of the energy dependence from the space and angular dependence within each group and within the region of interest.

We write below the transport equation in its three forms with G groups:

$$\forall g \epsilon [\![1; G]\!],$$
$$\vec{\Omega} \cdot \vec{\nabla}\big[\phi_g(\vec{r}, \vec{\Omega})\big] + \Sigma_g(\vec{r})\phi_g(\vec{r}, \vec{\Omega}) = Q_g(\vec{r}, \vec{\Omega}) \tag{2-69}$$

$$\Leftrightarrow \frac{d\phi_g(\vec{r}, +s\vec{\Omega}, \Omega)}{d_s} + \Sigma_g(\vec{r} + s\vec{\Omega})\phi_g(\vec{r}, \vec{\Omega}) = Q_g(\vec{r} + s\vec{\Omega}, \Omega) \tag{2-70}$$

$$\Leftrightarrow \phi_g(\vec{r}, \vec{\Omega}) = \int_0^{+\infty} e^{-\tau_g(s)}\big[Q_g(\vec{r} - s\vec{\Omega}, \vec{\Omega})\big]\,ds \tag{2-71}$$

with the optical path in group g:

$$\tau_g(s) = \int_0^s \Sigma_g(\vec{r} - s'\vec{\Omega})\,ds' \tag{2-72}$$

and the source density which couples the groups between themselves through the terms:
$$\Sigma_{s,l,g\leftarrow h}(\vec{r}) \; and \; (\chi_{j,g}):$$

$$Q_g(\vec{r}, \vec{\Omega}) = \sum_{h=1}^{G}\sum_{l=0}^{L}\frac{2l+1}{4\pi}\Sigma_{s,l,g\leftarrow h}(\vec{r})\sum_{m=-l}^{l}R_l^m(\vec{\Omega})\phi_{l,h}^m(\vec{r}) \tag{2-73}$$
$$+ \frac{1}{4\pi K_{eff}}\sum_{j=1}^{J^{fiss}}\chi_{j,g}\sum_{h=1}^{G}\nu\Sigma_{f,j,h}(\vec{r})\phi_h(\vec{r})$$

Performing a transport correction is also possible and leads to:

$$\forall g \in [\![1; G]\!],$$

$$\vec{\Omega} \cdot \vec{\nabla}\big[\phi_g(\vec{r}, \vec{\Omega})\big] + \bar{\Sigma}_g(\vec{r})\phi_g(\vec{r}, \vec{\Omega}) = Q_g(\vec{r}, \vec{\Omega}) - \frac{1}{4\pi}\Delta\Sigma_{tr,g}(\vec{r})\phi_g(\vec{r}) \tag{2-74}$$

2.8 The Collision Probability Method

The collision probability method is an old method of resolution of the transport equation, which is still used because of its relative robustness and speed.[2]

We start with the integral form of the transport equation, discretized in G energy groups:

$$\forall g \in [\![1; G]\!],$$

$$\phi_g(\vec{r}, \vec{\Omega}) = \int_0^{+\infty} e^{-\tau_g(s)}\big[Q_g(\vec{r} - s\vec{\Omega}, \vec{\Omega})\big]\,ds \tag{2-75}$$

We consider a lattice constituted by an assembly repeated to infinity with I regions where the materials are homogeneous. Each region will be referred to as a volume V_i. V_i^∞ refers to the infinite set of the same type of Regions V_i that are repeating themselves in all the assemblies.

The collision probabilities method assumes isotropic sources. The idea consists of discretizing the integral in each I region and producing a matrix of $I \times I$ size in each energy group.

$$\phi_g(\vec{r}) = \int_{4\pi} \phi_g(\vec{r}, \vec{\Omega}) \, d^2\Omega = \frac{1}{4\pi} \int_{4\pi} \int_0^{+\infty} e^{-\tau_g(s)} Q_g(\vec{r} - s\vec{\Omega}) \, ds \, d^2\Omega$$

$$\Leftrightarrow \quad \phi_g(\vec{r}) = \frac{1}{4\pi} \int_\infty \frac{e^{-\tau_g(s)}}{s^2} Q_g(\vec{r}') d^3r' \quad \text{with a change of variable:}$$

$$\begin{cases} \vec{r} = \vec{r} - s\vec{\Omega} \\ d^3r' = s^2 d^2\Omega ds \\ s = \|\vec{r} - \vec{r}'\| \end{cases}$$

$$\Rightarrow \int_{V_j} \Sigma_g(\vec{r}) \phi_g(\vec{r}) d^3r = \frac{1}{4\pi} \int_{V_j} \Sigma_g(\vec{r}) \sum_i Q_{i,g} \int_{V_i^\infty} \frac{e^{-\tau_g(s)}}{s^2} d^3r' d^3r \tag{2-76}$$

The source is explicitly given by:

$$Q_{i,g} = \sum_h \Sigma_{s0,i,g \leftarrow h} \phi_{i,h} + \frac{1}{K_{eff}} \underbrace{\sum_{j=1}^{J^{fiss}} \chi_{j,g} \sum_h \nu\Sigma_{f,j,h} \phi_{i,h}}_{Q_{i,g}^{fiss}} \tag{2-77}$$

The integration is performed saying that the flux and the cross sections are constant within a Region V_i. Therefore, the calculation is straightforward and leads to the equation[2]:

$$V_j \Sigma_{j,g} \phi_{j,g} = \sum_i Q_{i,g} V_i P_{ij,g} \tag{2-78}$$

where

$$\phi_{j,g} = \frac{1}{V_j} \int_{V_j} \phi_g(\vec{r}) \, d^3r$$

$$\Sigma_{j,g} = \frac{1}{V_j \phi_{j,g}} \int_{V_j} \Sigma_g(\vec{r}) \phi_g(\vec{r}) \, d^3r$$

$$P_{ij,g} = \frac{1}{4\pi V_i} \int_{V_i^\infty} \int_{V_j} \Sigma_g(\vec{r}) \frac{e^{-\tau_g(s)}}{s^2} \, d^3r \, d^3r' \tag{2-79}$$

We see that the terms $P_{ij,g}$ form a matrix (Indexes i and j) for each group. They represent the probability of a neutron being born uniformly and isotropically in any Region V_i of the lattice to undergo its first collision in the Region V_j of a unit cell or assembly. If the total Cross Section $\Sigma_g(\vec{r})$ is constant and equal to $\Sigma_{j,g}$ in Region V_j, we can define the reduced collision probabilities by:

$$P_{ij,g} = \frac{P_{ij,g}}{\Sigma_{j,g}} = \frac{1}{4\pi V_i} \int_{V_i^\infty} \int_{V_j} \frac{e^{-\tau_g(s)}}{s^2} \, d^3r \, d^3r' \tag{2-80}$$

The reduced collision probabilities have two properties:

$$\forall (i,j) \in \mathbb{N}, P_{ij,g} V_i = \quad P_{ji,g} V_{ij} \tag{2-81}$$

$$\forall i \in \mathbb{N}, \sum_j P_{ij,g} \Sigma_{j,g} = 1$$

Thus, Equation 2-78 can be further simplified to:

Thus, Equation 2-78 can be further simplified to:

$$\phi_{i,g} = \sum_j Q_{j,g} P_{ij,g}. \tag{2-82}$$

It can be written in matrix form:

$$\Phi_g = \mathbb{P}_g Q_g \tag{2-83}$$

with Φ_g being the vector containing the fluxes in each region, P_g the matrix of the reduced collision probabilities and Q_g a vector containing the sources in each region $V_i, i \in I$ with I the total number of regions:

$$\vec{\Phi}_g = (\phi_{i,g})_{i \in I} \quad \mathbb{P}_g = (P_{ij,g})_{(i,j) \in I^2} \quad \quad \vec{Q}_g = (Q_{i,g})_{i \in I} \tag{2-84}$$

Further transformations are performed for numerical reasons.

The CP algorithm proceeds in three steps[2]:

1. The geometry is tracked, which means that we draw a sufficient number of characteristics over the entire geometry. Typical tracking parameters are the number of angular directions for the tracks and the density of parallel tracks on a perpendicular segment per centimeter. Each time the characteristic encounters a boundary, another track, or a change of composition, it delimits a segment.

2. The integration of the collision probabilities is performed by assuming that the cross sections and the flux are constant on each segment. Parallelization is possible to compute each matrix in each group, because there are no interactions between them.

3. Finally, the integrated flux is calculated with iterations using Equations 2-82 and 2-77.

In the case of a finite domain, the tracks are cut at the boundaries where we set boundary conditions.

For large domains, an alternative to the traditional CP method consists of considering several uncoupled regions instead of one large coupled region. The calculation is performed in each region separately. The regions are then recoupled together by the knowledge of the interface currents on each edge surrounding each cell.[2] An iterative scheme shall converge to the solution satisfying both inner transport equations and interface coupling currents. Those currents obey a balance equation: the outgoing current of Cell A into Cell B is equal to the incoming current from Cell B to Cell A.

Again, like in the P_N method, the outgoing angular fluxes are expanded on an orthogonal basis of flux functions $\{\psi_i(\vec{\Omega} \cdot \vec{N})\}_{i \in \mathbb{N}}$ with $\vec{\Omega} \cdot \vec{N} > 0$ where $\vec{N}$ is the outgoing normal of the considered cell.

We usually limit ourselves to the Orders 0 or 1. Cutting the expansion at the Order 0 means that we consider an isotropic outgoing current. The first order introduces the possibility to have some anisotropy in the direction of the current, which is much more accurate and usually preferable.

2.9   The Method of Characteristics

This method uses the characteristic form of the transport equation:

$\forall g \in [\![1; G]\!]$,

$$\frac{d\emptyset_g(\vec{r}+ s\vec{\Omega},\vec{\Omega})}{ds} + \Sigma_g(\vec{r} + s\vec{\Omega}) \, \emptyset_g(\vec{r},\vec{\Omega}) = Q_g \emptyset_g(\vec{r} + s\vec{\Omega}, \vec{\Omega}) \tag{2-85}$$

The MOC consists of tracking several characteristics over the domain and following a neutron traveling through an entire characteristic from one side to the other side.[7] On the contrary, the CP method considers a point and counts the number of neutrons able to reach it from the other regions. But the two methods have a similar philosophy, and the same tracking file can be used.[9]

The characteristics are crossing different regions, and therefore are cut into K pieces, K being an integer representing the total number of regions crossed by one characteristic.[7] We consider the average flux on each segment and the average source:

$$\phi_{k,g} = \frac{1}{l_k} \int_0^{l_k} \phi_g(\vec{r}_k + s\vec{\Omega}) \, ds$$

$$Q_{k,g} = \frac{1}{l_k} \int_0^{l_k} Q_g(\vec{r}_k + s\vec{\Omega},\vec{\Omega} \,) ds \tag{2-86}$$

where $\vec{r}_k$ is the point of entry of the characteristic inside the region k. Thus, we have $\vec{r_{k+1}} = \vec{r}_k + l_k\vec{\Omega}$.

We consider a constant total cross section $\Sigma_{k,g}$ and also usually a constant isotropic source on the segment:

$$\forall s \epsilon \, [0, l_k], \quad Q_g(\vec{r}_k + s\vec{\Omega},\vec{\Omega}) = \frac{Q_{k,g}}{4\pi} \tag{2-87}$$

With this approximation known as the step-characteristic approximation,[7] we can integrate analytically the transport equation over one segment:

$$\frac{d\phi_{k,g}(\vec{r}+s\vec{\Omega},\vec{\Omega})}{ds} + \Sigma_{k,g}(\vec{r} + s\vec{\Omega})\phi_{k,g}(\vec{r},\vec{\Omega}) = \frac{Q_{k,g}}{4\pi} l_k$$

$$\Rightarrow \; \exists \, \phi_0 \in \mathbb{R} / \phi_{k,g}(s) = \phi_0 e^{-\Sigma_{k,g}s} + \frac{Q_{k,g}}{4\pi\Sigma_{k,g}} l_k \tag{2-88}$$

With the help of the boundary condition, we find the constant ϕ_0:

$$\phi_{k,g}(s - 0) = \phi_{k,g} = \phi_0 + \frac{Q_{k,g}}{4\pi\Sigma_{k,g}} l_k$$

$$\Leftrightarrow \; \phi_0 = \phi_{k,g} - \frac{Q_{k,g}}{4\pi\Sigma_{k,g}} l_k$$

$$\Rightarrow \; \phi_{k,g}(s) = \left(\phi_{k,g} - \frac{Q_{k,g}}{4\pi\Sigma_{k,g}} l_k\right) e^{\Sigma_{k,g}s} + \frac{Q_{k,g}}{4\pi\Sigma_{k,g}} l_k \tag{2-89}$$

It leads to $\phi_{k+1,g}$ for $s = l_k$:

$$\phi_{k+1,g} = \phi_{k,g} e^{-\Sigma l_k} + \frac{Q_k}{4\pi\Sigma_{k,g}} l_k \left(1 - e^{-\Sigma_{k,g}l_k}\right) \tag{2-90}$$

We can also discretize the derivation operator of the transport equation assuming that l_k is small and replacing the flux by a constant inside the segment 1_k. It leads to another relation[7]:

$$\phi_{k+1,g} - \phi_{k,g} + \Sigma_{k,g} l_k \phi_{k,g} = \frac{Q_{k,g}}{4\pi} \tag{2-91}$$

where $\phi_{k,g}$ is the average of the flux on the segment.

When we replace $\phi_{k+1,g}$ by its expression in function of $\phi_{k,g}$, we obtain:

$$\phi_{k,g} = \frac{\phi_{k,g}}{\Sigma_{k,g}l_k}\left(1 - e^{-\Sigma_{k,g}l_k}\right) + \frac{Q_{k,g}}{4\pi\Sigma_{k,g}}\left(1 - \frac{1 - e^{-\Sigma_{k,g}l_k}}{\Sigma_{k,g}l_k}\right) \tag{2-92}$$

The source term still depends on the flux. Equations 2-91 and 2-92 will have to be solved by iterations, taking care of the fact that the optical path $\tau_{k,g} = \Sigma_{k,g}l_k$ must remain small.

The interest is that it produces matrices of size $N_L \times L$ where N_L is the number of regions and L the number of interfaces. The CP method generates matrices of size $N_L \times N_L$, and for large domains we have $N_L > L$.[7] Therefore, the MOC is often recommended to do calculations over large domains if memory must be spared. But on the contrary, the CP algorithm reads the tracking once and then computes the matrices of collision probabilities, whereas the MOC algorithm needs to read the tracking at each iteration, which decreases the calculation speed. It explains the importance of preconditioning and acceleration methods for MOC to converge within a reasonable time.[7]

2.10 The Discrete Ordinates Method

The S_n method starts with the differential form of the transport equation by discretizing the angular variable $\vec{\Omega}$ into n directions.[4] The letter S stands for straight lines.[2] Each direction is characterized by a direction cosine. Those are chosen to maximize the accuracy of the integration—a quadrature is used to weight the importance of each direction (e.g., a classical Gauss-Legendre quadrature).[2] The method provides a convenient way to discretize the streaming operator $\vec{\Omega} \cdot \vec{\nabla}\phi(\vec{r},\vec{\Omega})$.

This method cannot be explained without a coordinate system to explicitly write the streaming operator. For example, in Cartesian coordinate system, we would have:

$$\vec{\Omega} = \mu\vec{\imath} + \eta\vec{\jmath} + \xi\vec{k} \quad \text{with } \mu^2 + \eta^2 + \xi^2 = 1$$

$$\Longrightarrow \vec{\Omega} \cdot \vec{\nabla} = \mu\frac{\partial}{\partial x} + \eta\frac{\partial}{\partial y} + \xi\frac{\partial}{\partial z} \tag{2-93}$$

For a one-dimensional x-oriented slab geometry, it gives for example[10]:

$$\left[\mu\frac{\partial}{\partial x} + \Sigma(x)\right]\phi(x,\mu) = \sum_{l=0}^{L}\frac{2l+1}{2}Q_l(x)P_l(\mu) \tag{2-94}$$

We see that a discretization of μ leads to a coupled set of equations:

$$\forall n \in [\![1; N]\!], \quad \left[\mu_n\frac{\partial}{\partial x} + \Sigma(x)\right]\phi_n(x,\mu) = \sum_{l=0}^{L}\frac{2l+1}{2}Q_1(x)P_l(\mu_n) \tag{2-95}$$

with the flux $\phi_n(x) = \phi(x,\mu_n)$. This enables us to compute the Legendre moments of the flux, which will be given by:

$$\phi_l(x) = \sum_{n=1}^{N}\omega_n\phi_nP_l(\mu_n) \quad \text{where } \omega_n \text{ is the quadrature value.} \tag{2-96}$$

Each problem needs to be studied separately: first a correct expression of the streaming operator has to be given. Then the S_N discretization can be applied to it. This method is very efficient, but was not available for hexagonal geometries in DRAGON.[9] However, we have been using it on a cylindrical geometry for our reflector calculations.

2.11 Other Elements of a Lattice Code

2.11.1 Neutron Slowing Down and Resonance Self-shielding

The neutrons produced by fission reactions are generally emitted at a high energy. At those energies, their probability to lead a nuclide to undergo a fission is low. They usually undergo scattering reactions until they reach the thermal energy domain, where the fission cross sections of the fissile isotopes are much higher. Up-scattering can only happen in the thermal domain. It describes the fact that a neutron, which has been slowed down below the thermal agitation energy, can regain some energy while encountering a nuclide. This phenomenon is limited to the thermal energies; it cannot take a neutron back into the highest energies.[2] The neutron slowing down is essentially the consequence of collisions with light nuclides, such as water or graphite, which are put intentionally in the reactor to increase the fission probabilities. Such materials are called moderators.[3] The slowing down effect of heavy nuclides is small because the mass difference between a single neutron and heavy nuclides prevents them from being efficient moderators. We can write a simplified version of the transport equation without the fission source called the slowing-down equation, where we separate the two slowing-down terms according to the type of nuclides:

$$\vec{\Omega} \cdot \vec{\nabla} \phi(\vec{r}, u, \vec{\Omega}) + \Sigma(\vec{r}, u) \phi(\vec{r}, u, \vec{\Omega}) = \frac{1}{4\pi} \left[\mathcal{R}^+ \left(\phi\left(\vec{r}, u, \vec{\Omega}\right) \right) + \mathcal{R}^* \left(\phi\left(\vec{r}, u, \vec{\Omega}\right) \right) \right] \tag{2-97}$$

where:

$$\begin{cases} \mathcal{R}^+ \left(\phi(\vec{r}, u, \vec{\Omega}) \right) = \text{slowing} - \text{down operator for light nuclides} \\ \mathcal{R}^* \left(\phi(\vec{r}, u, \vec{\Omega}) \right) = \text{slowing} - \text{down operator for heavy nuclides} \end{cases}$$

$$\Rightarrow \begin{cases} \mathcal{R}^+ \left(\phi(\vec{r}, u, \vec{\Omega}) \right) = \int_0^{+\infty} \Sigma_{s0}^+ (\vec{r}, u \leftarrow u', \vec{\Omega}) \, \phi(\vec{r}, u') \, du' \\ \mathcal{R}^* \left(\phi(\vec{r}, u, \vec{\Omega}) \right) = \int_0^{+\infty} \Sigma_{s0}^* (\vec{r}, u \leftarrow u', \vec{\Omega}) \, \phi(\vec{r}, u') \, du' \end{cases}$$

During the slowing down process, the absorption cross sections of the heavy nuclides stay low on average, but they show narrow, very large peaks at certain energies in the epithermal domain (see Figure 2-1). Those peaks are called resonances. A neutron coming out of a scattering reaction with an energy located right on a resonance will have a very high probability of being absorbed. Therefore, the flux will be depressed within every resonance because most neutrons arriving at those energies through the scattering reactions will be absorbed.[2,3] Fortunately, the resonances are usually much narrower than the lethargy gain of each scattering reaction so that overall, only a small number of neutrons are absorbed and the flux depressions remain small.[3] Paul Reuss describes this phenomenon in his book[3] with an analogy involving kangaroos—the lethargy interval gained by a neutron at each scattering reaction during the slowing down is much bigger than the lethargy width of a resonance, so that a neutron has generally a large probability of skipping the resonance region in three or four jumps, as a kangaroo would jump above deep but narrow traps.[3]

A first solution to deal with the resonances would be to discretize the energy variable enough, so that we arrive at the experimental curve of the cross sections. Some codes have adopted this way, in particular the codes preparing the cross-section libraries like NJOY99. However, it is not feasible to apply this solution in neutronics codes because it would require more than 12000 groups of energy; any calculation would have prohibitive costs of time and memory, and the convergence would be very difficult to achieve.

With a coarser energy mesh (e.g., 295 groups),[11] it is not possible to describe all resonances in the epithermal domain. Taking an average without precaution would result in a stronger average flux than the one observed because resonances are not described. This phenomenon is called resonance self-shielding.[3]

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

March 2011
25

The retained solution is to adjust the cross sections to recover somehow the average flux that would have been calculated with a very detailed energy mesh describing the resonances. The cross sections that lead to the correct average flux in the resonance regions are called self-shielded cross sections. They are calculated by the formula:

$$\vec{\sigma}_{\rho,g} = \mu_g \frac{\int_{u_{g-1}}^{u_g} \sigma_{\rho,g}\phi(u)\,du}{\int_{u_{g-1}}^{u_g} \phi(u)\,du} \tag{2-98}$$

where μ_g is a superhomogeneization factor that stands to correct the geometrical differences, and u the lethargy.

This procedure conserves the reaction rate that would be obtained with an exact calculation of the flux. The difficulty is that we do not know this exact flux. To solve this problem, Livolant and Jeanpierre have proposed a series of approximations.[2,12] The first one is that the effect of the resonances would be the same whatever the geometry; therefore, the flux can be separated into two pieces:

$$\phi(\vec{r},u,\vec{\Omega}) = \varphi(\vec{r},u,\vec{\Omega}) \times \psi(\vec{r},u) \tag{2-99}$$

where $\varphi(\vec{r},u,\vec{\Omega})$ is a resonant fine-structure function carrying the resonance information, and $\psi(\vec{r},u)$ the macroscopic flux that would be observed without resonances.

It is also the asymptotic behavior of the flux between the resonances.

The macroscopic flux is affected by the slowing down operator for nuclear reactions with non-resonant isotopes, but not by the slowing-down operator for resonant nuclides. On the contrary, the fine-structure function is only affected by the resonant operator. We write:

$$\mathcal{R}^+\left(\phi(\vec{r},u,\vec{\Omega})\right) = \Sigma_{s0}^+(\vec{r},u)\psi(\vec{r},u)$$

$$\mathcal{R}^*\left(\phi(\vec{r},u,\vec{\Omega})\right) = \psi(\vec{r},u)\mathcal{R}^*\left(\varphi(\vec{r},u)\right) \tag{2-100}$$

Another approximation consists of assuming that the macroscopic flux is spatially flat:

$$\vec{\Omega} \cdot \vec{\nabla}\phi(\vec{r},u,\vec{\Omega}) = \psi(\vec{r},u,\vec{\Omega})\left[\vec{\Omega} \cdot \vec{\nabla}\varphi(\vec{r},u,\vec{\Omega})\right] \tag{2-101}$$

This approximation enables us to simplify the macroscopic flux from the slowing-down equation and obtain an equation for the fine-structure function, which will be solved using more approximations. The fine-structure function will then be used to compute the self-shielded cross sections:

$$\begin{cases} \vec{\Omega} \cdot \vec{\nabla}\varphi(\vec{r},u,\vec{\Omega}) + \Sigma(\vec{r},u)\varphi(\vec{r},u) = \frac{1}{4\pi}\left[\Sigma_s^+(\vec{r},u) + \mathcal{R}^*\left(\varphi(\vec{r},u)\right)\right] \\ \vec{\sigma}_{\rho,g} = \mu_g \cdot \frac{\int_{u_{g-1}}^{u_g} \sigma_{\rho,g}\varphi(u)\,du}{\int_{u_{g-1}}^{u_g} \varphi(u)\,du} \end{cases} \tag{2-102}$$

Solving Equation 2-102 requires approximations on the slowing-down term before using classical tools of solution of the transport equation. Several possibilities have been tried, and it is still an object of interesting research. One possibility is called probability tables and has been widely used in the DRAGON calculations.[2,9] It consists of replacing a Riemann integral by a Lebesgue integral and then approximating it by Dirac Delta functions. Mathematically, we have:

$$\begin{cases} \underbrace{\frac{1}{u_g - u_{g-1}}\int_{u_{g-1}}^{u_g} \sigma_{\rho,g}\,\varphi\left(\sigma_{\rho,g}(u)\right)\,du}_{Riemann\ integral} = \underbrace{\frac{\int_0^{max(\sigma_{\rho,g})} \Pi(\sigma)\sigma_{\rho,g}\varphi(\sigma)\,d\sigma}{Legesgue\ integral}} \\ Approximation: \Pi(\sigma)\sigma_{\rho,g}\varphi(\sigma) \approx \sum_{k=1}^K \delta(\sigma - \sigma_{g,k})\omega_k\sigma_{\rho,g,k}\phi(\sigma_{g,k}) \end{cases}$$

$$\Rightarrow \frac{1}{\mu_g - \mu_{g-1}} \int_{u_{g-1}}^{\mu_g} \sigma_{\rho,g}\, \varphi\left(\sigma_{\rho,g}(u)\right)\, du \approx \sum_{k=1}^{K} \delta(\sigma - \sigma_{g,k})\omega_k \sigma_{\rho,g,k}\phi(\sigma_{g,k}) \tag{2-103}$$

The set of values $\{\omega_k, \sigma_{k,g}\}k \in [1; K]$ is the probability table of Order K for Group g, and for the reaction and nuclide considered. The ω_k constitutes a quadrature set. More information about their establishment and the details of this method can be found in articles from Livolent and Jeanpierre, Cullen, and Ribon and Maillard.[2,13,14] This method is currently considered to be one of the best to calculate the self-shielded cross sections.

The self-shielding step is very sensitive in a calculation scheme, because it directly affects the cross sections used for the solution of the transport equation. It can potentially create significant discrepancies between different codes. It is also one of the major differences between deterministic codes and Monte-Carlo codes that use continuous energy group structures.

2.11.2   The Homogeneization and Condensation: SPH Equivalence Technique

Once the transport equation has been solved, lattice codes are asked to produce homogenized cross sections for full-core calculations. In addition, full-core calculations are usually done with fewer groups to gain speed. Therefore, a condensation has to be performed (e.g., from 295 groups to 26 groups).

Merging the cross sections over different regions requires one to establish a macro-balance relation between the detailed cross sections and the merged one. We perform an average of those cross sections weighted by the flux. This procedure is called the flux-volume homogenization.[15]

Let us define a collection of N regions of volumes V_i, $i \in [1; N]$. The volume of the merged region is defined by:

$$V_m = \sum_{i=1}^{N} V_i \tag{2-104}$$

Let us consider that the lattice calculation has been done with a set of G_g groups, $g \in [1; G]$ and that we would like to condense it to M groups, $M \ll G$. We define M_k as a collection of several groups $g \in [1; G]$, so that $\cup_{k=1}^{M} M_k = \cup_{g=1}^{G} G_g$.

A lattice calculation provides the heterogeneous fluxes $(\emptyset_{i,g})$ in each group $g \in [1; G]$ and region $i \in [1; N]$. The homogenized cross sections of the merged Region V_m are calculated by the following formula for Reaction x:

$$\bar{\Sigma}_{x,m,k} = \frac{\sum_{g\epsilon M_k}\sum_{i=1}^{N}\Sigma_{x,i,g}|V_i\phi_{i.g} \ \}\, Reaction\ rate}{\sum_{g\epsilon M_k}\sum_{i=1}^{N} V_i\phi_{i,g} \quad \}\, Integrated\ flux} \tag{2-105}$$

Theoretically, this procedure conserves the reaction rates calculated by the lattice calculation. This is achieved only if, when using the merged cross sections, the full-core calculation produces for each coarse energy group k a homogeneous flux $\Phi_{\text{hom},k}$ equal to the integrated flux calculated above.[15] But in fact, this situation is not very likely to occur. In this case, the full-core calculation is producing a homogeneous flux $\Phi_{\text{hom},k}$ so that we have:

$$\Phi_{\text{hom},k} \neq \sum_{g\epsilon M_k}\sum_{i=1}^{N} V_i\phi_{i,g}$$

$$\bar{\Sigma}_{x,m,k}\Phi_{\text{hom},k} \neq \sum_{g\epsilon M_k}\sum_{i=1}^{N}\Sigma_{x,i,g}V_i\phi_{i,g} \tag{2-106}$$

If we think that the reaction rates provided by the lattice code are better than those obtained in the full-core calculation using the simple flux-volume homogenization, then it is possible to force the conservation of those reaction rates by doing a super homogenization (SPH).[2,15,16]

It consists in adding a constant μ_k called SPH-factor during the homogenization:

$$\tilde{\Sigma}_{x,m,k} = \underbrace{\mu_k}_{SPH-factor} \frac{\sum_{g\epsilon M_k} \sum_{i-1}^{N} \Sigma_{x,i,g} V_i \phi_{i,g}}{\sum_{g\epsilon M_k} \sum_{i-1}^{N} V_i \phi_{i,g}} \tag{2-107}$$

This SPH-factor is determined so that we preserve the reaction rates:

$$\tilde{\Sigma}_{x,m,k}\Phi_{hom,k} = \sum_{g\epsilon M_k}\sum_{i=1}^{N}\Sigma_{x,i,g}V_i\phi_{i,g}$$

$$\Leftrightarrow \mu_k \frac{\sum_{g\epsilon M_k}\sum_{i-1}^{N}\Sigma_{x,i,g}V_i\phi_{i,g}}{\sum_{g\epsilon M_k}\sum_{i-1}^{N}V_i\phi_{i,g}} \cdot \Phi_{hom,k} = \sum_{g\epsilon M_k}\sum_{i-1}^{N}\Sigma_{x,i,g}V_i\phi_{i,g}$$

$$\Leftrightarrow \mu_k \frac{\sum_{g\epsilon M_k}\sum_{i-1}^{N}V_i\phi_{i,g}}{\Phi_{hom,k}} \tag{2-108}$$

The algorithm starts at iteration $i = 1$ with: $\forall k \in M_k, \mu_k^1 = 1$. It produces a first set of homogeneous fluxes in each coarse group $\left(\Phi_{hom,k}^1\right)_{k\epsilon M_k}$. Then we compute the second set of SPH factors, use them to correct the homogenized macroscopic cross sections, and redo the full-core calculation. This second iteration produces a second set of homogeneous fluxes and a third set of SPH factors. We iterate on the process until the SPH factors are converged:

$$\forall\epsilon \in {}^+, \exists\, i \in \mathbb{R}^+, \exists i \in \mathbb{N}^*/\forall k\epsilon[\![1:M]\!] \left|\frac{\mu_k^{i+1} - \mu_k^i}{\mu_k^i}\right| \leqslant \epsilon \tag{2-109}$$

Certainly, this procedure is time-consuming, because we have to do the full-core calculation several times before reaching convergence. However, the full-core calculation may be fast and when it converges, the SPH homogenization ensures that we correctly preserve the reaction rates of the lattice calculation at the core level so that the solution may be much more accurate. This is particularly true regarding the power shape in the core.

2.12 Full-core Calculation: the P~N~ Method

Two classes of methods are used to perform whole core calculations within a relatively short calculation time: the diffusion codes and the transport P_N codes. Other transport methods would require too much time and memory to be efficient.

Diffusion codes necessitate the calculation of a diffusion coefficient after the lattice calculation. This is done using leakage models and the fundamental mode approximation. This strategy is well suited for reactors moderated by water because each assembly is quite decoupled from their neighbors. The mean free paths of the thermal neutrons are small regarding the dimensions of the assemblies, so that the fundamental mode approximation is well verified and the diffusion codes give good results. But for reactors with higher mean free paths like fast reactors or graphite moderated reactors, the diffusion approximations may be less suitable. It explains why transport codes were developed for core analysis purposes to avoid the diffusion approximation.

The P_N method is based on an expansion of the streaming operator with spherical harmonics and Legendre polynomials. The integer N of P_N represents the order of truncation of the series.

2.12.1 The P$_1$ Equations

We will first explain the P$_1$ equations with only one group. The derivation in multi-group theory is similar and will be developed later. To understand the derivation, keep in mind the following identities[10]:

$$\int_{4\pi} d^2\Omega = 4\pi \tag{2-110}$$

$$\int_{4\pi} \vec{\Omega} d^2\Omega = \vec{0} \tag{2-111}$$

$$\int_{4\pi} \vec{A} \cdot \vec{\Omega} d^2\Omega = 0 \tag{2-112}$$

$$\int_{4\pi} \vec{\Omega} \left(\vec{\Omega} \cdot \vec{A} \right) d^2\Omega = \frac{4\pi}{3} \vec{A} \tag{2-113}$$

$$\int_{4\pi} (\vec{\Omega} \cdot \vec{A}) (\vec{\Omega} \cdot \vec{B}) d^2\Omega = \frac{4\pi}{3} \vec{A} \cdot \vec{B} \tag{2-114}$$

$$\int_{4\pi} (\vec{\Omega} \cdot \vec{A}) (\vec{\Omega} \cdot \vec{B}) \vec{\Omega} d^2\Omega = \vec{0}. \tag{2-115}$$

where $\vec{A}$ and $\vec{B}$ are two vectors that do not depend on $\vec{\Omega}$.

We approximate the angular flux by:

$$\phi(\vec{r}, E, \vec{\Omega}) = \frac{1}{4\pi} \phi_0(\vec{r}, E) + \frac{3}{4\pi} \vec{\Omega} \cdot \vec{\phi}_1(\vec{r}, E). \tag{2-116}$$

We remark that those coefficients have a clear physical meaning[4]:

$$\phi(\vec{r}, E) = \int_{4\pi} \phi\left(\vec{r}, E, \vec{\Omega}\right) d^2\Omega \tag{2-117}$$

$$= \frac{1}{4\pi} \phi_0(\vec{r}, E) \underbrace{\int_{4\pi} d^2\Omega}_{4\pi} + \frac{3}{4\pi} \underbrace{\int_{4\pi} \vec{\phi}_1(\vec{r}, E) \cdot \vec{\Omega} d^2\Omega}_{0} \tag{2-118}$$

$$= \phi_0(\vec{r}, E) \tag{2-119}$$

$$\vec{J}(\vec{r}, E) = \int_{4\pi} \phi\left(\vec{r}, E, \vec{\Omega}\right) \vec{\Omega} d^2\Omega \tag{2-120}$$

$$= \frac{1}{4\pi} \phi_0(\vec{r}, E) \underbrace{\int_{4\pi} \vec{\Omega} d^2\Omega}_{0} + \frac{3}{4\pi} \underbrace{\int_{4\pi} (\vec{\phi}_1(\vec{r}, E) \cdot \vec{\Omega}) \vec{\Omega} d^2\Omega}_{\frac{4\pi}{3}\vec{\phi}_1(\vec{r}, E)} \tag{2-121}$$

$$= \vec{\phi}_1(\vec{r}, E) \tag{2-122}$$

Therefore, we will directly redefine the angular flux with the physical expression of the coefficients:

$$\phi(\vec{r}, E, \vec{\Omega}) = \frac{1}{4\pi} \phi(\vec{r}, E) + \frac{3}{4\pi} \vec{\Omega} \cdot \vec{J}(\vec{r}, E). \tag{2-123}$$

We transfer this approximation in the transport equation and we integrate over 4π with infinitesimal volume $d^2\Omega$:

$$\underbrace{\int_{4\pi} \vec{\Omega} \cdot \vec{\nabla} \left[\phi(\vec{r}, E, \vec{\Omega})\right] d^2\Omega}_{I_1} + \underbrace{\int_{4\pi} \Sigma(\vec{r}, E) \, \phi(\vec{r}, E, \vec{\Omega}) d^2\Omega}_{I_2}$$

$$= \underbrace{\int_{4\pi} \int_{4\pi} \int_0^{+\infty} \Sigma_s(\vec{r}, E \leftarrow E', \vec{\Omega} \leftarrow \vec{\Omega}') \, \phi(\vec{r}, E', \vec{\Omega}') \, dE' d^2\Omega' \, d^2\Omega}_{I_3}$$

$$+ \underbrace{\frac{1}{4\pi K_{\text{eff}}} \sum_{j=1}^{J^{fiss}} \int_{4\pi} Xj \, (E) \int_0^{+\infty} \nu\Sigma_{f,j}(\vec{r}, E') \, \phi(\vec{r}, E', \vec{\Omega}) \, dE' d^2\Omega}_{I_4} . \tag{2-124}$$

We will then deal with each term separately.

$$\vec{\Omega} \cdot \vec{\nabla} \left[\phi(\vec{r}, E, \vec{\Omega})\right] = \vec{\Omega} \cdot \vec{\nabla} \left[\frac{1}{4\pi} \phi(\vec{r}, E) + \frac{3}{4\pi} \vec{\Omega} \cdot \vec{J}(\vec{r}, E)\right] \tag{2-125}$$

$$= \frac{1}{4\pi} \vec{\Omega} \cdot \vec{\nabla}\phi(\vec{r}, E) + \frac{3}{4\pi} \vec{\Omega} \cdot \vec{\nabla} \left[\vec{\Omega} \cdot \vec{J}(\vec{r}, E)\right] \tag{2-126}$$

$$I_1 = \int_{4\pi} \vec{\Omega} \cdot \vec{\nabla} \left[\phi(\vec{r}, E, \vec{\Omega})\right] d^2\Omega \tag{2-127}$$

$$= \frac{1}{4\pi} \vec{\nabla}\phi(\vec{r}, E) \cdot \underbrace{\int_{4\pi} \vec{\Omega} \, d^2\Omega}_{=\vec{0}} + \frac{3}{4\pi} \int_{4\pi} \vec{\Omega} \cdot \vec{\nabla} \left[\vec{\Omega} \cdot \vec{J}(\vec{r}, E)\right] d^2\Omega \tag{2-128}$$

$$= \frac{3}{4\pi} \int_{4\pi} \vec{\Omega} \cdot \vec{\nabla} \left[\vec{\Omega} \cdot \vec{J} \, (\vec{r}, E)\right] d^2\Omega \tag{2-129}$$

$$\Leftrightarrow I_1 = \frac{3}{4\pi} \int_0^1 \int_{4\pi} \text{div} \left[(\vec{J} \, (\vec{r}, E) \cdot \vec{\Omega}) \, \vec{\Omega}\right] d^2\Omega \, \rho^2 \, d\rho$$

using the Green-Ostrogradski theorem:

$$= \frac{3}{4\pi} \int_{4\pi} (\vec{J}(\vec{r}, E) \cdot \vec{\Omega}) \, \vec{\Omega} \cdot \vec{\Omega} \, d^2\Omega$$

using the Green-Ostrogradski theorem in the other direction:

$$= \frac{3}{4\pi} \int_{4\pi} (\vec{J}(\vec{r}, E) \cdot \vec{\Omega}) \, \underbrace{(\vec{\Omega} \cdot \vec{\Omega})}_{=1} d^2\Omega$$

$$= \frac{3}{4\pi} \int_0^1 \int_{4\pi} \text{div} \, \vec{J}(\vec{r}, E) \, d^2\Omega \, \rho^2 d\rho$$

$$= \frac{3}{4\pi} \text{div} \, \vec{J} \, (\vec{r}, E) \underbrace{\int_0^1 \int_{4\pi} d^2\Omega \, \rho^2 \, d\rho}_{=\frac{4\pi}{3}}$$

$$= \text{div} \, \vec{J}(\vec{r}, E). \tag{2-130}$$

$$I_2 = \int_{4\pi} \Sigma(\vec{r}, E) \, \phi(\vec{r}, E, \vec{\Omega}) \, d^2\Omega$$

$$= \Sigma \int_{4\pi} \frac{1}{4\pi} \phi(\vec{r}, E) + \frac{3}{4\pi} \vec{\Omega} \cdot \vec{J}(\vec{r}, E) \, d^2\Omega$$

$$= \frac{1}{4\pi} \Sigma \phi(\vec{r}, E) \int_{4\pi} d^2\Omega$$

$$= \Sigma \phi(\vec{r}, E). \tag{2-131}$$

For the scattering source, we also assume a P_1 approximation:

$$\Sigma_s(\vec{r}, E \leftarrow E', \vec{\Omega} \leftarrow \vec{\Omega}') = \frac{1}{4\pi} \Sigma_{s0}(\vec{r}, E \leftarrow E') + \frac{3}{4\pi} \Sigma_{s,1}(\vec{r}, E \leftarrow E') \, \vec{\Omega} \cdot \vec{\Omega}'. \tag{2-132}$$

Using this, we have:

$$I_3 = \int_{4\pi} \int_{4\pi} \int_0^{+\infty} \Sigma_s(\vec{r}, E \leftarrow E', \vec{\Omega} \leftarrow \vec{\Omega}') \, \phi(\vec{r}, E', \vec{\Omega}') \, dE' d^2\Omega' \, d^2\Omega$$

$$= \int_{4\pi} \int_{4\pi} \int_0^{+\infty} \left[\frac{1}{4\pi} \Sigma_{s0}(\vec{r}, E \leftarrow E') + \frac{3}{4\pi} \Sigma_{s,1}(\vec{r}, E \leftarrow E') \, \vec{\Omega} \cdot \vec{\Omega}' \right] \times$$

$$\left[\frac{1}{4\pi} \phi(\vec{r}, E') + \frac{3}{4\pi} \vec{\Omega}' \cdot \vec{J}(\vec{r}, E') \right] dE' d^2\Omega' d^2\Omega$$

$$= \frac{1}{4\pi} \int_0^{+\infty} \int_{4\pi} \Sigma_{s0}(E \leftarrow E') \, \phi(\vec{r}, E') \, dE' d^2\Omega$$

$$= \int_0^{+\infty} \Sigma_{s0}(E \leftarrow E') \, \phi(\vec{r}, E') \, dE' \tag{2-133}$$

Indeed, we remark that:

$$\int_{4\pi} \int_{4\pi} \vec{\Omega} \cdot \vec{\Omega} \, d^2\Omega \, d^2\Omega' = \underbrace{\int_{4\pi} \vec{\Omega} \, d^2\Omega}_{=\vec{0}} \cdot \underbrace{\int_{4\pi} \vec{\Omega}' \, d^2\Omega'}_{=\vec{0}} = \vec{0} \tag{2-134}$$

For I_4, we just integrate the angular flux, which leads to:

$$I_4 = \frac{1}{4\pi K_{eff}} \sum_{j=1}^{J^{fiss}} \int_{4\pi} \chi_j(E) \int_0^{+\infty} \nu\Sigma_{f,j}(\vec{r}, E') \, \phi(\vec{r}, E') \, dE' d^2\Omega$$

$$= \frac{1}{K_{eff}} \sum_{j=1}^{J^{fiss}} \chi_j(E) \int_0^{+\infty} \nu\Sigma_{f,j}(\vec{r}, E') \, \phi(\vec{r}, E') \, dE' \tag{2-135}$$

Finally, we obtain a first equation:

$$\text{div} \, \vec{J}(\vec{r}, E) + \Sigma\phi(\vec{r}, E) = \int_0^{+\infty} \Sigma_{s0}(E \leftarrow E')\phi(\vec{r}, E) \, dE' \tag{2-136}$$

$$+ \frac{1}{K_{eff}} \sum_{j=1}^{J^{fiss}} \chi_j(E) \int_0^{+\infty} \nu\Sigma_{f,j}(\vec{r}, E') \, \phi(\vec{r}, E') \, dE'.$$

A second, vectorial equation can be obtained by multiplying the transport equation by $\vec{\Omega}$ before integrating over 4π on $d^2\Omega$:

$$\underbrace{\int_{4\pi} \left(\vec{\Omega} \cdot \vec{\nabla}\left[\phi(\vec{r},E,\vec{\Omega})\right]\right)\vec{\Omega} d^2\Omega}_{I_1} + \underbrace{\int_{4\pi} \Sigma(\vec{r},E)\phi(\vec{r},E,\vec{\Omega})\vec{\Omega} d^2\Omega}_{I_2}$$

$$= \underbrace{\int_{4\pi}\int_{4\pi}\int_0^{+\infty} \Sigma_s(\vec{r},E \leftarrow E',\vec{\Omega} \leftarrow \vec{\Omega}')\, \phi(\vec{r},E,\vec{\Omega}')\vec{\Omega}dE'd^2\Omega'\, d^2\Omega}_{I_3} \qquad (2\text{-}137)$$

$$+ \underbrace{\frac{1}{4\pi K_{eff}} \sum_{j=1}^{J^{fiss}} \int_{4\pi}\chi_j(E)\int_0^{+\infty} \nu\Sigma_{f,j}(\vec{r},E')\int_{4\pi}\phi(\vec{r},E',\vec{\Omega}')\,\vec{\Omega}\ dE'd^2\Omega'\, d^2\Omega}_{I_4}\,.$$

In this case, the simplification of the streaming operator part is not totally obvious.

$$I_1 - \int_{4\pi} \left(\vec{\Omega} \cdot \vec{\nabla}\left[\phi(\vec{r},E,\vec{\Omega})\right]\right)\vec{\Omega}\, d^2\Omega$$

$$= \int_{4\pi} \left(\vec{\Omega} \cdot \vec{\nabla}\left[\frac{1}{4\pi}\phi(\vec{r},E) + \frac{3}{4\pi}\vec{\Omega} \cdot \vec{J}(\vec{r},E)\right]\right)\vec{\Omega}\, d^2\Omega$$

$$= \underbrace{\frac{1}{4\pi}\int_{4\pi}\left[\vec{\Omega} \cdot \vec{\nabla}\left(\phi(\vec{r},E)\right)\right]\vec{\Omega}\, d^2\Omega}_{I_a} + \underbrace{\frac{3}{4\pi}\int_{4\pi}\left[\vec{\Omega} \cdot \vec{\nabla}\left(\vec{\Omega} \cdot \vec{J}(\vec{r},E)\right)\right]\vec{\Omega}\, d^2\Omega}_{I_b} \qquad (2\text{-}138)$$

$$I_a = \frac{1}{4\pi}\int_{4\pi}\left[\vec{\Omega} \cdot \vec{\nabla}\left(\phi(\vec{r},E)\right)\right]\vec{\Omega}\, d^2\Omega \qquad (2\text{-}139)$$

$$= \frac{1}{4\pi}\frac{4\pi}{3}\vec{\nabla}\phi(\vec{r},E) \qquad (2\text{-}140)$$

$$= \frac{1}{3}\overrightarrow{\mathrm{grad}}\phi(\vec{r},E) \qquad (2\text{-}141)$$

For I_b we have to work on the integrand: we expand the gradient:

$$\vec{\nabla}\left(\vec{\Omega} \cdot \vec{J}\right) = \left(\vec{\Omega} \cdot \vec{J}\right)\vec{J} + \left(\vec{J} \cdot \vec{\nabla}\right)\vec{\Omega} + \vec{\Omega} \wedge \left(\vec{\nabla} \wedge \vec{J}\right) + \vec{J} \wedge \left(\vec{\nabla} \wedge \vec{\Omega}\right)\,. \qquad (2\text{-}142)$$

We take the scalar product with $\vec{\Omega}$:

$$\vec{\nabla}\left(\vec{\Omega} \cdot \vec{J}\right) \cdot \vec{\Omega} = \left(\vec{\Omega} \cdot \vec{J}\right)\vec{J} \cdot \vec{\Omega} + \left(\vec{J} \cdot \vec{\nabla}\right)\underbrace{\vec{\Omega} \cdot \vec{\Omega}}_{=1} + \underbrace{\vec{\Omega} \wedge \left(\vec{\nabla} \wedge \vec{J}\right) \cdot \vec{\Omega}}_{=0 \text{ because } \vec{\Omega}\perp\vec{\Omega}\,\wedge(\dots)} + \vec{J} \wedge \underbrace{\left(\vec{\nabla} \wedge \vec{\Omega}\right)}_{=\overrightarrow{\mathrm{rot}}\,\vec{\Omega}=\vec{0}} \cdot \vec{\Omega} \qquad (2\text{-}143)$$

$$= \left(\vec{\Omega} \cdot \vec{J}\right)\vec{J} \cdot \vec{\Omega} + \vec{J} \cdot \vec{\nabla}\,. \qquad (2\text{-}144)$$

Recalling that $\overrightarrow{\mathrm{rot}}\,\vec{\Omega} = \vec{0}$ because $\vec{\Omega}$ is a unit vector, we replace in the integral I_b:

$$I_b = \frac{3}{4\pi}\int_{4\pi}\left[\vec{\Omega} \cdot \vec{\nabla}\left(\vec{\Omega} \cdot \vec{J}(\vec{r},E)\right)\right]\vec{\Omega}\, d^2\Omega \qquad (2\text{-}145)$$

$$= \frac{3}{4\pi}\underbrace{\int_{4\pi}\left(\vec{\Omega} \cdot \vec{J}\right)\left(\vec{J} \cdot \vec{\Omega}\right)\vec{\Omega}\, d^2\Omega}_{=\vec{0}} + \frac{3}{4\pi}\underbrace{\int_{4\pi}\left(\vec{J} \cdot \vec{\nabla}\right)\vec{\Omega}\, d^2\Omega}_{=\vec{0}} \qquad (2\text{-}146)$$

$$= \vec{0} \qquad (2\text{-}147)$$

Thus, we have:

$$I_1 = \int_{4\pi}\left(\vec{\Omega} \cdot \vec{\nabla}\left[\phi(\vec{r},E,\vec{\Omega})\right]\right)\vec{\Omega}\, d^2\Omega$$

$$= \frac{1}{3}\overrightarrow{\mathrm{grad}}\,\phi(\vec{r},E)\,. \qquad (2\text{-}148)$$

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor
Cross Section Generation

32   March 2011

$$I_2 = \int_{4\pi} \Sigma(\vec{r}, E)\, \phi(\vec{r}, E, \vec{\Omega})\, \vec{\Omega}\, d^2\Omega \tag{2-149}$$

$$= \Sigma(\vec{r}, E)\, \vec{J}(\vec{r}, E) \text{ by definition of } \vec{J}(\vec{r}, E). \tag{2-150}$$

$$I_3 = \int_{4\pi} \int_{4\pi} \int_0^{+\infty} \Sigma_s(\vec{r}, E \leftarrow E', \vec{\Omega} \leftarrow \vec{\Omega}')\phi(\vec{r}, E', \vec{\Omega}')\, \vec{\Omega}\, dE'\, d^2\Omega'\, d^2\Omega$$

$$= \int_{4\pi} \int_{4\pi} \int_0^{+\infty} \left[\frac{1}{4\pi}\Sigma_{s0}(\vec{r}, E \leftarrow E') + \frac{3}{4\pi}\Sigma_{s1}(E \leftarrow E')\, \vec{\Omega} \cdot \vec{\Omega}'\right]$$

$$\times \left[\frac{1}{4\pi}\phi(\vec{r}, E') + \frac{3}{4\pi}\vec{\Omega}' \cdot \vec{J}(\vec{r}, E')\right]\vec{\Omega}\, dE'\, d^2\Omega'\, d^2\Omega$$

$$= \left(\frac{3}{4\pi}\right)^2 \int_{4\pi}\int_{4\pi}\int_0^{+\infty} \Sigma_{s1}(E \leftarrow E')\, (\vec{\Omega} \cdot \vec{\Omega}')\left(\vec{\Omega}' \cdot \vec{J}(\vec{r}, E')\right)\vec{\Omega}\, dE'\, d^2\Omega'\, d^2\Omega \tag{2-151}$$

Other terms are null because: $\int_{4\pi} \vec{\Omega}\, d^2\Omega = \vec{0}$ and $\int_{4\pi} \vec{\Omega}'\, d^2\Omega' = \vec{0}$

$$I_3 = \left(\frac{3}{4\pi}\right)^2 \int_0^{+\infty} \Sigma_{s1}(E \leftarrow E')\int_{4\pi}\left(\vec{\Omega}' \cdot \vec{J}(\vec{r}, E')\right)\underbrace{\int_{4\pi}(\vec{\Omega} \cdot \vec{\Omega}')\,\vec{\Omega}\, d^2\Omega\, d^2\Omega'dE'}_{=\frac{4\pi}{3}\vec{\Omega}'}$$

$$= \frac{3}{4\pi}\int_0^{+\infty}\Sigma_{s1}(E \leftarrow E')\underbrace{\int_{4\pi}\left(\vec{\Omega}' \cdot \vec{J}(\vec{r}, E')\right)\vec{\Omega}'d^2\Omega'dE'}_{=\frac{4\pi}{3}\vec{J}(\vec{r}.E')}$$

$$= \int_0^{+\infty}\Sigma_{s1}(E \leftarrow E')\vec{J}(\vec{r}, E')\, dE' \tag{2-152}$$

The fission source disappears:

$$I_4 = \frac{1}{4\pi K_{eff}}\sum_{j=1}^{J^{fiss}}\chi_j(E)\int_0^{+\infty}\nu\Sigma_{f,j}(\vec{r}, E')\int_{4\pi}\phi(\vec{r}, E', \vec{\Omega})\int_{4\pi}\vec{\Omega}\, d^2\Omega\, d^2\Omega'dE' \tag{2-153}$$

$$= \frac{1}{4\pi K_{eff}}\sum_{j=1}^{J^{fiss}}\chi_j(E)\int_0^{+\infty}\nu\Sigma_{f,j}(\vec{r}, E')\left\{\frac{1}{4\pi}\phi(\vec{r}, E')\underbrace{\int_{4\pi}\vec{\Omega}'d^2\Omega'}_{=\vec{0}}\right. \tag{2-154}$$

$$\left. + \frac{3}{4\pi}\underbrace{\int^{4\pi}\left(\vec{\Omega}' \cdot \vec{J}(\vec{r}, E')\right)d^2\Omega'}_{=\vec{0}}\right\}\underbrace{\int_{4\pi}\vec{\Omega}\, d^2\Omega\, dE'}_{=\vec{0}} \tag{2-155}$$

$$= \vec{0} \tag{2-156}$$

Finally, we obtain the second equation:

$$\frac{1}{3}\overrightarrow{\text{grad}}\phi(\vec{r}, E) + \Sigma(\vec{r}, E)\vec{J}(\vec{r}, E) = \int_0^{+\infty}\Sigma_{s1}(E \leftarrow E')\vec{J}(\vec{r}, E')\, dE' \tag{2-157}$$

The P$_1$ approximation consists in resolving the system[4]:

$$\left\{ \begin{array}{l} \operatorname{div} \vec{J}(\vec{r}, E) \; + \; \Sigma\phi(\vec{r}, E) = \int_0^{+\infty} \Sigma_{s0}\left(E \leftarrow E'\right)\phi(\vec{r}, E') \, \mathrm{d}E' \\[6pt] \qquad + \; \dfrac{1}{K_{eff}} \sum_{j=1}^{J^{fiss}} \chi_j(E) \int_0^{+\infty} \nu\Sigma_{f,j}(\vec{r}, E') \, \phi(\vec{r}, E') \, \mathrm{d}E' \\[6pt] \dfrac{1}{3}\overrightarrow{grad}\phi(\vec{r}, E') \; + \; \Sigma(\vec{r}, E) \, \vec{J}(\vec{r}, E) = \int_0^{+\infty} \Sigma_{s1}\left(E \leftarrow E'\right)\vec{J}(\vec{r}, E') \, \mathrm{d}E' \end{array} \right.$$

2.12.2   The Equivalence of P₁ and Diffusion in One-group Theory

We will now assume that all neutrons have the same energy: the P₁ system becomes[4]:

$$\left\{ \begin{array}{l} \operatorname{div} \vec{J}(\vec{r}) \; + \; \Sigma\phi(\vec{r}) = \underbrace{\Sigma_{s0}\phi(\vec{r}) \; + \; \dfrac{1}{K_{eff}} \sum_{j=1}^{J^{fiss}} \nu\Sigma_{f,j}(\vec{r})\phi(\vec{r}))}_{= S(\vec{r})} \\[12pt] \dfrac{1}{3}\overrightarrow{grad}\phi(\vec{r}) \; + \; \Sigma(\vec{r})\vec{J}(\vec{r}) = \Sigma_{s1}\vec{J}(\vec{r}) \end{array} \right.$$

Notice that the macroscopic differential cross section $\Sigma_{s1}(E \leftarrow E')$ had the dimension of a macroscopic cross section divided by energy ($cm^{-1}.J^{-1}$) whereas Σ_{s1} is a macroscopic cross section, whose dimension is cm^{-1}. Therefore, the previous formulae are homogeneous.

We can rewrite those equations with the average cosine of the scattering angle $\vec{\mu}_0 = \dfrac{2}{3A}$ and the transport cross section Σ_{tr}:

$$\vec{\mu}_0 \; = \; \dfrac{\Sigma_{s1}}{\Sigma_{s0}} \quad \Sigma_{tr} = \Sigma_{s0}\left(1 - \vec{\mu}_0\right) \quad \Sigma_a = \Sigma - \Sigma_{s0}$$

$$\Rightarrow \left\{ \begin{array}{l} \operatorname{div} \vec{J}(\vec{r}) \; + \; \Sigma_a\phi(\vec{r}) = S(\vec{r}) \\[6pt] \dfrac{1}{3}\overrightarrow{grad} \; \phi(\vec{r}) \; + \; [\Sigma_a(\vec{r}) + \Sigma_{tr}(\vec{r})]\vec{J}(\vec{r}) = 0 \, . \end{array} \right. \tag{2-158}$$

In the second equation, we recognize Fick's diffusion law:

$$\vec{J}(\vec{r}) \; = \; -D\overrightarrow{grad}\,\phi(\vec{r}) \; with \; D \; = \; \dfrac{1}{3(\Sigma_{tr} + \Sigma_a)} \tag{2-159}$$

The substitution into 2.158 leads to a neutron diffusion equation[4]:

$$D\Delta\phi(\vec{r}) - \Sigma_a\phi(\vec{r}) + S(\vec{r}) = 0.$$

$$\Leftrightarrow \Delta\phi(\vec{r}) - \dfrac{1}{L^2}\phi(\vec{r}) + \dfrac{S(\vec{r})}{D} = 0 \; with \; L = \sqrt{\dfrac{D}{\Sigma_a}} = \sqrt{\dfrac{1}{3\Sigma_a(\Sigma_{tr} + E_a)}} \tag{2-160}$$

The previous development demonstrates that solving the P₁ equation is equivalent to solving the diffusion equation when we just have one energy group.[4] In this case, we could mimic a diffusion calculation with a P₁ solver by performing the following change in the solver:

$$\tilde{\Sigma}_{total} = \dfrac{1}{3D} = \Sigma_a + \Sigma_{tr} \tag{2-161}$$

$$\tilde{\Sigma}_{s1} = 0 \tag{2-162}$$

$$\tilde{\Sigma}_{s0} = \Sigma_{tr} = \dfrac{1}{3D} - \Sigma_a \tag{2-163}$$

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

34   March 2011

assuming that we know D, Σ_{total} (or Σ_a) and Σ_{tr}.

Replacing the normal cross sections in 2.158 by the cross sections defined with a tilde above will lead to the exact formulation of the diffusion equation using the P_1 solver. Therefore, it is a good way to mimic a diffusion code because the two solvers should give the same answer.[1]

The independence of cross sections and fluxes from energy is clearly an approximation The following subsection demonstrates the consideration of more than one group of energy and its outcomes

2.12.3 Comparison of P_1 Equations and Diffusion Theory with a Multi-group Formulation

With more than one group, we cannot derive the Fick's Law as done previously because the second equation for the P_1 system is written:

$$\frac{1}{3}\overrightarrow{\mathrm{grad}}\,\phi(\vec{r},E) + \Sigma(\vec{r},E)\,\vec{J}(\vec{r},E) = \int_0^{+\infty} \Sigma_{s1}(E \leftarrow E')\,\vec{J}(\vec{r},E')\,\mathrm{d}E' \tag{2-164}$$

In this equation, we see that $\phi(\vec{r},E)$ is not only related to $\vec{J}(\vec{r},E)$, but also to all other currents at other energies through the scattering term. The multi-group formulation of the P_1 equations becomes[4]:

$$\forall g \in [\![1,G]\!], \begin{cases} \mathrm{div}\,\vec{J}_g(\vec{r}) + \Sigma_g^0\phi_g(\vec{r}) = \displaystyle\sum_{h=1}^{G}\Sigma_{s0,g\leftarrow h\phi h}\vec{\mathrm{r}} \\[2mm] \qquad\qquad + \dfrac{1}{K_{eff}}\displaystyle\sum_{j=1}^{J^{fiss}}\chi_{g,j}\sum_{h=1}^{G}v_h\Sigma_{f,j,h}(\vec{r})\phi_h(\vec{r}) \\[3mm] \frac{1}{3}\overrightarrow{grad}\phi_g(\vec{r}) + \Sigma_g^1(\vec{r})\,\vec{J}_g(\vec{r}) = \displaystyle\sum_{h=1}^{G}\Sigma_{s1,g\leftarrow h}\vec{J}_h(\vec{r}) \end{cases}$$

Notice that Σ_g^0 is weighted by the flux whereas Σ_g^1 is weighted by the current.

We rewrite those equations in a matrix form. To do this, we define the vector of the flux, carrying all fluxes in G groups and the matrix of the current carrying the three coordinates of the current vectors of the G groups. The following definitions are used:

$$\Phi = \begin{pmatrix}\phi_1\\\phi_2\\\vdots\\\phi_G\end{pmatrix} \underline{\underline{J}} = (J_{g,j})_{\substack{g\in[\![1,G]\!]\\ j\in[\![1,3]\!]}} = \begin{pmatrix} J_{1,1} & J_{1,2} & J_{1,3}\\ J_{2,1} & J_{2,2} & J_{2,3}\\ \vdots & \vdots & \vdots\\ J_{G,1} & J_{G,2} & J_{G,3}\end{pmatrix} \tag{2-165}$$

Then the system of P_1 equations becomes:

$$\begin{cases}\overrightarrow{\mathrm{Div}}\,\underline{\underline{J}}(\vec{r}) + \underline{\underline{\Sigma}}^0\overrightarrow{\Phi}(\vec{r}) = \underline{\underline{\Sigma}}_{s0}\overrightarrow{\Phi}(\vec{r}) + \dfrac{1}{K_{eff}}\Sigma_{j=1}^{J^{fiss}}\underline{\underline{\Sigma}}_{f,j}(\vec{r})\overrightarrow{\Phi}(\vec{r})\\[3mm] \qquad\frac{1}{3}\underline{\mathrm{Grad}}\Phi(\vec{r}) + \underline{\underline{\Sigma}}^1(\vec{r})\underline{\underline{J}}(\vec{r}) = \underline{\underline{\Sigma}}_{s1}\underline{\underline{J}}(\vec{r})\end{cases}$$

With the following definitions:

$$\overrightarrow{\text{Div}\underline{J}} = \left(\Sigma_{j=1}^3 \frac{\partial J_{i,j}}{\partial x_j}\right)_{i\in[\![1,G]\!]} = \begin{pmatrix} \frac{\partial J_{1,1}}{\partial x_1} + \frac{\partial J_{1,2}}{\partial x_2} + \frac{\partial J_{1,3}}{\partial x_3} \\ \frac{\partial J_{2,1}}{\partial x_1} + \frac{\partial J_{2,2}}{\partial x_2} + \frac{\partial J_{2,3}}{\partial x_3} \\ \vdots \\ \frac{\partial J_{G,1}}{\partial x_1} + \frac{\partial J_{G,2}}{\partial x_2} + \frac{\partial J_{G,3}}{\partial x_3} \end{pmatrix}$$

$$= \left[\left(\frac{\partial}{\partial x_1} \; \frac{\partial}{\partial x_2} \; \frac{\partial}{\partial x_3}\right) \cdot \underbrace{\begin{array}{l} J_{1,1} \, J_{2,1} \, J_{3,1} \; \cdots \; J_{G,1} \\ J_{2,1} \, J_{2,2} \, J_{3,2} \; \cdots \; J_{G,2} \\ J_{3,1} \, J_{2,3} \, J_{3,3} \; \cdots \; J_{G,3} \end{array}}_{\underline{\underline{J^T}}}\right]^T \tag{2-166}$$

$$\underline{\underline{\Sigma_{s0}}} = \left(\Sigma_{s0,g\leftarrow j}\right)_{\substack{g\in[\![1,G]\!] \\ j\in[\![1,G]\!]}} = \begin{pmatrix} \Sigma_{s0,1\leftarrow 1} & \Sigma_{s0,1\leftarrow 2} & \cdots & \Sigma_{s0,1\leftarrow G} \\ \Sigma_{s0,2\leftarrow 1} & \Sigma_{s0,2\leftarrow 2} & \cdots & \Sigma_{s0,2\leftarrow G} \\ \vdots & \vdots & \ddots & \vdots \\ \Sigma_{s0,G\leftarrow 1} & \Sigma_{s0,G\leftarrow 2} & \cdots & \Sigma_{s0,G\leftarrow G} \end{pmatrix} \tag{2-167}$$

$$\underline{\underline{\Sigma^0}} = diag\left(\Sigma_g^0\right)_{g\in[\![1,G]\!]} = \begin{pmatrix} \Sigma_1^0 & 0 & \cdots & 0 \\ 0 & \Sigma_2^0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \Sigma_G^0 \end{pmatrix}$$

$$\underline{\underline{\Sigma_f}} = \left(\nu_g \chi_g \Sigma_{f,j}\right)_{\substack{g\in[\![1,G]\!] \\ j\in[\![1,G]\!]}} = \begin{pmatrix} \nu_1\chi_1\Sigma_{f,1} & \nu_1\chi_1\Sigma_{f,2} & \cdots & \nu_1\chi_1\Sigma_{f,G} \\ \nu_2\chi_2\Sigma_{f,1} & \nu_2\chi_2\Sigma_{f,2} & \cdots & \nu_2\chi_2\Sigma_{f,G} \\ \vdots & \vdots & \ddots & \vdots \\ \nu_G\chi_G\Sigma_{f,1} & \nu_G\chi_G\Sigma_{f,2} & \cdots & \nu_G\chi_G\Sigma_{f,G} \end{pmatrix} \tag{2-168}$$

And for the second equation:

$$\underline{\underline{\text{Grad}\overrightarrow{\Phi}}} = \left(\frac{\partial \phi g}{\partial xj}\right)_{\substack{g\in[\![1,G]\!] \\ j\in[\![1,3]\!]}} = \begin{pmatrix} \frac{\partial \phi_1}{\partial x_1} & \frac{\partial \phi_1}{\partial x_2} & \frac{\partial \phi_1}{\partial x_3} \\ \frac{\partial \phi_2}{\partial x_1} & \frac{\partial \phi_2}{\partial x_2} & \frac{\partial \phi_2}{\partial x_3} \\ \vdots & \vdots & \vdots \\ \frac{\partial \phi G}{\partial x_1} & \frac{\partial \phi G}{\partial x_2} & \frac{\partial \phi G}{\partial x_3} \end{pmatrix} = \left[\begin{pmatrix} \frac{\partial}{\partial x_1} \\ \frac{\partial}{\partial x_2} \\ \frac{\partial}{\partial x_3} \end{pmatrix} \cdot \underbrace{(\phi 1 \quad \phi 2 \quad \cdots \quad \phi G)}_{\overrightarrow{\Phi T}}\right]^T \tag{2-169}$$

$$\underline{\underline{\Sigma_{s1}}} = diag\left(\Sigma_g^1\right)_{g\in[\![1,G]\!]} = \begin{pmatrix} \Sigma_1^1 & 0 & \cdots & 0 \\ 0 & \Sigma_2^1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \Sigma_G^1 \end{pmatrix} \tag{2-170}$$

$$\underline{\underline{\Sigma_{s1}}} = \left(\Sigma_{s1,g\leftarrow j}\right)_{(g,j)\in[\![1G]\!]^2} = \begin{pmatrix} \Sigma_{s1,1\leftarrow 1} & \Sigma_{s1,1\leftarrow 2} & \cdots & \Sigma_{s1,1\leftarrow G} \\ \Sigma_{s1,2\leftarrow 1} & \Sigma_{s1,2\leftarrow 2} & \cdots & \Sigma_{s1,2\leftarrow G} \\ \vdots & \vdots & \ddots & \vdots \\ \Sigma_{s1,G\leftarrow 1} & \Sigma_{s1,G\leftarrow 2} & \cdots & \Sigma_{s1,G\leftarrow G} \end{pmatrix} \tag{2-171}$$

So that we have, for the scattering terms for example, the matrix:

$$\underline{\underline{\Sigma}}_{s1}\underline{J} = \left(\sum_{k=1}^{G} \Sigma_{s1,g\leftarrow k}J_{k,j}\right)_{\substack{g\in[\![1,G]\!] \\ j\in[\![1,3]\!]}}$$

$$= \begin{pmatrix} \sum_{k=1}^{G}\Sigma_{s1,1\leftarrow k}J_{k,1} & \sum_{k=1}^{G}\Sigma_{s1,2\leftarrow k}J_{k,2} & \sum_{k=1}^{G}\Sigma_{s1,G\leftarrow k}J_{k,3} \\ \sum_{k=1}^{G}\Sigma_{s1,2\leftarrow k}J_{k,1} & \sum_{k=1}^{G}\Sigma_{s1,2\leftarrow k}J_{k,2} & \sum_{k=1}^{G}\Sigma_{s1,2\leftarrow k}J_{k,3} \\ \vdots & \vdots & \vdots \\ \sum_{k=1}^{G}\Sigma_{s1,G\leftarrow k}J_{k,1} & \sum_{k=1}^{G}\Sigma_{s1,G\leftarrow k}J_{k,2} & \sum_{k=1}^{G}\Sigma_{s1,G\leftarrow k}J_{k,3} \end{pmatrix} \qquad (2\text{-}172)$$

We will then try to formulate the second equation in terms of Fick's Law:

$$\frac{1}{3}\underline{\mathrm{Grad}}\vec{\Phi}(\vec{r}) + \underline{\underline{\Sigma}}^1(\vec{r})\underline{J}(\vec{r}) = \underline{\underline{\Sigma}}_{s1}(\vec{r})\underline{J}(\vec{r})$$

$$\Leftrightarrow 3\left(\underline{\underline{\Sigma}}^1(\vec{r}) - \underline{\underline{\Sigma}}_{s1}(\vec{r})\right)\underline{J}(\vec{r}) = -\underline{\mathrm{Grad}}\vec{\Phi}(\vec{r}) \qquad (2\text{-}173)$$

We can suppose that the matrix in front of $\underline{J}(\vec{r})$ is invertible, so that:

$$\exists \underline{\underline{D}} \in M_n(\mathbb{R})/\underline{\underline{D}} = \frac{1}{3}\left(\underline{\underline{\Sigma}}^1(\vec{r}) - \underline{\underline{\Sigma}}_{s1}(\vec{r})\right)^{-1} \qquad (2\text{-}174)$$

We get:

$$\underline{\underline{D}}^{-1} = 3 \times \begin{pmatrix} \Sigma_1^1 - \Sigma_{s1,1\leftarrow 1} & \Sigma_{s1,1\leftarrow 2} & \dots & \Sigma_{s1,1\leftarrow G} \\ \Sigma_{s1,2\leftarrow 1} & \Sigma_2^1 - \Sigma_{s1,2\leftarrow 2} & \ddots & \Sigma_{s1,2\leftarrow G} \\ \vdots & \vdots & \ddots & \vdots \\ \Sigma_{s1,G\leftarrow 1} & \Sigma_{s1,G\leftarrow 2} & \dots & \Sigma_G^1 - \Sigma_{s1,G\leftarrow G} \end{pmatrix} \qquad (2\text{-}175)$$

A classic diffusion law implies that $\underline{\underline{D}}$ is diagonal, and therefore $\underline{\underline{D}}^{-1}$ is as well. This case is achieved if and only if we consider isotropic scattering because then $\underline{\underline{\Sigma}}_{s1} = \underline{\underline{0}}$. Otherwise, there is no equivalence between the P_1 method and the diffusion method. Indeed, we cannot be sure that the diffusion coefficients D_g of each group are equal to $\frac{1}{3}\left(\Sigma_g^1 - \Sigma_{s1,g\leftarrow g}\right)$, which would be equivalent to the assumption that the off-diagonal terms of the previous Matrix $\underline{\underline{D}}^{-1}$ are equal to 0.

If we use the transport correction to calculate $\underline{\underline{\Sigma}}_{s1}$, then we have:

$$\Sigma_{s1,g\leftarrow h} = \overline{\mu}_0\Sigma_{s0,g\leftarrow h} \text{ with } \overline{\mu}_0 = \frac{2}{3A}constant;$$

$$\Sigma_{tr,g\leftarrow h} = \left(1 - \overline{\mu}_0\right)\Sigma_{s0,g\leftarrow h}; \Sigma_{r,g} = \Sigma_g - \Sigma_{s0,g\leftarrow g} \qquad (2\text{-}176)$$

$$\underline{\underline{D}}^{-1} = 3 \times \begin{pmatrix} \Sigma_{r,1} + \Sigma_{tr,1\leftarrow 1} & -\mu_0\Sigma_{s0,1\leftarrow 2} & \dots & -\mu_0\Sigma_{s0,1\leftarrow G} \\ -\mu_0\Sigma_{s0,2\leftarrow 1} & \Sigma_{r,2} + \Sigma_{tr,2\leftarrow 2} & \dots & -\mu_0\Sigma_{s0,2\leftarrow G} \\ \vdots & \vdots & \ddots & \vdots \\ -\mu_0\Sigma_{s0,G\leftarrow 1} & -\mu_0\Sigma_{s0,G\leftarrow 2} & \dots & \Sigma_{r,G} + \Sigma_{tr,G\leftarrow G} \end{pmatrix} \qquad (2\text{-}177)$$

The problem remains the same—there is no formal equivalence with diffusion theory because $\underline{\underline{D}}$ is not diagonal. However, now there may be a way to diagonalize it because the linear anisotropic terms are proportional to the isotropic terms, so that we may arrive at Fick's Law.[4] However, this has to be shown.

In addition, even if this were feasible, then the expression of the new matrix with the formalism of the first matrix may not be totally trivial.

A common procedure to try to mimic a diffusion code with a P_1 solver is described hereafter, as proposed in the INSTANT manual[1]:

$$\forall g \in [\![1, G]\!]: \tag{2-178}$$

$$\tilde{\Sigma}_{total,g} = \frac{1}{3D_g} - \Sigma_{r,g} + \Sigma_{tr,g} \tag{2-179}$$

$$\tilde{\Sigma}_{s1,g \leftarrow h} = 0 \ \forall h \in [\![1, G]\!] \ and \ h \neq g. \tag{2-180}$$

$$\tilde{\Sigma}_{s0,g \leftarrow g} = \Sigma_{tr,g} = \frac{1}{3D_g} - \Sigma_{r,g} \tag{2-181}$$

$$\tilde{\Sigma}_{s0,g \leftarrow h} = \Sigma_{s0,g \leftarrow h} \ \forall h \in [\![1, G]\!] \ and \ h \neq g. \tag{2-182}$$

with the assumption that we know the diffusion coefficients, the transport, and absorption cross sections in each group.

It leads to the computation of the following matrix in the P_1 code:

$$\underline{\underline{\tilde{D}}}^{-1} = 3 \times \begin{pmatrix} \Sigma_{r,1} + \Sigma_{tr,1} & 0 & \cdots & 0 \\ 0 & \Sigma_{r,2} + \Sigma_{tr,2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \Sigma_{r,G} + \Sigma_{tr,G} \end{pmatrix} \tag{2-183}$$

Such an approximation will miss the off-diagonal terms by assuming them to be equal to 0, and therefore calculate a wrong Matrix $\underline{D}$. Therefore, it will not be a pure diffusion calculation, and it will also not be a correct P_1 calculation because we do not consider:

- Pure isotropic scattering: the diagonal terms contain a part of the linear anisotropic scattering information through the transport cross sections $\Sigma_{tr,g}$

- A true linear anisotropic scattering source with a transport correction to calculate those terms because we do not take into account the off-diagonal terms of $\underline{\Sigma_{s1}}$

- A linear anisotropic scattering that would not be calculated through the transport correction.

In conclusion, there is no easy way to mimic a diffusion code with a P_1 solver in a multi-group situation unless we consider pure isotropic scattering or find a way to analytically diagonalize the transport corrected $\underline{\Sigma_{s1}}$ and then find the correct combination of terms to replace in the P_1 solver.

However, in the case of graphite, we may assume[4]:

- Anisotropic terms are small compared to the isotropic terms. Indeed, the graphite atom is much heavier than the neutron, which tends to create nearly isotropic scattering. Anisotropic terms of an order higher than 1 may be easily neglected, and the transport correction probably gives a good representation of the linear anisotropy.

- Off-diagonal terms of $\underline{\Sigma_{s1}}$ are small compared to the diagonal terms. This may be especially true with fewer energy groups because the graphite is not a very good moderator.[3] It means that after a collision, the loss of lethargy of the neutron is small so that the neutrons have more chances to stay in the same energy group than to switch to another group. The fewer groups we have, the larger they are, so that this approximation becomes better.

In this case, we can:

- Keep an anisotropy of Order 1 for the neutron source using the transport correction

- Neglect the off-diagonal terms of $\underline{\Sigma_{s1}}$ in the calculation.

With this set of assumptions, we arrive at Fick's Law and we can use the equivalence from Equation 2-182 to mimic the results of a diffusion code.[1]

The other choice is to let the code perform the normal P_1 calculation with a linear anisotropic source provided by the $\underline{\Sigma_{s1}}$ evaluation data processed in DRAGON. This solution was finally chosen in the report, as shown later.

2.12.4   The P_3 Equations

The development in spherical harmonics of the flux can be extended to higher orders. In INSTANT, the P_3 method was used, which means that we calculated the flux with an approximation of Order 3. The scattering source may have a different expansion. For example, a linear anisotropic source (Order 1) was used. In this case, the transport correction was not used and the $\underline{\Sigma_{s1}}$ given by DRAGON were preferred.

This matrix comes directly from the evaluation. DRAGON is only producing homogenized, condensed matrices using the flux calculated on the lattice level. This procedure may lead to better results than the P_1 because a higher order both for the source and the flux calculation is being used.

On the contrary, for example, it is not trivial to predict which calculation would be the best between a P_1, transport corrected solution and a P_3 solution with an isotropic source.

2.13 The Monte Carlo Method: A Different Philosophy

The Monte Carlo method uses a completely different philosophy. The idea is to simulate the random walk of a neutron from its creation (by fission or [n,xn] reactions) to its death (by absorption or leakage) in the core.[2] Records are kept of every reaction that the neutron encounters during its life. Millions of simulations enable one to produce reaction rates with statistical meaning. The statistical uncertainty decreases proportional to $\frac{1}{\sqrt{N}}$ where N is the number of neutrons simulated. Therefore, the convergence is low, and such a method requires a great deal of computation time. However, this method is a perfect candidate for parallelization because in theory we can assign one neutron per processor—as stated at the beginning of the section, we neglect the neutron-neutron interactions.

The Monte Carlo method has many advantages. For example, any geometry can be defined and we can use a continuous energy structure so that the self-shielding step disappears from the calculation. The method is meant to be exact, as far as the cross sections are correct and the number of simulated particles is sufficient. Thus, the Monte Carlo codes are often used to validate deterministic codes. The method is also used in other applications like detector studies, criticality studies, medical studies, fusion studies, and especially when experimental results are not available. It faces its own problems (e.g., statistical instabilities).

Several codes have been developed to implement the Monte Carlo method for reactor physics problems. MCNP is a well-known method developed at the Los Alamos National Laboratory. Its fifth version served as a reference for this study.

SERPENT is another stochastic code developed by Jaako Leppänen for VTT (Valtion Teknillinen Tutkimuskeskus, Technical Research Centre of Finland), which was designed for the generation of cross sections.[17] It is the first example of a stochastic code, which was built for operational purposes. Usually calculations related to design or reactor operations are carried out with deterministic codes. With the

increase of calculation speed and massive parallelization, stochastic production codes like SERPENT may become the future of reactor analysis.

2.14 Isotopic Depletion

When we operate a reactor, the material inside the core is changing due to two phenomena[2]:

- The nuclear reactions resulting from collisions between neutrons and nuclei

- The radioactive decay of some nuclei.

In the case of the nuclear reactions produced by exposure to neutron flux, the variation of the number density is proportional to the reaction rate, that is to say the probability to have an absorption reaction:

$$\frac{dN}{dt}(t) = -\langle \sigma_a \phi(t) \rangle N(t) \tag{2-184}$$

with

$$\langle \sigma_a \phi(t) \rangle = \int_0^{+\infty} \sigma_a(u)\phi(u,t)du. \tag{2-185}$$

This equation does not depend on which mode of decay the compound nucleus will follow: fission, radiative capture, etc. The only important information is that we know that this reaction will not produce the initial nucleus. All reactions are taken into account except the elastic and inelastic scattering reactions:

$$\sigma_a = \sigma_{total} - \sigma_e - \sigma_{in}. \tag{2-186}$$

In the case of the radioactive decay of an isotope, the number density of a radioactive nucleus obeys to a simple differential equation:

$$\frac{dN}{dt}(t) = -\lambda N(t) \tag{2-187}$$

λ is the radioactive decay constant and is independent of time, but varies depending on the way of decay and the isotope considered.

In a reactor, nuclear reactions created by the flux and radioactive decay occur together on all isotopes at the same time, and an isotope can be produced by reactions or decays involving other isotopes. Therefore, we can write for each single nuclide k[4]:

$$\frac{dN_k}{dt}(t) = \phi \sum_{i=1}^{M} N_i \sigma_{f,i} \mathcal{Y}_{i,k} + \phi \sum_{s=1}^{M} N_s \sigma_{a,s} \gamma_{s,k} + \phi \sum_{j=1}^{M} N_j \lambda_j \alpha_{j,k} - \sigma_{a,k}\phi(t)N_k(t) - \lambda_k N_k(t) \tag{2-188}$$

where

N_k = number density of isotope k

ϕ = flux $\qquad\qquad$ (2-189)

$\sigma_{f,i}$ = fission cross-section of isotope i

$\sigma_{a,s}$ = absorption cross-section of isotope s

λ_k = decay constant of isotope k $\qquad\qquad$ (2-190)

$\mathcal{Y}_{i,k}$ = yield of isotope k due to a fisison of isotope i

$\gamma_{s,k}$ = probability that a neutron absorption in isotope i produces isotope k

$\alpha_{i,k}$ = probability that the decay of isotope j produces isotope k $\qquad\qquad$ (2-191)

A system is obtained containing an equation describing the evolution for each nuclide. Those equations are coupled between all nuclides. The information on decay channels and reaction yields is stored in the nuclear reaction chains and is available in the evaluations.

The problem with solving such a system is that the number of ways for a nuclide to evolve can be large, and it is highly coupled to the depletion of other nuclides. The time constants can also be very different, which may create numerical issues.[2]

In addition, the depletion affects the flux. Indeed, macroscopic cross sections have to be updated regarding the number density, but it also indirectly affects the microscopic cross sections, because the spectrum may change according to the new core configuration. However, those changes are relatively slow. A common assumption is made and consists of discretizing the time into time steps. The flux is then considered as constant during each time step, and then is recalculated. We always stay within the steady-state approximation.[2,16]

The calculation is usually performed at a constant specific power (power per mass of initial heavy isotopes). This power is used to scale the flux so that its magnitude acquires some sense. In this model, the power at the beginning and end of the time step is equal. It enables us to calculate the flux at the end of the time step. A few iterations are needed to correctly calculate the number densities at the end of the time step according to the specific power.

The burnup $B(t)$ measures a sort of average of the energy released by an assembly per initial mass of heavy isotopes.[2] It is usually expressed in MW day .tonne^{-1}. This value provides some information about the energy that has been extracted from an assembly during his life. Mathematically, we define it at time t_f tf by:

$$B(t) = \frac{V}{W} \int_0^{t_f} \langle H\phi(t') \rangle dt'. \tag{2-192}$$

Where V is the volume of the fuel and W is the volume of heavy isotopes at $t = 0$. H is a factor giving the recoverable energy from neutron-induced reactions.

Depletion calculations have to be performed at the lattice level to generate microscopic cross sections for the core. The cross sections are then tabulated with the burnup and passed to the core calculation. The number densities for the core may be recovered either from the lattice calculation, or less frequently from the core calculation if the solver contains an in-core depletion module. The principle of the depletion remains the same in the lattice as in the core.

3.   DESCRIPTION OF THE DIFFERENT WAYS TO PREPARE CROSS SECTIONS

3.1   General Considerations on the Calculation Schemes

3.1.1   Description of the Calculation Schemes

As stated in Section 2, a detailed full-core 3-D transport calculation would require a huge amount of memory and would take significant central processing unit (CPU) time to be used by utilities in production calculations or even in laboratories for design purposes. While trying to produce accurate cross sections for this reactor, we should keep in mind that utilities need to perform numerous core calculations to predict the fuel depletion during approximately 1 year of operation, verify that reactivity margins are respected during the whole operation cycle, and analyze many core configurations. Design studies also require several runs of core calculation to establish the behavior in transient conditions. Performing all of these studies with a detailed full-core calculation, even in 2-D, is currently unfeasible within a reasonable amount of time.

Therefore, the typical way to proceed is to use a two-step calculation scheme[2]:

- Detailed calculation at the assembly level with reflective boundary conditions, which gives homogenized cross sections for the assemblies, condensed to a certain number of groups; this step is called the lattice calculation.

- Second calculation at the core level with homogenized properties in each assembly and usually a small number of groups. We call it the full-core calculation or whole-core calculation.

This scheme is particularly effective in the case of LWRs because the typical mean free path of thermal neutrons in water is equal to few millimeters. Therefore, modeling the assemblies with a small layer of water around them is very effective at capturing the main physical effects. The coupling across the full core is low, which justifies this kind of approach. The assemblies in the full core exist largely in an infinite lattice condition with the exception of those at the periphery.

For high temperature reactors, the mean free-path of neutrons is much longer, and therefore the decoupling between the block[a] level and the full-core level is less accurate. Facing this problem, we keep a two-step calculation scheme because an alternative has not been well established. However, knowing that the spectrum will completely differ between the lattice and the core calculation, especially in the blocks adjacent to the reflector, particular care must be taken to provide cross sections to the core calculation. The purpose of this report is to determine which models can be used at the lattice level to obtain good accuracy at the core level. The lattice calculations will be performed using DRAGON, whereas INSTANT will be used for full-core calculations. The temperature remains set at 293.6 K, as the study of its influence is beyond the scope of the report.

At the lattice level, a single block of fuel may be quite representative for the central blocks located far from reflectors. However, the blocks located at the periphery are receiving a lot of thermal neutrons coming back from the reflector. For them, the single-block infinite lattice model may not be relevant due to the neutron energy spectral modification created by the reflector's presence. To deal with this specific feature of this reactor, it is assumed that keeping a large number of groups for the INSTANT calculation would compensate for the poor quality of the microscopic cross sections used to describe those blocks. With this point of view, we can still generate microscopic cross sections from a single-block model at the

---

a.   For high temperature reactors, the term "block" replaces "assembly."

lattice level if sufficient energy groups are retained for the whole core calculation. It will be the first calculation scheme evaluated, named "single-block path."

Secondly, it is proposed to treat the peripheral blocks separately with supercell models including one row of neighboring blocks around the block of interest at the lattice level. The other central blocks of the core remain calculated with the previous path. This method forms the second calculation scheme and is called "supercell path." Our assumption is that one may be able to catch some of the effects due to the surrounding graphite blocks, and have a calculation in INSTANT based on more accurate homogenized cross sections. In addition, a quite large number of groups may also correct the remaining errors of the cross sections passed to INSTANT. In this report, we aim to quantify the accuracy of such a calculation compared to the other path.

A full-core transport calculation would, in theory, provide the best solution to prepare cross sections for the core model as the assumptions on the decoupling of the fuel cells vanishes. However, this is not practical with currently available tools. DRAGON faces memory limitations due to addresses coded on 32 bits, and even with the fastest solver (SYBILT) and no discretization of the geometry, the calculation is cumbersome. As the use of EXCELT and the discretization has proved to be necessary on supercells, and given the fact that it was not possible to do it in DRAGON, this idea was not pursued. Note also that in general, most deterministic lattice codes were not designed to do this kind of calculation and few studies have been done to determine the precision of the codes with large models like that.

Moreover, if a full-core transport calculation was feasible within a reasonable time, then there would be no interest to generate homogenized cross sections for another core solver as the detailed solution of the problem is already known.

Passing the homogenized block cross sections from DRAGON to INSTANT is not trivial. INSTANT needs macroscopic cross sections for each block. DRAGON is able to produce either microscopic or macroscopic homogenized cross sections. However, the macroscopic cross sections are generated using the volumes upon which the homogenization is done. Therefore, if block sizes are different between the two codes, then the homogenized densities of atoms will differ between them, and the macroscopic homogenized cross sections computed by DRAGON will be not be suitable for INSTANT. As it happens, this is the case in our models because of the actual limitation of the tracking modules in DRAGON. Currently, it is impossible to model some hexagons embedded into a larger one with a straight boundary.[9] Therefore, we have to produce homogenized *microscopic* cross sections in DRAGON and then reconstruct the *macroscopic* cross sections for INSTANT using the correct number densities.

More details to circumvent this difficulty are given in Appendixes C, D, and in Section 3.4 when we detail the single-block model built in DRAGON. Notice that this point is of great importance. The block volumes between DRAGON and INSTANT differ by 8.26%, which leads to an error of the same amount on the value of all isotopes' densities if we do not take care of it. In the future, an improvement of the tracking module EXCELT of DRAGON to model the real geometry would simplify this part of the modelization and eliminate a geometrical approximation as well as a source of potential error by the analyst.

A program written in Fortran90 by Javier Ortensi (INL) enables us to recover the homogenized microscopic cross sections of every isotope used in a DRAGON calculation, and the corresponding densities provided by the user. The program then calculates the correct homogenized macroscopic cross-sections for INSTANT and creates this part of the INSTANT input file. The calculation of the correct homogenized densities in INSTANT has been performed and is given in Appendixes C and D. The difficult part of this calculation is to correctly deal with the volume packing fraction and the spherical geometry of the TRISO particles projected on a plane surface. The reader is encouraged to examine the details of this calculation to precisely understand the underlying problem.

The JEFF-3.1.0 cross section evaluation was used with the SHEM-295 group-structure in DRAGON for all our calculations. The SHEM group structure was developed by Alain Hébert as an adaptation of the 281-group Santamarina-Hfaiedh energy mesh[18,19] and of the 361-group energy mesh.[20] Its boundaries can be found in Hébert's 2009 article.[11] This library can be freely downloaded in several formats including the American Standard Code for Information Interchange (ASCII) format on the public repository (http://www.polymtl.ca/merlin/libraries.htm).

3.1.2   Reference Calculations: MCNP

MCNP models have been provided for every type of calculations by Michael Pope (INL). It is assumed that those models represent the best solutions at every step, considering the fact that no experimental data is available for this particular reactor.

Nevertheless, the double heterogeneity treatment for spherical TRISO particles in MCNP is questionable. It must be assembled from a 3-D lattice sized such that the correct packing fraction is achieved. Several possibilities may be used to create the lattice geometry of TRISO particles inside the pellet: squares, hexagons, hexagonal lattice with the corners being removed, etc. Even if we manage to get the same packing fraction, the geometrical differences introduce an uncertainty of about 100 pcm on the K_{eff}. This should be kept in mind in comparison of DRAGON to MCNP calculations.

At the lattice level, the multiplication factor (K_{eff}) obtained in MCNP and DRAGON will be compared. The error on the K_{eff} was the criterion to decide which models must be retained. This was used in choosing between the different tracking options in DRAGON.

At the core level, not only were the K_{eff} obtained in INSTANT and MCNP compared, but also the block fission rates. Indeed, this parameter is of very high interest because it will directly influence the distribution of power used for thermofluid calculations and also the flux used for depletion calculations. Therefore, a good prediction of the fission rate shape is essential; perhaps more than the K_{eff} prediction since in reality, this parameter is biased to meet the experimental results for each reactor.

A comparison of the detailed shapes of the flux or of the fission rates obtained with the two codes would also have been interesting, but MCNP is not able to produce this information easily. It is anticipated that a good agreement between the block fission rates may indicate that the detailed fission rates are not too far from each other. To give some basis for this assumption, we will draw the shape of the fission rates across the core along one radius and compare this to MCNP. Results have been provided for every calculation path.

Notice here that the calculation in MCNP has been performed with continuous energy cross sections and with much more detailed reactions than in the deterministic calculations. Thus, a true comparison with the same library is not possible. The same cross section evaluation between DRAGON and MCNP was used—JEFF-3.1.0. Unfortunately, the thermal scattering cross sections for the graphite from this evaluation were not available for MCNP at the laboratory. We therefore used the ENDFB-VII evaluation for those cross sections. This difference of certain cross sections surely introduces some bias in the comparisons and should be kept in mind.

3.2   Core Model: INSTANT

3.2.1   Model of the Core in INSTANT

Our core is symmetric, and so a one-sixth core model has been built to speed up the calculations. This core model is used for all calculation schemes, no matter the DRAGON model that prepares the cross sections. Thus there is no bias coming from different core models when we compare two methods of cross section preparation.

The one-sixth core is made up of 56 blocks of fuel and reflector. The main geometric dimensions are given in Table 3-1 and Figure 3-1 provides a drawing of the core.
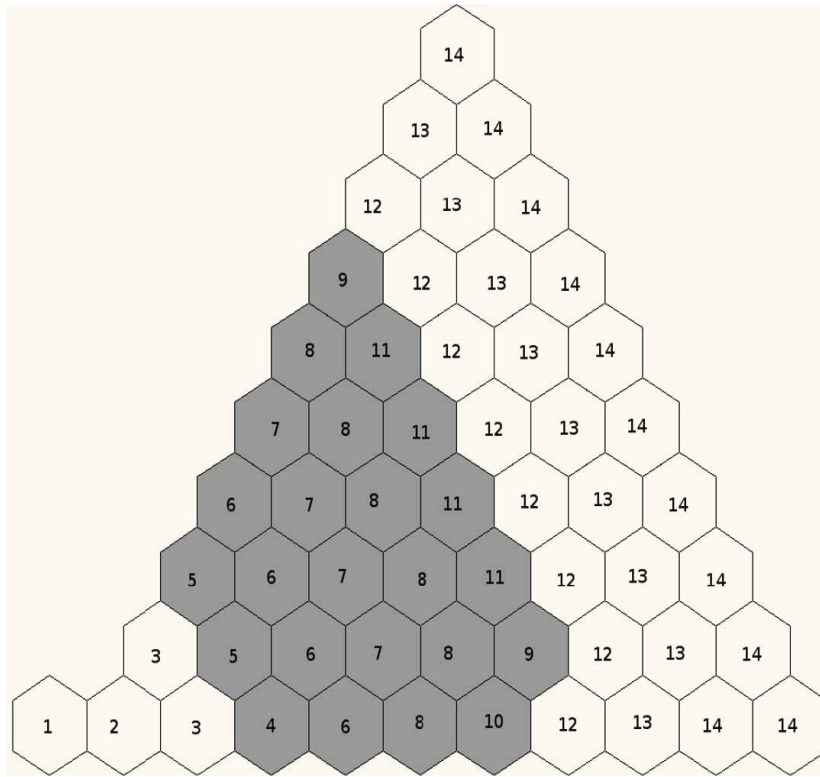
Figure 3-1. INSTANT one-sixth core geometry – mixtures.

Table 3-1. One-sixth core – geometrical parameters in INSTANT.

Parameter	Value
Block pitch (cm)	36.0
Block side (cm)	20.78461
N maximum number of hexagons in a direction	21
T total number of hexagons (one-sixth core)	56

The INSTANT model allows the provision of different cross sections in each row—14 different mixtures are defined. Mixtures 1, 2, and 3 correspond to the inner reflector, whereas Mixtures 12, 13, and 14 stand for the outer reflector. Those cross sections will be generated once and kept for every calculation to better isolate the impact of the different sets of fuel cross sections (Mixtures 4 to 11). Details about the generation of cross sections for the reflector will be provided later.

The code uses the hybrid finite element method (FEM) for the spatial discretization and the P_N method for the angular discretization.[21] The interior and interface shape functions used for the FEM are polynomials. The solver is governed by four parameters: the interior polynomial expansion order p_{int}, the interface expansion order p_{surf}, the P_N order p_n, and the source order expansion p_{sourc}, which controls the spatial expansion for the source calculation. Those parameters must respect the following rules in 2-D hexagonal geometry[1]:

$1 \leqslant p_n \leqslant 33$ and p_n odd number (3-1)

$p_{int} \geqslant p_{surf} + 3$

$0 \leqslant p_{sourc} \leqslant p_n$

The script to pass cross sections from DRAGON to INSTANT permits the use of only the isotropic component of the scattering source (scattering Order 0), or add a linear anisotropic component (scattering Order 1). In this work, the linear anisotropic part is typically used, but some calculations have been performed to determine the importance of this order. The scattering cross sections have been taken directly from the DRAGON output (SCAT00 and SCAT01). This way of handling the scattering is called the regular method in the following pages.

The script also permits the option of performing a calculation with the isotropic part of the scattering source, but with "tricked" cross sections so that one may reproduce a sort of diffusion calculation. As explained in Section 2.12.3, this way of treating cross sections is not completely equivalent to a diffusion calculation, but it provides indication on what it would give. An example is also given with this feature to assess by comparison with the regular transport calculation whether the core has a diffusive behavior.

3.2.2   Convergence Study of the Core Model

A convergence study has been performed to determine an acceptable set of parameters in the solver. For this purpose, cross sections were prepared by the first DRAGON model (single-block path). The K_{eff} has been taken as a convergence criterion. The effects of increasing p_{surf} order, p_{int} order, and p_{sourc} order were analyzed. The study has been done for P_1 and P_3 calculations. It will then be shown how well it is converged in angles (P_N parameter).

P_1 Convergence Study, Scattering Order 1, Regular Method

The behavior of p_{sourc} was first studied, setting $p_{surf} = 2$ and $p_{int} = 6$. Results are given in Table 3-2(a). It is concluded that the convergence is ensured with $p_{sourc} = 5$. This value is kept for the next study on p_{surf} and p_{int}, the results of which are collected in Table 3-2(b).

Table 3-2. Convergence study for the P1 core model.

(a) Study on p_{sourc}.

p_{sourc}	Core K_{eff}
0	1.20008
1	1.25369
2	1.25068
3	1.25054
4	1.25049
5	1.25046
6	1.25046

(b) Study on p_{surf} and p_{int}.

p_{int}	$p_{surf} = 2$ K_{eff}	$p_{surf} = 3$ K_{eff}	$p_{surf} = 4$ K_{eff}
5	1.25046		
6	1.25046	1.25046	
7	1.25047	1.25047	1.25047

This shows that these settings ensure convergence, thus $p_{surf} = 2$ and $p_{int} = 6$ were used because the convergence is easier to achieve with only 44 iterations.

P_3 Convergence Study, Scattering Order 1, Regular Method

The same study has been repeated with a P_3 calculation using exactly the same material information.

Table 3-3(a) gives the results of the convergence study on p_{sourc} with $p_{surf} = 2$ and $p_{int} = 6$. The convergence is ensured with $p_{sourc} = 5$ and this value is kept for the next study on p_{surf} and p_{int}

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

46   March 2011

(Table 3-3[b]). We finally keep $p_{surf} = 2$ and $p_{int} = 6$ because the convergence is easier to achieve with those values (95 iterations, see Table 3-3[b]).

In both cases, the default refinement of the mesh proved to be sufficient to reach convergence. Further discretization does not affect the solution.

Table 3-3. Convergence study for the P_3 core model.

(a) Study on p_{sourc}.		(b) Study on p_{surf} and p_{int}.				
p_{sourc}	Core K_{eff}			$p_{surf} = 2$ Core K_{eff}	$p_{surf} = 3$ Core K_{eff}	$p_{surf} = 4$ Core K_{eff}
0	1.20132		p_{int}			
1	1.25349		5	1.25024		
2	1.25046		6	1.25023 (95 it.)	1.25023 (partial convergence)	
3	1.25030		7	1.25023	1.25023	1.25023 (1439 it.)
4	1.25026					
5	1.25023					
6	1.25023					

P_5 Calculation, Scattering Order 1, Normal Method

Finally, a P_5 calculation was performed with those settings. This produces the same K_{eff} as with a P_3 calculation with less than 1 pcm of difference. Those results are given in Appendix E. Therefore, it is concluded that a P_3 calculation gives a completely converged model and it is thus kept for the remainder of the study. Some examples in P_1 will be provided in Appendix E to show the variability of the core calculation according to this parameter.

In summary, the following settings were used in INSTANT:

- P_3 calculation

- $p_{source} = 5$; $p_{surf} = 2$; $p_{int} = 6$

- Default mesh refinement.

3.2.3   Scattering Order and Diffusion Imitation

A calculation was also performed using the same cross sections, but with a scattering order reduced to 0 (isotropic scattering) in 26 groups. The following is obtained: $K_{eff\,P3\,scat0} = 1.25258$, compared to $K_{eff\,P3\,scat0} = 1.25023$. Both have been done in P_3. With 188 pcm of difference, it is concluded that the importance of the linearly anisotropic part of the scattering is small, but not totally negligible.

Another trial was performed in P_1 by "tricking" cross sections with the transport correction so that it approaches the results of a diffusion code. This resulted in $K_{eff\,P3\,scat0} = 1.25056$. This is very close to $K_{eff\,P3\,scat0} = 1.25023$, so it is concluded the core is rather diffusive. The graphite moderator is known to have a rather isotropic behavior,[4] so this result is expected.

3.3   Single Hexagonal Fuel Cell: DRAGON, SERPENT and MCNP

3.3.1   Effect of the Boundary Condition, Homogeneous Cell

Hexagonal Cell

A single hexagonal cell containing a homogeneous fuel pellet was first simulated with DRAGON and SERPENT. The difference of modeling is about the boundary conditions.

This study was performed because DRAGON is only able to set a white boundary condition, whereas MCNP can use either a white boundary condition or a true specular reflective condition. It is admitted that a reflective boundary condition is more representative of the reality than a white condition. Therefore, this study is used to assess the importance of the type of boundary condition on the simulation, even if DRAGON is not offering any choice for hexagonal geometries.

The Monte-Carlo code SERPENT was used to compare to DRAGON because it has a double-heterogeneity module allowing one to better isolate the effect of the boundary condition. This would not have been the case with MCNP where the TRISO treatment may be sometimes questionable.

The SERPENT simulation was performed by Nicolas Martin (École Polytechnique de Montréal) with a library based on the JEFF-3.1.1 evaluation. The following result was obtained:

$$K_{eff \ SERPENT \ hexcell} = 0.97858 \pm 22 \ pcm. \tag{3-2}$$

This serves as the reference. The simulation was performed with a reflective condition at the boundary.

In DRAGON, only a white reflective boundary condition can be used for hexagonal geometries. The calculation is done with the MOC technique. The basic calculation uses two iterations in the self-shielding, with correlation between the fissile isotopes and a transport correction. Table 3-4 details the results obtained for different parameters.

Table 3-4. Hexagonal cell—different set of parameters.

	Parameter		
Angles	TISO dens	USS/Geometry	K_{eff}
12	20	PASS 3	0.97585
12	20	NOTR	0.97585
12	20	NOCO	0.97586
12	20	No sect, 4 radii	0.97588
12	30	No sect, 7 radii	0.97551
12	40	No sect, 7 radii in fuel, 2 radii in graphite	0.97557

We remark that the results are located between 0.97551 and 0.97588. It means about 300 pcm of difference compared to the SERPENT reference, and indicates that the different boundary condition seems to have a big effect. However, it is also admitted that the larger the lattice is, the smaller is the difference between the two types of boundary conditions. Therefore, a larger lattice was created in DRAGON with one-twelfth symmetry, and with a central cell and eight rings of fuel cell. There are 25 cells in the assembly or 217 if we had taken the complete geometry. With this lattice, and TISO 12 20,0, we obtain:

$$K_{eff \ DRAGON \ hexcell} = 0.97854. \tag{3-3}$$

This is 4 pcm less than SERPENT.

In conclusion, this study shows a very good agreement between the white and the reflective conditions for a large domain, whereas a white condition on a single hexagonal cell leads to an error of 300 pcm down compared to SERPENT. It therefore demonstrates that with large lattices, the two conditions have quite a similar effect. As our block model will have 11 rings of cells, a white boundary condition may be sufficient to perform the calculation. A white boundary condition is set in MCNP, too, to remain consistent in comparisons.

Square Pin Cell

A second study of the boundary conditions was performed in DRAGON on square cells to assess this effect using the same code. The hexagonal cell was changed into a square, so that a cyclic tracking can be performed in DRAGON. The square had a side of 1.7495 cm, so as to conserve the same area between the two DRAGON models.

With tracking parameters 16 60.0 and a reflective boundary condition in DRAGON, $K_{eff\,DRAGON} = 0.97978$ is obtained and the maximal error on the volumes is equal to 0.8%. Table 3-5 details some other choices for the parameters. It is shown that the model is converged.

Table 3-5. Square cell: different set of parameters.

Parameter			
TISO		No. of Radii	
Angles	dens	in the Fuel Pellet	K_{eff}
12	20	4 radii	0.98024
12	30	4 radii	0.97982
12	40	7 radii	0.97995
12	50	6 radii	0.97977
12	60	6 radii	0.97978

The SERPENT calculation for this case gives $K_{eff\,SERPENT} = 0.97916 \pm 44$ pcm. Another Monte Carlo calculation in DRAGON using the new stochastic module Monte Carlo of DRAGON has also been performed by Nicolas Martin. He was thus able to use the same library as in the deterministic calculation, and the same self-shielding parameters. We get:

- $K_{eff} = 0.97942 \pm 55$ pcm with a correlated self-shielding

- $K_{eff} = 0.97970 \pm 55$ pcm without correlation in the self-shielding.

We observe that the K_{eff} are all located in the same range of ± 30 pcm.

On the contrary, a white condition would introduce a bias of about 300 pcm. In DRAGON, a tracking with 16 60 and a white boundary condition gives $K_{eff} = 0.97615$.

We conclude that in both geometries, for a single cell, the type of boundary condition is important for the K_{eff} value. A white condition will introduce a bias of 300 pcm on small domains compared to a reflective condition. However, as shown in the previous study, the increase of the domain leads to the convergence of the results using the two different boundary conditions. Even if the cyclic tracking has not yet been coded into DRAGON for hexagonal cells, it can be assumed that the result would be similar in this case.

3.3.2   Effect of the Surrounding Graphite on the Energy Self-shielding

Finally, a study was performed to evaluate the effect of the graphite number density in the region surrounding the pellet on the energy self-shielding calculation. This is of interest because self-shielding calculation will not be performed on the supercell models. For these, the self-shielding calculation will be performed on a block without its environment in a pre-calculation before passing homogenized cross sections to the supercell calculation in DRAGON. Therefore, it is of interest to know if this can be a significant source of error compared to MCNP where the self-shielding is implicitly accounted for the entire domain. As deep burn fuel is abundant in resonant absorbers, the energy self-shielding may depend on the fuel block environment.

The difficult part of this study is that the effect of having more graphite outside the fuel pellet has to be studied only on the energy self-shielding calculation. It would be difficult to study cross sections in 295 groups; therefore, a condensation is needed, but this operation requires a flux. However, an increase of the graphite density will produce a different flux calculation, which would prevent one to isolate the self-shielding effect. This is why something more special has to be performed.

Here is the principle of the study:

- Perform a complete calculation on a single cell

- Perform the self-shielding calculation

- Perform the flux calculation

- Produce the homogenized and condensed cross sections

- Save this flux calculation

- Hold the homogenized and condensed cross sections as a reference.

Then the same geometry is taken but the graphite number density is increased by a factor of 100 outside the fuel compact. The self-shielding is recalculated, but not the flux calculation. Instead, the flux previously saved is called back and used to homogenize the cross sections. Thus, those cross sections are self-shielded differently, but they are homogenized and condensed by the same weighting function (i.e., the flux). One is then able to compare them in six groups. The six-group structure is given in Appendix F, "Group Structures." If the self-shielding has changed, then it should appear in the cross sections.

Table 3-6 gives the cross sections obtained with a normal regular graphite density in the region surrounding the fuel pellet whereas Table 3-7 gives the same cross sections when the graphite number density has been multiplied by 100. It is shown that there are almost no differences. The maximum difference is of 0.7% of the regular cross section on the absorption cross section in Group 2.

Table 3-6. Single cell – homogenized cross sections with a normal graphite density around the fuel pellet.

| | | Homogenized Cross Sections – Normal Density | | | |
Group	Total XS	Absorption XS	$\nu\Sigma_{fission}$	Scattering XS within Group	Scattering XS out of Group
1	2.4540E-01	2.0650E-04	4.8208E-04	2.2969E-01	1.5501E-02
2	3.9499E-01	1.8231E-03	2.1290E-03	3.8701E-01	6.1570E-03
3	4.0725E-01	8.9779E-03	1.1225E-02	3.6992E-01	2.8350E-02
4	4.5111E-01	4.8804E-02	5.4703E-03	3.6158E-01	4.0726E-02
5	4.7317E-01	7.3240E-02	1.1485E-01	3.8322E-01	1.6712E-02
6	4.6411E-01	5.5374E-02	9.7179E-02	4.0043E-01	8.3055E-03

Table 3-7. Single cell – homogenized cross sections with an increased graphite density by 100 around the fuel pellet.

| | | Homogenized Cross Sections – 100 × Normal Density | | | |
Group	Total XS	Absorption XS	$\nu\Sigma_{fission}$	Scattering XS within Group	Scattering XS out of Group
1	2.4540E-01	2.0650E-04	4.8208E-04	2.2969E-01	1.5501E-02
2	**3.9501E-01**	**1.8362E-03**	**2.1359E-03**	**3.8702E-01**	6.1570E-03
3	4.0725E-01	8.9779E-03	1.1225E-02	3.6992E-01	2.8350E-02
4	4.5111E-01	4.8804E-02	5.4703E-03	3.6158E-01	4.0726E-02
5	4.7317E-01	7.3240E-02	1.1485E-01	3.8322E-01	1.6712E-02
6	4.6411E-01	5.5374E-02	9.7179E-02	4.0043E-01	8.3055E-03

It is concluded that the self-shielding is not really affected by the surrounding graphite density. This result was quite unexpected, because the neutron slowing-down may have been modified by the graphite, but it seems that it is not the case. The energies where the self-shielding is taking place may be too high to see the influence of the graphite.

3.4 Single Fuel Block Model: DRAGON

The first scheme of calculation begins with the modeling of a single block of fuel in DRAGON.

In MCNP, the calculation over the real block was performed and is held as a reference. The following result was obtained:

$K_{eff\ SB\ MCNP} = 1.25969 \pm 0.00012.$ (3-4)

3.4.1 Description of a Fuel Block and Boundary Approximation

The real fuel block is comprised of 11 rows of hexagons, plus the central hexagonal cell. Figure 1-2 gives a representation of the geometry. Table 1-2(a) gives the dimensions considered.

Special attention must be paid to the boundary cells. Indeed, DRAGON does not have the capability to include a portion of a hexagonal lattice into larger hexagon. Thus, it is necessary to terminate the lattice by a row of hexagons with an adjusted graphite density. This density has been determined to conserve the total number of graphite atoms in the whole core calculation (see Appendix C). This solution has been also used by other authors confronted by the same problem[22,23,24,25] and it appears to be accurate.

In DRAGON, a homogenization of the whole cell is performed and the resulting microscopic cross sections are used for the fuel mixture used in INSTANT. The densities are corrected to fit with the volume of a block in INSTANT (see Appendix D, "Density Calculation for all Isotopes in INSTANT").

3.4.2   Spatial Convergence Study of the Single-block Model

DRAGON enables the user to set many different parameters. In particular, it is necessary to choose the level of discretization to ensure that the mesh is sufficient for this problem. Two tracking modules were tested: EXCELT and SYBILT.[9]

EXCELT Tracking and MOC

A first possibility is to use the MOC to solve the Boltzmann equation. This is done in DRAGON with the EXCELT tracking module followed by the module MCCGT, which converts the tracks for the MOC solver.

A convergence study over all the parameters together would require a huge amount of calculations. But the parameters seem to be partially decoupled, so that they can be independently studied. By setting relative high values for the parameters that are fixed while others are varied, convergence criteria can be met. For the double-heterogeneity model, the Hébert model was be used for the whole study.[26]

We first attempt to find whether a discretization of the fuel pellet is needed. The annular region of helium was not studied because helium is basically neutron transparent. The number of radii in Table 3-8(a) corresponds to the number of radii inside the fuel region. One radius means that this region is not discretised. For those calculations, relative high values were set for generating the tracking lines: four angles and 40 lines per cm. It was concluded from these results that there is no need to discretize the fuel pellet, but the division of the hexagonal fuel cell into six sectors counts for 10 pcm and will be kept in further calculations. The division of hexagons into sectors is called sectorization or azimuthal sectorization in the remainder of the report. It is abbreviated by "Azimuth. sect." in the tables.

A study was then performed to determine the influence of the discretization in the graphite and coolant regions. For this, the fuel pellets remain sectorized. The parameters TISO 4 40.0 were kept for the tracking. Results are shown in Table 3-8(b).

Table 3-8(b) shows that neither the discretization nor the sectorization of the graphite regions are significantly changing the K$_{eff}$. It was concluded that discretization of the graphite cells is unnecessary. But as these results were not complete at the beginning of the study, the next calculations were performed with sectorized graphite cells containing one radius.

A study over the tracking parameters was completed to determine the minimum that could be used with acceptable accuracy. Table 3-9(a) shows the density of lines that has to be used and Table 3-9(b) considers the number of angles.

It can be concluded that TISO 4 10 creates enough integration lines to cover the whole geometry and produce an efficient integration. The parameters TISO 4 10 were used for all the proceeding calculations. This allows the tracking to place six segments in the fuel pellet, which appears to be enough.

Table 3-8. Convergence study for the single block: geometrical mesh.

(a) Fuel cell radial discretization.

Cell Type	Number of Radii	Azimuth. Sector.	K-eff
Fuel	4	Yes	
Graphite	1	Yes	1.25919
Coolant	0	No	
Fuel	3	Yes	
Graphite	1	Yes	1.25918
Coolant	0	No	
Fuel	2	Yes	
Graphite	1	Yes	1.25915
Coolant	0	No	
Fuel	1	Yes	
Graphite	1	Yes	1.25911
Coolant	0	No	
Fuel	1	No	
Graphite	1	Yes	1.25902
Coolant	0	No	

(b) Graphite cell radial discretization.

Cell Type	Number of Radii	Azimuth. Sector.	K-eff
Fuel	1	Yes	
Graphite	1	Yes	1.25913
Coolant	1	Yes	
Fuel	1	Yes	
Graphite	0	No	1.25908
Coolant	0	No	
Fuel	1	Yes	
Graphite	1	No	1.25908
Coolant	0	No	

Table 3-9. Convergence study for the single block: tracking parameters.

(a) Density of lines.

TISO		
Number of Angles	Density (lines.cm^{-1})	K-eff
4	100	1.25911
4	50	1.25911
4	40	1.25911
4	30	1.25911
4	20	1.25913
4	10	1.25918

(b) Number of angles.

TISO		
Number of Angles	Density (lines.cm^{-1})	K-eff
12	30	1.25911
10	30	1.25928
8	30	1.25928
6	30	1.25827
4	30	1.25911

The QUAB parameter defines the number of points used to calculate the integrals inside the TRISO particles. The study's results are summarized in Table 3-10(a). It is concluded that a 5-point basis for the integration in the TRISO particles works well. A Gauss quadrature is always used because it ensures the best accuracy for the numerical integration.

Table 3-10. Convergence study for the single block: integration points in the TRISOs and self-shielding parameters.

(a) Integration points in TRISOs.		(b) Self-shielding parameters.			
		Parameters for the Self-shielding			
QUAB	K-eff	PT Accuracy	Number of External Iterations	Correlation	K-eff
7	1.25919				
6	1.25919				
5	1.25918	3	2	Yes	1.25918
4	1.25911	3	3	Yes	1.25918
3	1.25874	2	2	Yes	1.25918
2	1.25107	3	2	No	1.25914

Finally, several parameters were evaluated in the self-shielding module USS, for example the number of external iterations, the accuracy of the probability tables and the correlation of the isotopes. The variation of K_{eff} was used as a figure of merit for the convergence. Table 3-10(b) gives the results.

This shows that having three and two external iterations gives a good representation of the self-shielding. It also shows that the correlation effect between the fissile isotopes is low. This does not automatically imply that all isotopes are independent, but that the coupling effect is small.

In conclusion, the model chosen for an EXCELT tracking of the single-block model is made of:

- Geometry: sectorization of the fuel pellet without any radial discretization, no discretization nor sectorization of the graphite and coolant cells

- Tracking: TISO 4 10, QUAB 5

- Self-shielding: two iterations (PASS 2), probability tables with an accuracy of three, no correlation between fissile isotopes during the self-shielding.

  With this model, the following K_{eff} is obtained:

K_{eff} SB MOC = 1.25911.   (3-5)

Compared to MCNP, it is −46 pcm lower. Thus, the agreement between the two codes is very good.

SYBILT Tracking and Collision Probabilities Method

The other module available to do the calculation in hexagonal geometry is called SYBILT and uses the CP method with interface currents coupling. The module is very fast but less reliable than EXCELT because it uses more approximations.

In SYBILT, parameters must be set for the geometrical mesh, the integration lines and the number of self-shielding iterations. The technique of interface current method was used with a P_1 expansion (option DP01 in DRAGON).

A convergence study was first performed over the geometric specifications of the mesh: radial discretisation and sectorization of cells. For this study, three iterations were used for the self-shielding. The integration line parameters were set to: 8 angles, 10 segments per line (QUA2 8 10), and 5 segments in TRISO particles (QUAB 5). Table 3-11(a) contains the results for the discretization of the fuel cells and Table 3-11(b) contains the same information for the graphite and coolant cells. Table 3-11(b) gives also the results of different trials to determine whether the graphite and coolant cells had to be sectorized.

It was concluded that the geometry is sufficiently discretized when there are 3 radii in the fuel pellet, no sectorization, and no radial discretisation in the coolant cells and graphite cells. However, these studies had not been completed at the outset, so the following parameters were used for further convergence studies: 3 radii and sectorization in the fuel pellet, no sectorization in the other cells, and 1 radius with sectorization in the graphite cells.

The next study was used to determine the parameters for the integration lines that will be adapted to the geometry. Those parameters are called QUA2 and QUAB in DRAGON. QUA2 refers to the macroscopic geometry, whereas QUAB refers to the double-heterogeneity model. Three iterations were kept for the self-shielding. The results are given in Table 3-12. It shows that QUA2 4 8 is a good combination.

A study over the self-shielding iterations had also been performed. The parameters were set to QUA2 8 10 and QUAB 5, to be fully converged at the geometrical level. Table 3-13(a) shows the results for different options. This shows that the self-shielding iterations are totally converged after two iterations. The effect of the correlation between the fissile isotopes was also studied. Until now, it has been assumed that there was a mutual self-shielding between all of the resonant isotopes that had to be taken into account. On the contrary, assuming that all isotopes can be self-shielded independently may improve the calculation time, but it must be verified that it does not affect K_{eff} (see Table 3-13[b]).

Table 3-11. Convergence study for the single block: geometrical mesh.

(a) Fuel cell radial discretization.

Cell Type	Parameters Number of Radii	Parameters Azimuth. Sector.	K-eff
Fuel	9	Yes	
Graphite	1	Yes	1.26052
Coolant	1	Yes	
Fuel	7	Yes	
Graphite	6	Yes	1.26052
Coolant	1	Yes	
Fuel	4	Yes	
Graphite	6	Yes	1.26052
Coolant	1	Yes	
Fuel	3	Yes	
Graphite	6	Yes	1.26052
Coolant	1	Yes	
Fuel	2	Yes	
Graphite	6	Yes	1.26052
Coolant	1	Yes	
Fuel	1	Yes	
Graphite	6	Yes	1.26052
Coolant	1	Yes	

(b) Graphite cell radial discretization, coolant and graphite cells sectorization.

Cell Type	Parameters Number of Radii	Parameters Azimuth. Sector.	K-eff
Fuel	4	Yes	
Graphite	6	Yes	1.26050
Coolant	1	Yes	
Fuel	4	Yes	
Graphite	4	Yes	1.26049
Coolant	1	Yes	
Fuel	4	Yes	
Graphite	2	Yes	1.26049
Coolant	1	Yes	
Fuel	4	Yes	
Graphite	1	Yes	1.26049
Coolant	1	Yes	
Fuel	4	Yes	
Graphite	0	None	1.26047
Coolant	1	Yes	
Fuel	4	Yes	
Graphite	1	Yes	1.26048
Coolant	1	None	
Fuel	4	Yes	
Graphite	0	None	1.26046
Coolant	1	None	

Table 3-12. Convergence study for the single block: integration lines.

Parameters			
QUA2			
No. Angles	No. Segments	QUAB	K-eff
8	10	5	1.26047
6	10	5	1.26047
4	10	5	1.26046
3	10	5	1.26052
2	10	5	1.26020
8	10	5	1.26047
8	8	5	1.26046
8	6	5	1.26044
8	4	5	1.26042
8	3	5	1.26040
8	10	8	1.26047
8	10	7	1.26047
8	10	5	1.26047
8	10	3	1.26003

Table 3-13. Convergence study for the single block: self-shielding.

(a) Number of self-shielding iterations Parameters.		(b) Self-shielding correlation Parameter.	
Number of Self-shielding Iterations	K-eff	Correlation	K-eff
5	1.26047	Yes	1.26043
4	1.26047	No	1.26039
3	1.26047		
2	1.26047		
1	1.26046		

In conclusion, the following model was conserved for all single-block calculations with SYBILT:

- Geometrical discretization:

    - Fuel cells (F): 3 radii inside the pellet (0.0 0.207 0.415 0.6225) and sectorization.
    - Coolant cells (K, C): no radial discretization, no sectorization.
    - Graphite cells (G, J): no radial discretization, no sectorization.

- Tracking:

    - Four basis points for the angular integration of the blocks.
    - Eight basis points for the spatial integration of the blocks.

- Five basis points for the numerical integration of the collision probabilities in the micro-volumes using the Gauss-Jacobi formula. Hébert double-heterogenity model is used (QUA2 4 8 QUAB 5 HEBE).

- Self-shielding: two external flux iterations for the self-shielding calculation, and handle all fissile isotopes as non-correlated.

- Iterative techniques were used to compute the self-shielding and the collision probabilities. With those parameters, the following multiplication factor is obtained for the single block:

$$K_{eff\ SB\ SYB} = 1, 26039.$$ (3-6)

It is +56 pcm higher than the MCNP result. The agreement is comparable to the MOC (EXCELT).

Conclusion: Summary for the Single-block Calculations:

Table 3-14 summarizes the results obtained on the single-block model with the chosen parameters for the two trackings.

The single-block model is used to produce homogenized cross sections over the whole block, condensed to different numbers of energy groups. It is also used to produce cross sections for other DRAGON calculations which avoid some details (e.g., in the supercell calculations, for which cross sections are partially homogenized, or for the reflector calculation). In those cases, the EXCELT tracking is used because it is more accurate than the SYBILT one.

Table 3-14. Convergence study for the single block: summary.

Code	Tracking Module	Solving Method	K-eff	Diff. to MCNP
MCNP	\	Monte-Carlo	$1.25969 \pm 0,00012$	\
DRAGON	SYBILT	Pij	1.26039	+56 pcm
DRAGON	EXCELT	MOC	1.25911	-46 pcm

3.5   Reflector Model: DRAGON

3.5.1   Description of the Reflector Model

The reflector graphite properties for INSTANT have been calculated with a simplified 2-D cylindrical model of the whole core. Fuel homogenized cross sections were first produced in 295 groups using the single-block model, and then a cylindrical core geometry was used whose dimensions were determined such that the total area of the real geometry for each annular region is conserved. Table 3-15 shows the dimensions used. The number densities are the one corresponding to the INSTANT model, which ensures consistency. A void boundary condition is set.

Table 3-15. Determination of the corresponding radii for the reflector regions.

Region	Type of Material	No. Hex.	Cumulated Area (cm^2)	Adjusted Radius (cm)
1	Reflector	1	1128.61	18.954
2	Reflector	7	7900.29	50.147
3	Reflector	19	21443.64	82.618
4	Fuel	169	190735.59	246.400
5	Reflector	217	244909.014	279.208
6	Reflector	271	305854.11	312.020
7	Reflector	331	373570.89	344.835

The INSTANT calculation can be performed in P_3 or in P_1 approximation. In P_1, a sort of diffusion approximation in multigroup can be performed, as explained in Section 2.12.3. However, to do this, the transport cross sections in DRAGON are required, which can only be produced with white boundary conditions. Therefore, an extra ring was added 5.165 cm thick filled with ^{10}B at the periphery with a white boundary condition. This simulates the void condition because the density of ^{10}B is very high, so it can be assumed that the neutrons will be absorbed. This is probably not strictly true for the fast neutrons, and it is a source of error. The other solution is to work with a regular P_3 method in INSTANT. For this method, one may use a void condition at the periphery, and therefore the extra ring of B^{10} would not be needed. Both have been done, which allowed to perform both types of calculations in INSTANT.

The reflector properties will be kept the same for the whole study, so as not to introduce a bias while recalculating them with different methods or different cross sections for the fuel rings. This study focuses only on the preparation of the fuel homogenized cross sections for INSTANT. Later, it would be naturally possible to evaluate other methods of modeling the reflector.

For the same reason, one set of homogenized fuel cross sections must be chosen to perform all reflector calculations. It was decided to use the EXCELT tracking to produce the fuel cross sections for the reflector calculation because this module is considered as more accurate.

The reflector calculation was not validated using MCNP. Nevertheless, the SNT solver has been tested in other projects at Montréal and has proven to be reliable. The K_{eff} of this model should also be relatively close to the K_{eff} of the real one-twelfth core.

3.5.2   Convergence Study of the Reflector Model

A convergence study was necessary to choose the parameters in DRAGON for the reflector model, especially the spatial discretization. Two solvers are available: the SNT solver and SYBILT. The criterion to choose between them was convergence of K_{eff}. The study was performed with the extra ring of B^{10} so that a white boundary condition could be used and transport cross sections produced. The study was not duplicated for both the real and simulated void conditions.

Convergence Study for the SNT Tracking:

Table 3-16(a) gives the results of different discretizations of the annular regions. G stands for the graphite reflector, F for the fuel annular region and ^{10}B for the external layer of boron. In DRAGON, the option SPLITR enables a mesh splitting along the radial direction into zones of equal volumes. Other DRAGON parameters were set to: SN14; LIVO 10 5; MAXI 100; EPSI 1E-05; QUAD10.

Table 3-16. Convergence study for the reflector: geometry discretization and angular discretization.

(a) Spatial discretization.										(b) No. of angles.	
SPLITR: Number of volumes in region:											
1	2	3	4	5	6	7	8				
G	G	G	F	G	G	G	^{10}B	K-eff		SN	K-eff
-15	-15	-25	-50	-25	-15	-15	-2	1.25208		22	1.25206
-10	-10	-20	-50	-20	-10	-10	-5	1.25207		18	1.25206
-20	-7	-8	-30	-8	-9	-10	-10	1.25204		14	1.25206
-10	-10	-20	-20	-20	-10	-10	-5	1.25206		10	1.25206
-5	-5	-10	-20	-10	-5	-5	-5	1.25208		6	1.25205
-20	-7	-8	-15	-8	-9	-10	-10	1.25201		4	1.25205

From this, one sees that the results are rather strongly influenced by the spatial discretization: some discretization seems to be essential, even in the reflector regions. A good compromise between the precision and the calculation time seems to be: SPLITR -10 -10 -20 -20 -20 -10 -10 -5, each number corresponding to the number of divisions in each region. This leads to a K_{eff} of 1.25206, which is close to the K_{eff} obtained with the highest discretization (1.25208). These options were kept for further calculations.

The number of angles taken to do the calculation (SN) was then studied. Results are given in Table 3-16(b). The eigenvalues are particularly stable even with a low number of angles. This comes from the cylindrical symmetry. Fourteen angles were kept for the next calculations, but ten or even six would likely have been sufficient. However, 14 angles enabled the use of previous calculations again for this study. For the final model, 10 angles were used. Calculations with two different angular quadratures types were then performed, whose results are given in Table 3-17.

The two quadratures give almost the same results. This parameter might have more influence in 3-D geometries. The quadrature number 10 was kept: a product of the Gauss-Legendre and the Gauss-Chebyshev quadrature.

Table 3-17. Convergence study for the reflector: angular quadrature type.

Quadrature	K-eff
4: Legendre-Chebyshev quadrature	1.25209
10: Product of Gauss-Legendre and Gauss-Chebyshev quadrature	1.25206

The effect of the Livolant acceleration method (Table 3-18) was also examined.

Table 3-18. Convergence study for the reflector: Livolant acceleration.

| LIVO: Livolant acceleration | | | | |
No. of Free Iterations	No. of Accelerated Iterations	K-eff	No. of Outer Iterations	Calculation Time (s)
30	5	1.25206	79	1434
20	5	1.25206	97	1046
10	5	1.25206	78	1347
5	5	1.25206	44	674
10	25	1.25206	35	727
10	20	1.25211	59	757
10	15	1.25206	68	800
10	10	1.25206	86	1120

Except for the trial with 20 accelerated iterations, all the eigenvalues were the same. Thus, it was concluded that the case with 20 free iterations may have had numerical issues and may not be relevant. Ten accelerated iterations and five free iterations were kept for the next calculations, but it seemed that the best choice for the final model would be to take five free iterations and 25 accelerated iterations.

Finally, the effect of allowing different maximum numbers of thermal iterations in the flux solution (FLU module) was studied. The results are given in Table 3-19. It shows no variation on the K_{eff}, but the number of thermal iterations has an effect on the duration of the calculation. Five thermal iterations will be used to keep calculations as fast as possible.

The following settings were kept for all calculations on this geometry:

- SPLITR -10 -10 -20 -20 -20 -10 -10 -5

- SN 10

- QUAD 10 (quadrature: product of Gauss-Legendre and Gauss-Chebyshev)

- LIVO 5 25

- THER 5 1.10^{-5}.

Table 3-19. Convergence study for the reflector: Livolant acceleration.

THER: Maximum Number of Thermal Iterations	K-eff	No. of Outer Iterations	Durations (s)
5	1.25206	72	828
10	1.25206	78	1347
15	1.25206	92	1719
20	1.25206	88	1538
30	1.25206	75	1036
40	1.25206	78	1520

We obtain:

$K_{eff\ Refl\ SNT}$ = 1.25206.   (3-7)

This is about 194 pcm higher than the K_{eff} obtained in MCNP for the one-twelfth core (see Section 4.1 page 100), which suggests that the reflector model is adequate. The external convergence is reached after 61 iterations and the calculation time is equal to 343 s in the FLU module.

Convergence Study for the SYBILT Tracking:

The same type of study was performed with a SYBILT tracking and the CP method. However, it was necessary to put a smaller concentration of boron at the periphery to simulate the void boundary condition. Otherwise, DRAGON was unable to perform the calculation due to excessively small collision probabilities in the boron region. A higher concentration leads to convergence difficulties. The higher the concentration of boron is, the better the simulation of the void condition may be, so it was decided not to decrease the concentration used with the SNT tracking. This boron concentration is anyway completely arbitrary and virtual, so the model whose K_{eff} convergence is the best was kept. A linear anisotropic model for the interface currents (DP01) was used, because this feature may increase the accuracy of the solution.

The spatial discretization was first studied. Results of several settings are provided in Table 3-20.

Table 3-20. Convergence study for the reflector: spatial discretization.

SPLITR: Number of volumes in region:								
1	2	3	4	5	6	7	8	
G	G	G	F	G	G	G	B^{10}	K-eff
-15	-20	-35	-50	-35	-30	-20	-5	1.24737
-15	-20	-30	-45	-30	-20	-20	-5	1.24718
-10	-10	-25	-40	-25	-10	-10	-5	1.24470
-10	-10	-20	-30	-20	-10	-10	-5	1.24511
-10	-10	-20	-20	-20	-10	-10	-5	1.24533
-20	-7	-8	-15	-8	-9	-10	-10	1.24784
-5	-5	-10	-15	-10	-5	-5	-5	1.23869
-2	-5	-10	-10	-10	-5	-2	-1	1.23237
0	0	0	0	0	0	0	0	1.11115

It was observed that the convergence is quite poor with this solver, even when the geometry is highly discretized. SYBILT may have problems to converge with a large number of regions. The discretization SPLITR -10 -10 -20 -25 -20 -10 -10 -5 was kept for the following calculations, but it is known that it is not well converged.

The number of integration points needed was then examined (Table 3-21[a]).

Table 3-21. Convergence study for the reflector: integration parameters and number of thermal iterations.

(a) Number of integration points.		(b) Number of thermal iterations.			
QUA1: Number of Integration Points	K-eff	THER: Maximum Number of Thermal Iterations	K-eff	No. of Outer Iterations	Duration(s)
20	1.24522	5	1.24522	34	27
15	1.24520	10	1.24523	44	36
10	1.24521	20	1.24522	36	31
8	1.24523	30	1.24521	32	29
5	1.24523	40	1.24523	35	31
3	1.24522	50	1.24523	35	31

Seeing the results, five integration points were kept: QUA1 5. This ensured a good convergence and minimized the calculation time. The effect of different maxima for the thermal iterations was then examined (Table 3-21[b]). As shown, there are not significant differences, because usually the source convergence is reached after a small number of iterations. Therefore, five thermal iterations were kept, which minimizes the calculation duration.

The final parameters chosen for a SYBILT calculation on the reflector model were:

- SPLITR -10 -10 -20 -25 -20 -10 -10 -5

- QUA1 5

- DP01

- THER 5 EPSI 1.10^{-5}.

These gives:

$$K_{eff\,Refl\,SYB} = 1.24522. \tag{3-8}$$

The calculation requires only three outer iterations and a time of 27 s in the flux module. It confirms the speed of this solver. However, the solution is quite unstable regarding the spatial discretization—the eigenvalue was not converged while discretizing the geometry. Therefore, the spatial convergence cannot be really insured. The eigenvalue was also farther from the eigenvalue calculated by MCNP on the one-twelfth core model—a difference of -352 pcm was observed between this and the MCNP model (see Section 4.1).

Decision for the Final Model of Reflector

Considering the results, it was decided to keep the SNT, which tracks all the calculations requiring reflector properties. This solver may provide a better representation of the void boundary condition by increasing the boron concentration by 100 if this feature is needed, and its convergence is much more ensured than with SYBILT. The cylindrical geometry may be particularly suitable for the SN method because the mesh proved to be stable with different spatial discretizations.

In conclusion, when the reflector calculation was performed with SN, an extra ring of boron and a reflective condition at the periphery, it gave:

$$K_{eff\,Refl\,void} = 1.25206. \tag{3-9}$$

The same model without boron and with a void condition at the periphery gives:

$$K_{eff\ Refl\ boron} = 1.24656.$$ (3-10)

Both K_{eff} are quite close to the eigenvalue of the one-twelfth core calculated by MCNP (+194 pcm and -247 pcm, respectively), which suggests that this simplified model may be suitable to provide the reflector cross sections.

3.6   Supercell Model: DRAGON

3.6.1   Description of the Supercell Models

The supercell model attempts to find an intermediate path between the single-block model and a full detailed one-twelfth core model. It consists of adding some external rings to the single-block model so that the central block of interest is influenced by its neighbors. Therefore, the cross sections recovered from this path should be weighted by a somewhat more realistic spectrum, but with a shorter calculation time than in the case of directly modeling a one-twelfth core with all details in DRAGON.

The supercells have 33 rings of cells, which corresponds to one row of neighboring blocks next to the fuel block of interest. In total, the model contains 3367 hexagonal cells. Three different types of supercells were defined to allow for the various numbers of reflector blocks adjacent to the peripheral fuel blocks: either 1, 2, or 3 reflector blocks. Figure 3-2 shows the geometries considered. The block of interest is always the central fuel block, from which the homogenized cross sections are recovered.

| (a) 1 Neighboring Reflector Block. | (b) 2 Neighboring Reflector Blocks. | (c) 3 Neighboring Reflector Blocks. |
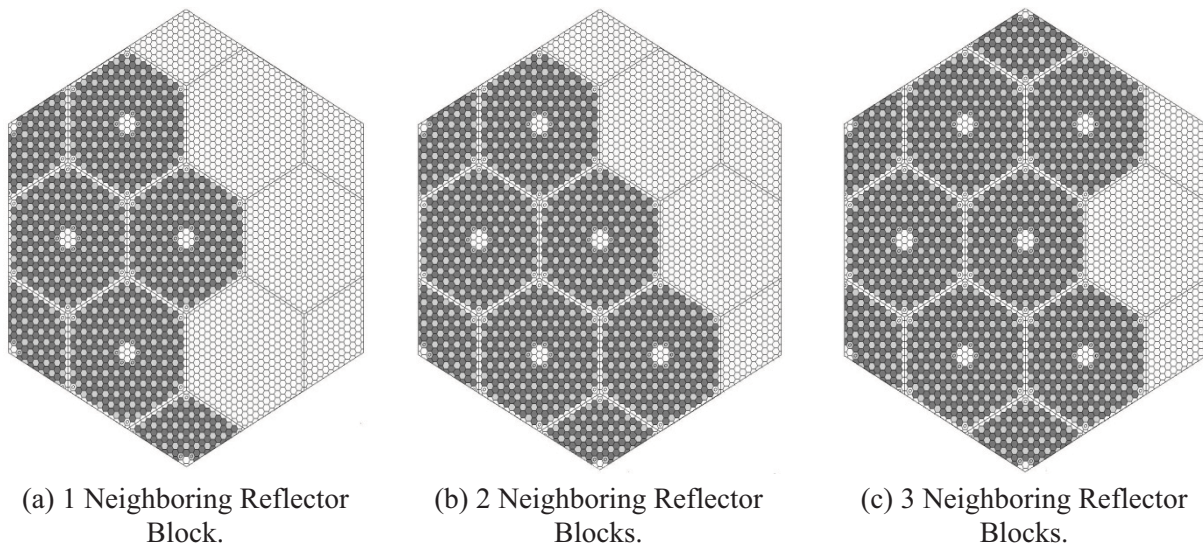
Figure 3-2. Geometrical models for supercells.

A full detailed calculation of this domain with compacts containing double-heterogeneity would require a large amount of memory and time. In addition, the version of DRAGON in use in this work uses an addressing system in 32 bits. This implies that the different objects created to store the data cannot exceed a size of about 2.1 GB. This limitation forces the declaration of homogeneous cells for which homogenized cross sections have to be prepared using the single-block model. It also prevents the use of the method of CP with the EXCELT tracking because the building of the Pij matrices requires more addresses than are available.

In general, the SYBILT tracking is not considered reliable for such a large domain, it was decided to keep the possibility to do this kind of calculation with the EXCELT tracking and the MOC, because the MOC solver is able to use iterative techniques instead of building full matrices.

To reduce the calculation time, a single-block calculation is performed first and the cross sections are homogenized separately, depending on their location. Supercells are then constructed with cross sections for the fuel pellet, the graphite surrounding the fuel pellet, the small coolant cells, the large coolant cells, the plain graphite cells, and the graphite cells of the boundary whose densities are adjusted so that the same number of graphite atoms between the DRAGON models and reality (MCNP) is conserved. Keeping the fuel pellet inside its surrounding graphite instead of homogenizing the whole fuel cell may increase the accuracy, and it also makes the spatial convergence of the source easier.

Ideally, one would not condense the group structure after the single-block calculation, so that the supercell calculation is performed with 295 groups. This is the usual way to do lattice calculations. However, keeping 295 groups for the supercell calculation is very time consuming, especially if discretized geometry is used with the EXCELT tracking module and the MOC. A way to manage this problem consists of condensing the cross sections to 26 groups after the single-block calculation and performing the supercell calculation with only 26 groups. The consequences of this are evaluated in Section 4. However, it may not be adequate if depletion calculations are needed.

After the supercell calculation, the block of interest is homogenized, eventually condensing the group structure. Then the cross sections are recovered to be transferred to INSTANT. The boundary fuel blocks in INSTANT will have cross sections coming from supercell models, whereas the central fuel blocks remain calculated using the single fuel block model of DRAGON. Indeed, building a supercell for them with only fuel neighboring blocks should deliver the same result as the reflective fuel block because all neighbors are copies of the central fuel block of interest.

The tracking module used to do the calculation is critical. The use of SYBILT decreases the calculation time by a factor of 10 compared to EXCELT. However, the SYBILT tracking may not be accurate for such a large domain (personal communication with Alain Hébert). Therefore, it was decided to use EXCELT as well, and a convergence study has been performed with both tracking modules to determine the appropriate parameters. Results will be presented for both modules in Section 4.

EXCELT enables the use of two types of solving methods for the transport equation: the MOC or the CP (PIJ). The limitation in the addressing system prevents the use of the PIJ method, but the MOC solver is working with iterative techniques and is supposed to deliver more accurate results especially when there is anisotropy. Graphite leads to rather isotropic scattering, so that if one is using a 64-bit version of DRAGON, a switch from MOC to PIJ might be interesting to save time.

3.6.2 Convergence Study for a Supercell Model with 2 Steps

It is assumed that the supercells are similar enough so that all that is necessary is a convergence study over just one type of supercell. The supercell where the block of interest is surrounded by three reflector blocks (see Figure 3-2[c]) was selected. A MCNP calculation was performed for this supercell. The reference K_{eff} obtained through MCNP was:

$$K_{eff\ MCNP\ SupCell} = 1.30955 \pm 0.00011. \tag{3-11}$$

The supercell model was built with homogeneous fuel pellets surrounded by graphite in the fuel cells, and homogeneous hexagonal cells for the others (coolant and graphite cells). A white reflective boundary condition was set. All cross sections for the supercell calculations were first generated by DRAGON using the single-block model (MOC with converged parameters) described above in Section 3.4.2.

EXCELT – Method of Characteristics

The flux calculation is first studied with the MOC and EXCELT. This method enables the use of iterative techniques instead of computing the whole collision probabilities matrix. This matrix is usually too large to store in memory (DRAGON encounters are addressing the problems). However, it is likely that a converged mesh for MOC will be suitable for a calculation using collision probabilities. Thus, a mesh that

showed to be discretized enough for one method may also be adequate for the other. Furthermore, it is very likely that the number of groups may not impact the precision of the tracking. Therefore, the convergence study will be done starting with 26 groups to save time. It is assumed that the mesh that will show convergence in 26 groups will also be converged for a calculation with 295 groups.

EXCELT enables the user to define the density of lines over the geometry and the number of angles. It is also possible to discretize and sectorize the cells.

The first set of calculations will study the density of lines. The number of angles will be kept at four angles to minimize the calculation time. Moreover, the anisotropy of this reactor is low, which leads to good results even with a small number of angles. The geometry is not discretized nor sectorized. Results are given in Table 3-22(a). It was concluded that a density of 10 lines per cm is enough to converge on this parameter.

Table 3-22(b) records the results for different numbers of angles but a constant density of lines set to 20 lines.cm^{-1}. The other parameters are the same as above. It was concluded that four angles are enough to describe this geometry. This small number is due to the graphite being rather isotropic because of its high mass compared to the neutrons. On the contrary, if the moderator was water, the hydrogen would have a mass near to the one of the neutrons, which would introduce more anisotropy in the calculation and probably necessitate a finer angular discretization.

Table 3-22. Convergence study for the supercell: tracking parameters.

(a) Density of lines.			(b) Number of angles.		
TISO			TISO		
No. Angles	Density (lines.cm^{-1})	K-eff	No. Angles	Density (lines.cm^{-1})	K-eff
4	500	1.31283	36	20	1.31286
4	400	1.31283	30	20	1.31285
4	300	1.31283	24	20	1.31285
4	200	1.31284	18	20	1.31284
4	100	1.31283	12	20	1.31281
4	70	1.31283	10	20	1.31286
4	40	1.31283	8	20	1.31285
4	30	1.31283	6	20	1.31255
4	20	1.31283	4	20	1.31283
4	10	1.31282	2	20	1.31255

Finally, some calculations were performed to see if further discretization of the geometry would significantly change the K$_{eff}$. For those calculations, eight angles and 70 lines per cm were used. The geometry was sectorized and radially discretized in the fuel and then in the other types of cells. Table 3-23(a) gives the results. From this, one sees that the radial discretization of the fuel pellet is not of interest, but the sectorization introduces about 58 pcm of difference, which is not negligible. Therefore, in the final model, the sectorization of the fuel cells is retained.

Three radii were kept in the fuel pellet and the sectorization was performed into six triangles to study the effect of discretization in the surrounding graphite and coolant cells. Results are given in Table 3-23(b). This shows that there is a very significant effect of the sectorization in the graphite cells. Fortunately, they do not require fine radially discretization, but the sectorization complicates the input decks, and above all it necessitates more memory resources and it slows down the calculation.

Table 3-23. Convergence study for the supercell: fuel and graphite cell discretization and sectorization.

(a) Fuel cells.			(b) Graphite cells.		
Geometry			Geometry		
Number of Radii	Sectorization	K-eff	Number of Radii	Sectorization	K-eff
9	Yes	1.31210	3	Yes	1.30843
6	Yes	1.31211	2	Yes	1.30843
5	Yes	1.31210	1	Yes	1.30850
1	Yes	1.31209	1	No	1.31210
1	No	1.31285			

In conclusion, the following settings were adopted for the supercells using EXCELT and the MOC:

- Tracking parameters: TISO, four angles, 10 lines.cm^{-1}

- Fuel cells: no discretization of the pellet: radii 0.0 0.6225. Sectorization into six triangles.

- Graphite and coolant cells: 1 radius at 0.4 cm and sectorization.

With this configuration and 26 groups, the following is obtained:

$$K_{eff\ SC\ MOC\ 26gr} = 1, 30854 \qquad (3-12)$$

which is at -77 pcm from MCNP. The external convergence is reached after 44 iterations. A complete calculation with the generation of the cross section files takes slightly less than 3 hours on a single processor.

With 295 groups, the following was obtained:

$$K_{eff\ SC\ MOC\ 295gr} = 1.30807 \qquad (3-13)$$

which is at -113 pcm of MCNP, that is to say the same order of magnitude. The calculation times differ greatly: 2 h 39 min with 26 groups compared to 82 h 10 min with 295 groups (e.g., 3, 5 days).

The other two types of supercell were built with the same parameters.

SYBILT – Method of Collision Probabilities

The calculation on supercells were also performed with the SYBILT tracking module. This module uses the CP method and the interface currents method so that the calculation time is greatly decreased. However, the module is usually not trusted on large lattices, and therefore it has to be compared to another module.

A convergence study was performed following the same scheme as for EXCELT, so extensive explanation is not repeated here. The feature DP01 prevented the code from converging, so all cases set a DP00 condition (i.e., an isotropic distribution of the current at the interfaces). Table 3-24 gives the different eigenvalues obtained when varying the integration parameters. QUA2 4 8 (four angles, eight basis points) was kept for the next calculations.

The discretization of the geometry on fuel and graphite cells was then investigated (Table 3-25).

Table 3-25(a) shows that the radial discretization is not important in the fuel cells, but that the sectorization has a strong impact on the K_{eff}. The sectorization with no radial discretization was kept for the fuel cells in the following calculations, which evaluate different configurations for the other types of cells (Table 3-25[b]).

Table 3-24. Convergence study: number of angles and basis points.

QUA2		
No. of Angles	No. of Segments	K-eff
8	10	1.28670
6	10	1.28671
4	10	1.28670
3	10	1.28671
2	10	1.31283
8	10	1.28670
8	8	1.28671
8	6	1.28670
8	4	1.28666
8	3	1.28665

Table 3-25. Supercell: discretization, sectorization of the graphite and coolant cells.

Geometry			Geometry		
Number of Radii	Sectorization	K-eff	Number of Radii	Sectorization	K-eff
6	Yes	1.28672	3	Yes	1.28651
3	Yes	1.28672	1	Yes	1.28671
1	Yes	1.28671	1	No	1.29134
1	No	1.28798	0	No	1.29137

For the graphite or coolant cells, the sectorization proved to be important as well, and so did the radial discretization to a lesser extent. Those features will be kept in the final model. Again, this will significantly slow down the calculations, as was the case with the EXCELT tracking method.

In conclusion, the following model was kept for all supercells which use the SYBILT tracking module:

- Tracking parameters: QUA2 4 6

- Fuel cells: no discretization of the pellet: radii 0.0 0.6225 (cm). Sectorization into six triangles.

- Graphite and coolant cells: three radii at 0.0 0.207 0.415 0.6225 (cm) and sectorization into six triangles.

With this configuration and 26 groups, the following is obtained:

$$K_{eff\ SC\ SYB\ 26gr} = 1.28650 \tag{3-14}$$

which is at -1760 pcm compared to the MCNP result. The external convergence is reached after 26 iterations. A complete calculation with the generation of the cross-sections files takes slightly less than 7 minutes in 26 groups on a single processor.

With 295 groups, it takes about 8 hours and the following was obtained:

$$K_{eff\ SC\ SYB\ 295gr} = 1.28544. \tag{3-15}$$

This is at -1841 pcm of the MCNP eigenvalue. Curiously, the error seems to be higher with more groups, but it remains in the same order of magnitude.

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

March 2011   67

Notice here that the SYBILT solver is much faster than the MOC solver. In 26 groups, performing the calculation with the MOC takes already too much time, not speaking of the week needed to perform it in 295 groups. However, the SYBILT solver seems to be less accurate when comparing its eigenvalue to the MCNP results. It will be evaluated whether this poor lattice solution really affects the cross section generation and to what extent compared to the MOC, which appears to be more reliable.

In the future, the MOC solver may be improved by some parallelization of the code and better acceleration techniques. However, this is outside the scope of this work.

4.   RESULTS AND DISCUSSION OF A CALCULATION SCHEME FOR THE DEEP BURN VHTR

4.1   Reference MCNP Calculation

In MCNP, the K_{eff} for a one-twelfth core calculation with a void boundary condition is 1.24963 ± 0.00010. This will be used to evaluate other solutions coming from INSTANT calculations, along with the average fission rates over each block. These give a measure of the accuracy of the shape of the flux and of the cross sections. This point is of capital importance from an industrial point of view for predicting the power peaks as well as the depletion of the fuel.[3]

All calculations were performed on the high-performance computer Helios of INL using a single node with four Intel® Xeon[b] processors for each calculation. Each processor has a frequency of 2.66 GHz. Once selected, the node is reserved for the calculation, so that each input is running alone.

4.2   First Path: Cross Sections from Single-block Paths

In this first section, results are presented of INSTANT calculations obtained with cross sections generated by the single-block path in DRAGON.

4.2.1   MOC Path: Comparison of the Eigenvalue

Single Block, Double Heterogeneity, 295 Groups, MOC

A first set of core calculations was performed where the fuel cross sections were generated by the single-block model computed from scratch (including compacts containing double heterogeneity, with 295 energy groups, abbreviated by DH295) with the MOC. In most cases, the homogenized cross sections were then condensed to a coarse energy structure. Table 4-1 summarizes the results for every group-structure used in INSTANT whole core calculations.

Table 4-1. Single block, case MOC DH295: eigenvalues.

| | SB MOC DH295 | |
| | P_3 scat 1 | |
No. of Groups in INSTANT	Core K-eff	Deviation from MCNP (pcm)
295	1.25010	38
26	1.25023	48
23	1.24970	6
12	1.24968	4
10	1.24987	19
9	1.24660	-242
6	1.24758	-164
4	1.20891	-3259
2	1.21162	-3042

[b] References herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the U.S. Government, any agency thereof, or any company affiliated with the Idaho National Laboratory.

The deviation is calculated with the following formula:

$$\text{Deviation} = \frac{K_{\text{eff INSTANT}} - K_{\text{eff MCNP}}}{K_{\text{eff MCNP}}} \cdot 10^5 \text{ (pcm)} \tag{4-1}$$

First, it can be seen that when a sufficiently high number of groups is used in the whole core calculation, the agreement between deterministic calculation and MCNP is quite good. This is especially the case between 295 groups and 10 groups.

These results are very encouraging, because they show that keeping a quite large number of groups in the whole core calculation may help to reduce the errors created by a lattice calculation, which does not take into account the environment. The rather inaccurate neutron energy spectrum used to homogenize cross sections over the single block may be partially compensated for by keeping a large number of groups, so that the neutron energies may be redistributed during the core calculation, thus accounting for neighboring reflector blocks, which were not present in the lattice calculation.

It can also be observed that keeping 26 groups at the core level is enough to reproduce the results obtained with 295 groups, with a discrepancy as small as 10 pcm. This shows that there is no need to perform the core calculation with 295 groups if the single-block path is used. It does not provide such an improvement, which would indicate a better modeling of the physics in the core. This statement is valid at this point for the determination of the eigenvalue only. A study of the fission maps is still to complete this statement.

With the nine groups and fewer in INSTANT, the results are becoming worse, or even completely wrong with four or two groups. The definition of the group boundaries may explain that better results are achieved with six groups than with nine groups. Compensation of errors can also give fortuitous results in eigenvalue.

On a more general consideration, the establishment of the boundaries for the groups seems to have a large influence, as the accuracy seems to be improved with 23 groups compared to 26 groups. This may be due to more appropriate boundaries of the 23-group structure, which may avoid cutting some resonances. A deeper analysis should be performed to confirm this statement and an optimized group structure may be adapted to this fuel.

Single Block, Homogeneous Hexagons, 295 Groups, MOC

A second set of core calculations were performed where the fuel cross sections had been generated by the single-block model computed using a previous single-block calculation, which gives homogeneous cross sections for the individual hexagonal cells in 295 groups (abbreviated by HOM295). This was done to provide a fair comparison to the supercell paths. Again, MOC was used for the DRAGON calculations.

Table 4-2 summarizes the results for every group structure. They are very similar to those computed from the original detailed block model. It means that the first single-block calculation was well suited to carry the double-heterogeneity effect through the homogenization. It is encouraging to see this result, because it indicates that the supercell calculations will be started with a good set of cross sections. Indeed, the same initial set of cross sections will be used for this case.

The agreement with MCNP is essentially equivalent to the previous results given that some discrepancy is expected between MCNP and a deterministic code. It shows that both ways are relevant and that it will be worthwhile to analyze the fission maps of the supercell paths.

Table 4-2. Single block, case MOC HOM295: eigenvalues.

No. of Groups in INSTANT	SB MOC HOM295 P_3 scat 1	
	Core K-eff	Deviation from MCNP (pcm)
295	1.24971	6
26	1.24977	11
23	1.24947	-13
12	1.24926	-30
10	1.24947	-13
9	1.24625	-270
6	1.24716	-198
4	1.20861	-3283
2	1.21094	-3096

Single Block, Homogeneous Hexagons, 26 Groups, MOC

Here, to generate homogenized cross sections for INSTANT, a single-block calculation was performed using homogeneous cells, but this time starting with 26 groups. It means that a first detailed single-block calculation is performed in 295 groups to produce homogenized cross sections for each type of elemental hexagonal cells, but those cross sections are condensed to 26 energy groups instead of keeping the initial 295-group structure. Then a single-block calculation with homogeneous elemental hexagonal cells is performed with 26 groups, producing homogenized cross sections for the entire fuel block, which are used in the INSTANT whole core calculation. The cross sections may be condensed to a coarser energy group structures (with less than 26 energy groups) before doing the whole core calculation. The MOC solver was used in DRAGON.

This has been done for two reasons:

- Supercell calculations starting with 26 groups are much easier to complete. A fair comparison was needed for the single-block paths to estimate the added value of carrying a better spectrum at the lattice level.

- It provides an estimation of the influence of the number of groups taken at the lattice level for a standard fuel block.

Table 4-3 summarizes the results. Fewer types of group structures could be used because starting with 26 groups, the possibilities of choosing new boundaries are limited by the boundaries of the 26-group structure. It happens that the 23, 13, and 9-group structures could not be produced because some of their epithermal groups were located inside one large energy group of the 26-group structure. As it is not straightforward to unfold an energy group into smaller intervals, the condensation was not performed for them.

The 10-group structure used in these calculations was created from the 12-group structure to take into account this problem. Other group structures were left alone. Nevertheless, the 26-group calculation is here probably the most interesting of them, because the run time in INSTANT is already very manageable.

Table 4-3. Single block, case MOC HOM26: eigenvalues.

| No. of Groups in INSTANT | SB MOC HOM26 P$_3$ scat 1 | |
	Core K-eff	Deviation from MCNP (pcm)
26	1.25060	78
10	1.25070	86
6	1.24743	-176
4	1.21207	-3006
2	1.21211	-3002

The same errors can be observed appearing when using fewer than 10 groups in INSTANT. The results with 26 or 10 groups are quite comparable. They both lead to a higher eigenvalue than the previous calculations, but it remains acceptable. It is shown here that conducting a lattice calculation with 26 groups instead of 295 may be conceivable if it is necessary due to time constraints. Otherwise, it is of course preferable to conserve 295 groups at the lattice level.

4.2.2   SYBILT Path: Comparison of the Eigenvalue

In this section, all calculations over the single blocks were performed using the SYBILT solver, that is to say a CP method and the interface current coupling method. When a previous calculation was necessary to perform a pre-homogenization, the same set of cross sections for the MOC calculation was used to be consistent.

Single Block, Double Heterogeneity, 295 Groups, SYBILT

Table 4-4 presents the results using directly the double-heterogeneity model.

Table 4-4. Single block, case SYB DH295: eigenvalues.

| No. of Groups in INSTANT | SB SYB DH295 P$_3$ scat 1 | |
	Core K-eff	Deviation from MCNP (pcm)
295	1.25105	114
26	1.25114	121
23	1.25084	97
12	1.25069	85
10	1.25089	101
9	1.24764	-159
6	1.24863	-80
4	1.20993	-3177
2	1.21270	-2955

The SYBILT solver clearly gives worse results than the MOC, but the eigenvalues are reasonably good. The error is about 100 to 140 pcm with MCNP, which is around a factor of two larger than the errors obtained with the MOC.

The SYBILT solver is faster than the MOC. This advantage may be reduced in the near future, as the development team of the École Poly-technique de Montréal is considering working on other acceleration methods for the MOC solver.

Single Block, Homogeneous Hexagons, 295 Groups, SYBILT

As mentioned previously, the cross sections used to perform the homogeneous calculations in DRAGON were recovered from a heterogeneous single-block calculation with MOC so that subsequent DRAGON calculation are started with the same set of cross sections.

Table 4-5 summarizes the results for every group-structure.

Table 4-5. Single block, case SYB HOM295: eigenvalues.

	SB SYB HOM295 P_3 scat 1	
No. of Groups	Core K-eff	Deviation from MCNP (pcm)
295	1.25024	49
26	1.25031	54
23	1.24994	25
12	1.24993	24
10	1.25011	38
9	1.24691	-218
6	1.24776	-150
4	1.20897	-3254
2	1.21184	-3024

Note that the eigenvalues of the better group-structures (from 10 to 295 groups) agree well with MCNP. This may come from the fact that the SYBILT calculation was begun with a set of cross sections that have been generated by the MOC. The double-heterogeneity treatment may be better handled by the MOC solver, and the calculation with homogeneous cells is easier for SYBILT. It would explain the fact that we see a better agreement between the MOC and the SYBILT calculations.

Again, calculations with fewer than 10 groups appear irrelevant.

Single Block, Homogeneous Hexagons, 26 Groups, SYBILT

Finally, Table 4-6 shows the results for the available group-structures starting with 26 groups and homogeneous cells. The same set of cross sections as for the MOC method was used.

The results are quite similar to those obtained with the MOC. The same explanation is proposed—cross sections were prepared by an MOC solution so that the double-heterogeneity treatment is performed with an accurate solver, and then the calculation over homogeneous cells leads to a good agreement with MCNP.

Compared to MCNP, there is an error of about 90 to 110 pcm in the best cases (26 and 10 groups).

Table 4-6. Single block, case SYB HOM26: eigenvalues.

No. of Groups in INSTANT	SB SYB HOM26 P$_3$ scat 1	
	Core K-eff	Deviation from MCNP (pcm)
26	1.25074	89
10	1.25099	109
6	1.24765	-158
4	1.21241	-2978
2	1.21243	-2977

4.2.3   Fission Rate Maps

General Overview of the fission Rate Maps for the First Path

To compare various whole core solvers, power shape is an important metric. However, INSTANT does not contain information on energy per fission. Therefore, in this work, fission rates were used for comparison to MCNP.

It is not possible to plot all fission maps in all cases in this report. It was decided to present the fission maps obtained from the single-block case starting with double heterogeneity and without condensation in energy groups, so that the INSTANT calculation is performed with 295 groups. The P$_3$ calculation was used in INSTANT. Those choices were driven by the consideration that this case should theoretically be the best case of all the calculations using the single-block path.

Figure 4-1 shows a detailed fission rate map of the core. Starting from the center, there are first blocks of reflector containing no fissile elements and therefore no fissions. Then the first inner ring of fuel blocks is encountered where the fission rate reaches a maximum and then decreases. The gradient inside this block is very important, and presents a modeling challenge. Though in an actual design, this power peak would be suppressed with burnable poisons, the thermal flux peak still needs to be accurately resolved in any case.

The fission peak is explained by the high number of thermal neutrons coming from the reflector, which are quickly absorbed inside the first few cells containing fuel. Because the fuel surrounds the inner reflector, the thermalization of neutrons is very efficient because leakage is not likely to occur. Almost all fast neutrons emitted by the core and entering the inner reflector are well thermalized and then reabsorbed in the innermost fuel adjacent to the central reflector.
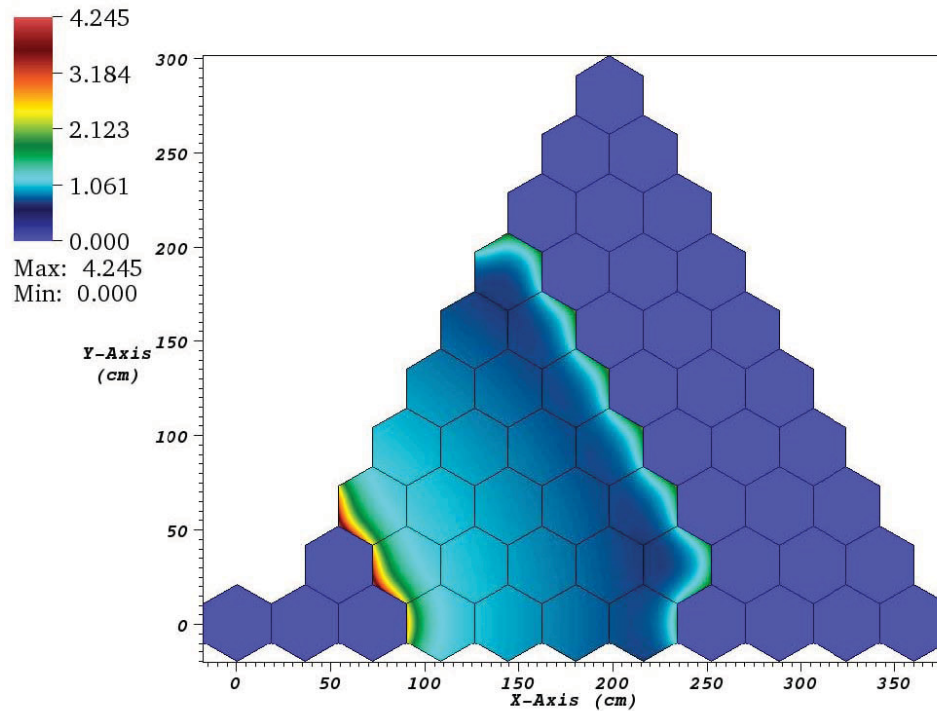
Figure 4-1. Map of the detailed fission rates in the core -SB, 295 groups.

Moving radially outward from the power peak, the fission rates decrease. The reflector is probably no more affecting the flux, and a shape that follows approximately the fundamental mode with leakage may be observed. But upon reaching the outer fuel boundary adjacent to reflector, the fission rate rises again and forms a second peak before vanishing at the outer reflector. This peak comes from the neutron thermalization in the outer reflector; some neutrons coming from the fuel enter the reflector, are thermalized, and re-enter the fuel where they are absorbed quickly in the first fuel cells. This peak is smaller because contrary to the inner part, leakage is more probable. Figure 4-2(a) shows the same data on a 3-D plot, where the fission average is normalized to 1.

It is also possible to plot the fission rate along a line taken across the core, starting inside the reflector, crossing the fuel regions and finishing in the outer reflector. This is shown in Figure 4-2(b).

In this plot, the power peaks at both fuel/reflector interfaces are well appearing, with the highest power reached at the inner interface. The power in the reflector is null, naturally.

There is a factor of 4.2 between the average fission rate (normalized to 1 here) of the fuel and the maximum fission rate at the inner interface. In an actual design, burnable poisons and control rods will flatten the peak. The goal is to achieve the flattest power distribution as possible while the reactor remains critical. This is important regarding thermo-hydraulic concerns, mechanical concerns to minimize the stress in the TRISO's particles, and also regarding the depletion of the fuel, which should be as homogeneous as possible to increase the duration of one campaign and facilitate the refueling operations such as the treatment and recycle process of the spent fuel. However, the peaks may not disappear completely and the spectral effect will still be present.
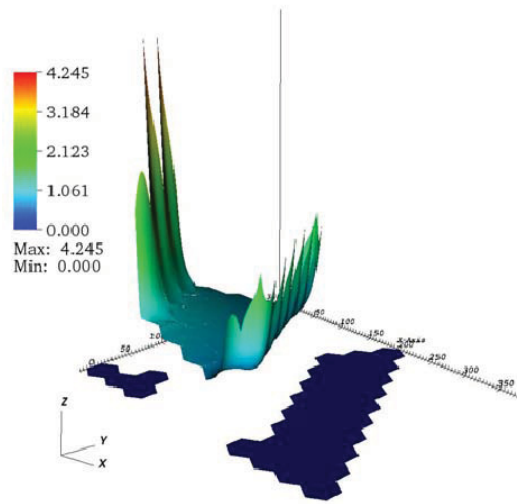
(a) Elevation:the Z-axis represents the fission rate.   (b)Radial traverse across core.
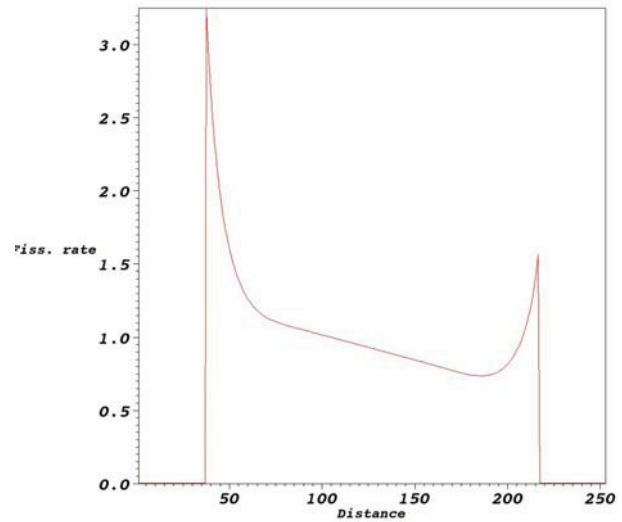
Figure 4-2. Fission rates using SB model and 295 groups in INSTANT.

This awkward shape of the flux is characteristic of an annular core like those designed for VHTR. The annular design was chosen so that it will withstand conduction cool-down.

Figure 4-3(a) shows the flux for neutrons belonging to Group 295, that is to say the neutrons of lowest energy. The scale is arbitrary.
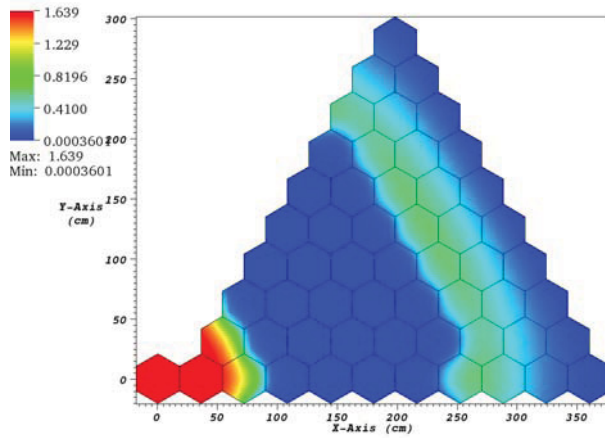
(a) Thermal flux in the core, Group 295.   (b) Fast flux in the core, Group 1.
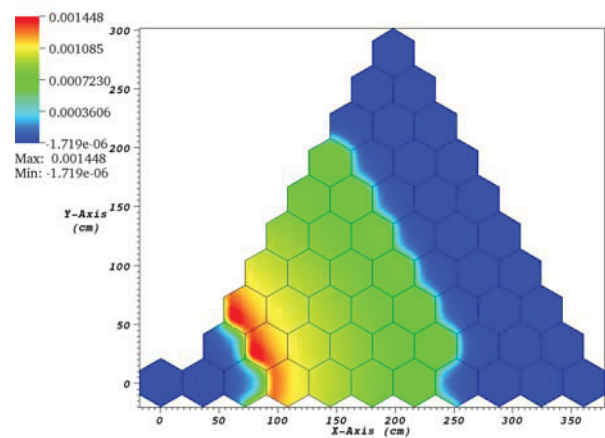
Figure 4-3. Flux analysis: SB MOC DH295, 295 groups.

This shows that the reflector moderates the neutron energy spectrum. As explained above, a larger fission peak is expected in the first row of fuel blocks than in the outer row. Indeed, the flux is lower at the outer interface because of greater leakage.

On the contrary, the neutron flux in Group 1 (Figure 4-3[b], arbitrary scale), where neutrons have the highest energies, is logically high in the fuel regions and nearly null in the reflector. We notice that the peak of the fast flux is located on the inner part of the fuel. This is because of the fission rate is higher in the inner part of the fuel, generating more fast neutrons in this region.

Finally, a fission rate map is shown where the fission rates have been averaged over each hexagonal block (Figure 4-4[a]). This will allow comparison with block-averaged fission rate tallies from MCNP.

In these plots, the fission rates have been normalized such that the average of the fission rates over all fuel assemblies in the core are equal to 1. Similar power peaking observations can be made. As seen in previous plots, we observe the two thermal peaks in the fuel regions adjacent to the reflectors, with a higher peak in the inner part.

(a) Block-averaged Fission Rates.   (b) Percent Error on Block-averaged Fission Rate.
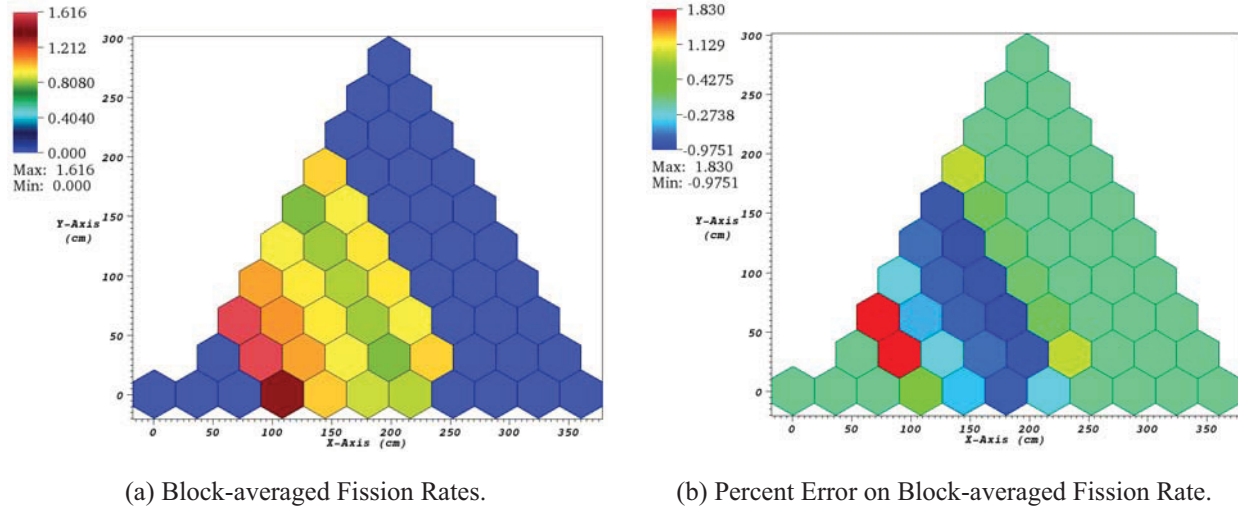
Figure 4-4. SB MOC DH295, 295 groups - block-averaged fission rates and associated error compared to the MCNP reference calculation.

The formula used to generate the errors is in each block is given by:

Fission rate in INSTANT -Fission rate in MCNP

$$\text{Error} = \frac{\text{Fission rate in INSTANT} - \text{Fission rate in MCNP}}{\text{Fission rate in MCNP}} \times 100 (\%) \tag{4-2}$$

Figure 4-4(b) shows the differences observed between the case SB MOC DH295 295 groups with the MCNP reference. Note that the largest statistical uncertainty on the fission rate tallies in MCNP is worth 9.10^{-3}, so about 0.1%.

From this, it can be observed that the inner regions of the fuel is under-predicting fission rates relative to the MCNP results, while it over-predicts the fission rate in the innermost ring of fuel at the inner interface. There is an over-estimation of 1.83% at the inner fuel reflector interface, and an under-estimation of -0.98% in the middle-right of the core. In average, we have -0.14% of errors with MCNP. The standard deviation is equal to 0.81%, which is quite large compared to the average value. This is due to the high magnitude of the errors in the inner peripheral blocks.

Figures 4-5, 4-6, and 4-7 give the same types of maps for different group condensations. With six or more groups, the global aspect of the fission maps remains the same with the presence of two peaks in the blocks surrounded by reflector. With two or four groups, this shape totally disappears.

Figures 4-8, 4-9, and 4-10 show the results of the comparison with MCNP. It appears more clearly that the two and four group structures give a poor shape for the power calculation in the fuel blocks. In the next section, summary information will be presented on the fission maps for each type of calculation in DRAGON, each type of calculation in INSTANT and each condensation (see Section 4.2.3).
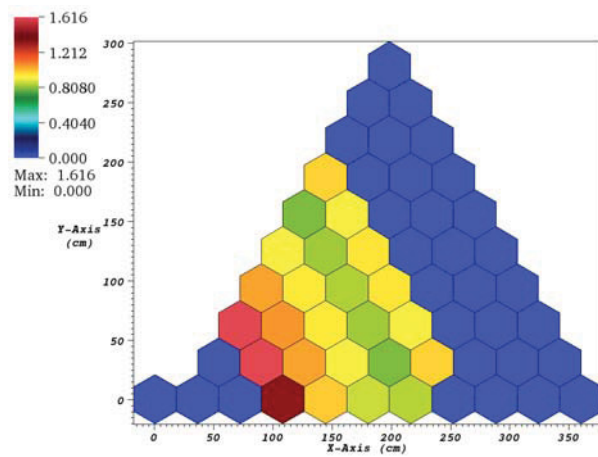
(a) SB, 295 Groups.
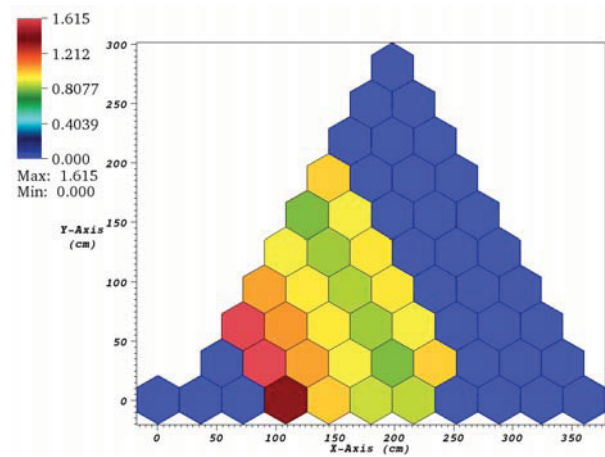
(b) SB, 26 Groups.

Figure 4-5. Fission maps for various group condensations in INSTANT, SB MOC DH295.
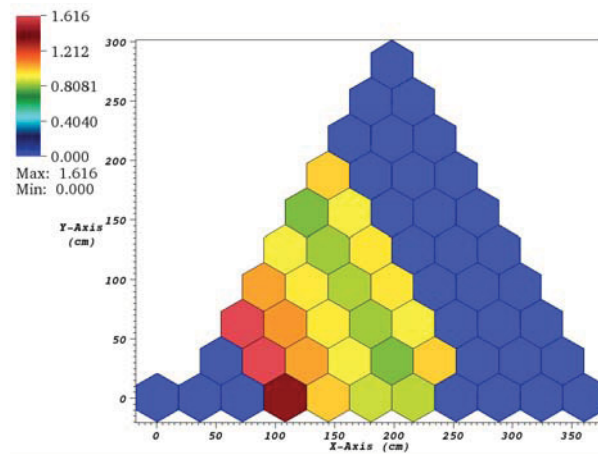
(a) SB, 23 Groups.
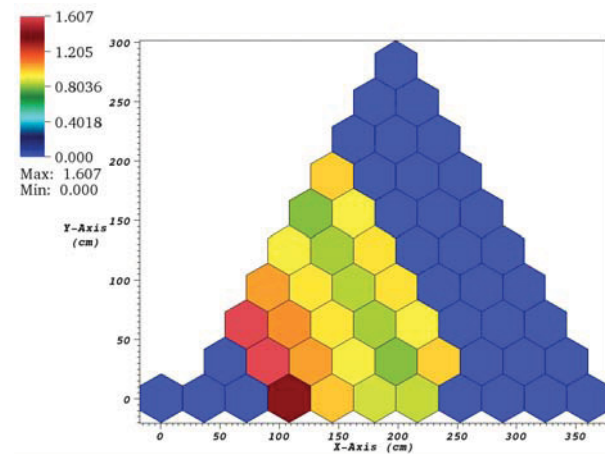
(b) SB, 12 Groups.
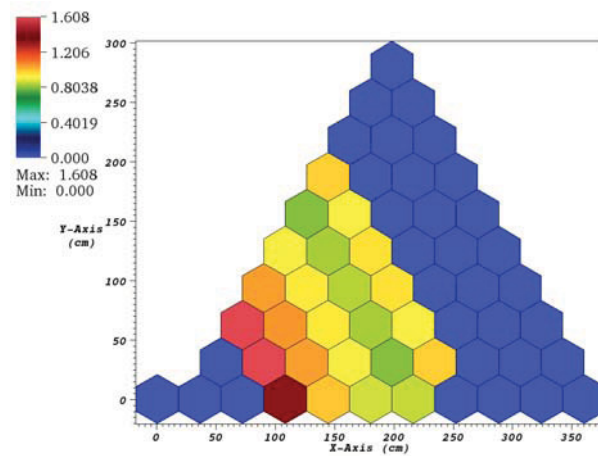
(c) SB, 10 Groups.
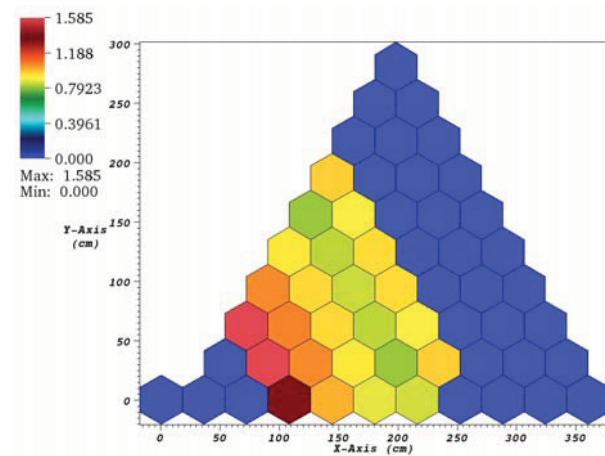
(d) SB, 9 Groups.

Figure 4-6. Fission maps for various group condensations in INSTANT, SB MOC DH295 (continued).

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

78

March 2011
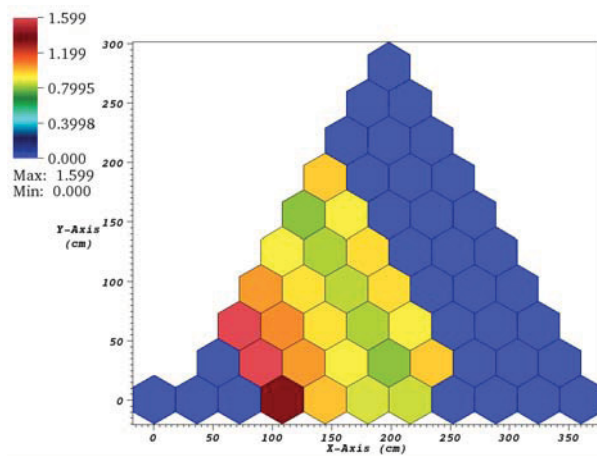
(a) SB, 6 Groups.
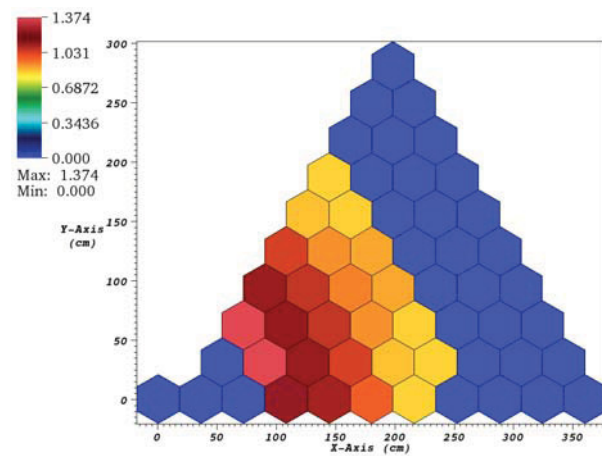
(b) SB, 4 Groups.
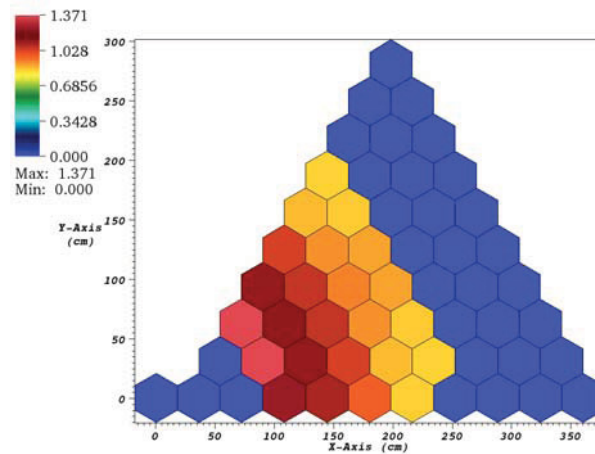
(c) SB, 2 Groups.

Figure 4-7. Fission maps for various group condensations in INSTANT, SB MOC DH295 (end).
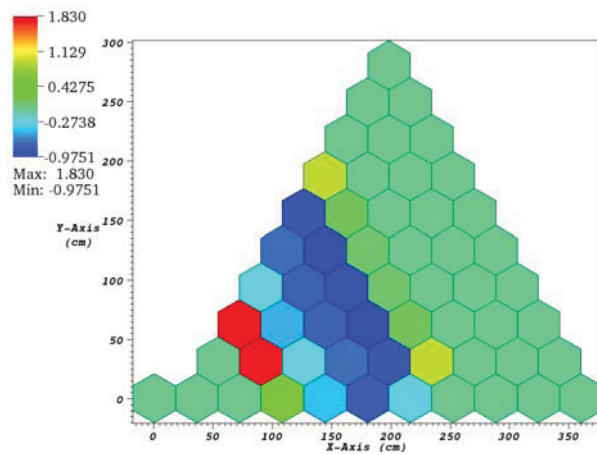
(a) SB, 295 Groups.
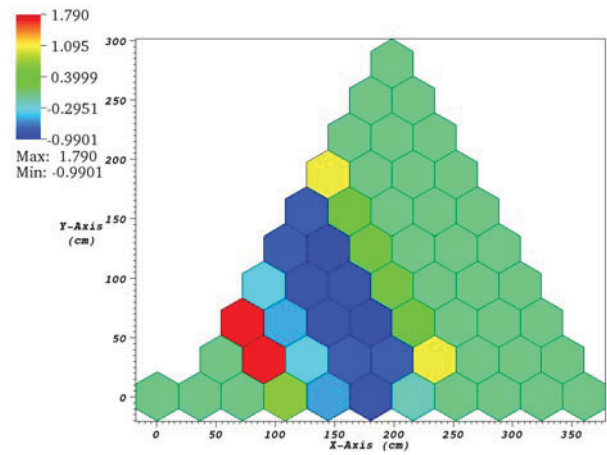
(b) SB, 26 Groups.
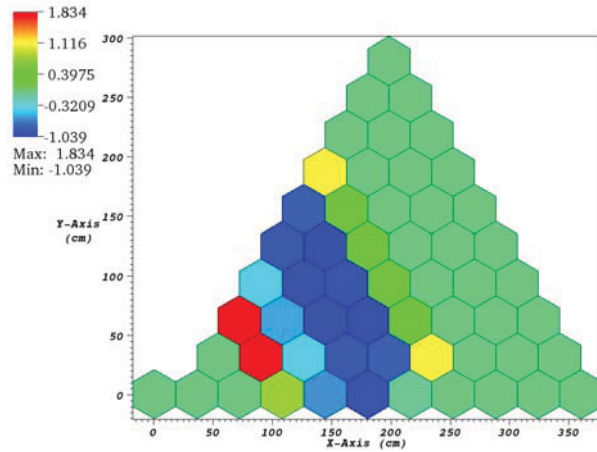
(c) SB, 23 Groups.
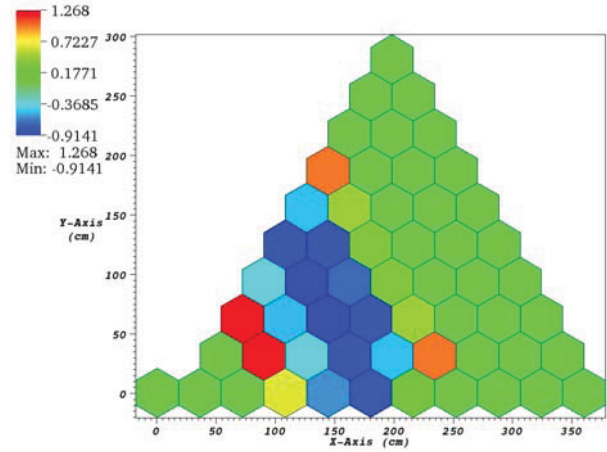
(d) SB, 12 Groups.

Figure 4-8. Block-averaged percent deviation from MCNP fission rate for various group structures in INSTANT, SB MOC DH295.

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

80

March 2011
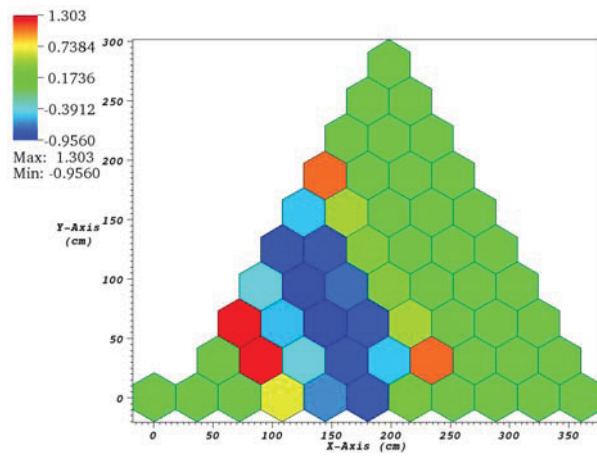
(a) SB, 10 Groups.
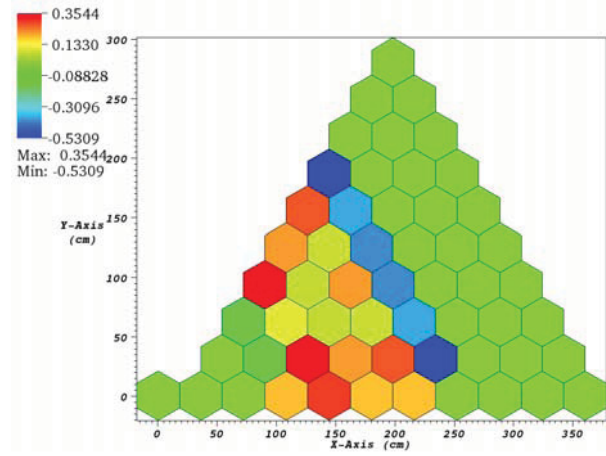
(b) SB, 9 Groups.
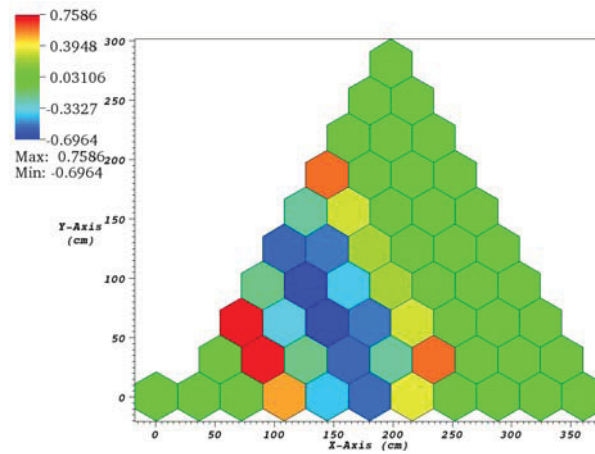
(c) SB, 6 Groups.

Figure 4-9. Block-averaged percent deviation from MCNP fission rate for various group structures in INSTANT, SB MOC DH295 (continued).

(a) SB, 4 Groups.   (b) SB, 2 Groups.
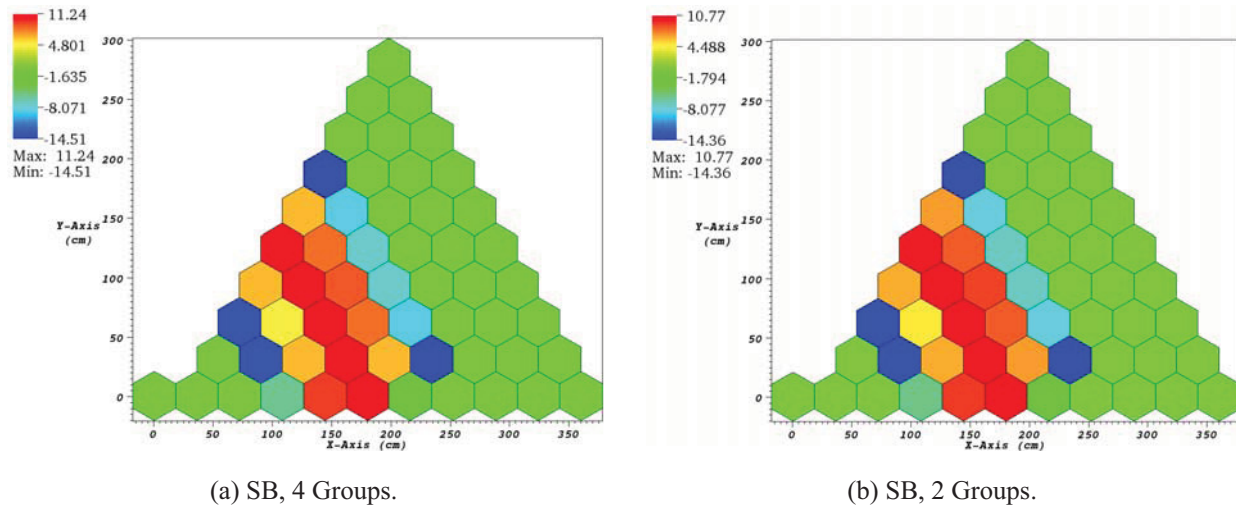
Figure 4-10. Block-averaged percent deviation from MCNP fission rate for various group structures in INSTANT, SB MOC DH295 (end).

Extensive Study of the Fission Maps

The following tables summarize the fission rates obtained through the single-block paths with maximal positive errors (Max), maximal negative errors (Min), average of the errors (Av. μ), standard deviation of errors (Std σ) and the absolute deviation (AbsD), which is the average of the absolute errors.

Single-block, MOC Models. Table 4-7 shows the results using MOC and the detailed double heterogeneity. Curiously, the 9-group structure gives the best results compared to MCNP with errors in the interval [-0.3%; +0.24%], a small average error and a small standard deviation or absolute average deviation. This unexpected result does not appear consistent with the K_{eff} analysis shown previously, where the 9-group structure lead to a K_{eff} much farther from the reference MCNP value.

Table 4-7. Single block, case MOC DH295: fission rate study.

No. of Groups in INSTANT	SB MOC DH295 P$_3$ scat 1				
	Max	Min	Av. μ	Std σ	AbsD
295	1.83	-0.98	-0.14	0.81	0.69
26	1.79	-0.99	-0.13	0.85	0.74
23	1.83	-1.04	-0.14	0.90	0.80
12	1.27	-0.91	-0.10	0.73	0.68
10	1.30	-0.96	-0.10	0.76	0.71
9	0.35	-0.53	-0.009	0.48	0.25
6	0.76	-0.70	-0.06	0.29	0.45
4	11.2	-14.5	0.99	9.36	9.01
2	10.8	-14.4	1.01	9.22	8.86

Using two or four groups is clearly not sufficient to achieve good precision. When the fission rates are plotted, it is clear that the gradients are completely missed, with the maximum reached in the center of the fuel regions instead of the regions adjacent to reflector blocks. The other group-structures better match the

MCNP reference with small errors. But, perhaps unexpected, it does not show that the calculation with 295 groups in the whole core calculation is the closest to MCNP regarding the fission rates.

Table 4-8 gives the results for the path using a pre-calculation and the MOC. It is very similar to the previous ones. Indeed, the DRAGON calculations have been done with the same number of groups, and the cross sections were probably well homogenized by the code before entering the main calculation. Therefore, the resulting cross sections passed to INSTANT are very similar to those coming from a direct single-block model with all details, and it leads to similar whole-core results.

Table 4-8. Single-block, case MOC HOM295: fission rate study.

No. of Groups in INSTANT	SB MOC HOM295 P_3 scat 1				
	Max	Min	Av. μ	Std σ	AbsD
295	1.86	-0.99	-0.14	0.82	0.70
26	1.82	-1.01	-0.14	0.86	0.76
23	1.85	-1.05	-0.14	0.91	0.81
12	1.29	-0.93	-0.10	0.75	0.70
10	1.33	-0.97	-0.10	0.78	0.73
9	0.35	-0.51	-0.007	0.28	0.24
6	0.78	-0.71	-0.06	0.49	0.46
4	11.2	-14.5	0.99	9.46	9.01
2	10.8	-14.4	1.01	9.31	8.86

Again, it is observed that although the K_{eff} matched best between INSTANT and MCNP with larger numbers of groups (23, 26, 295), the fission rate distribution matches best at 9 groups. This is counter-intuitive because more groups should enable one to carry more spectral information and produce more accurate results.

The fission rates remain close to MCNP when the calculations are started with only 26 groups in DRAGON (Table 4-9). It seems that this approximation works well.

Table 4-9. Single-block, case MOC HOM26: fission rate study.

No. of Groups in INSTANT	SB MOC HOM26 P_3 scat 1				
	Max	Min	Av. μ	Std σ	AbsD
26	2.05	-1.03	-0.16	0.87	0.71
10	1.47	-0.91	-0.11	0.71	0.60
6	1.13	-0.69	-0.09	0.55	0.48
4	9.90	-12.7	-0.85	8.27	7.88
2	10.6	-14.2	-0.98	9.19	8.75

Single-block, SYBILT Models. The same study was produced with cross sections generated using SYBILT in DRAGON and the single-block path. Table 4-10 shows results from the path directly using the double-heterogeneity model, whereas Table 4-11 shows results using cross sections computed in 295 groups with a pre-calculation over a block, and Table 4-12 give the results for a calculation performed in 26 groups.

The results are very similar to those obtained from cross sections computed with EXCELT. Curiously, they seem to be even slightly better than the previous results where MOC was used (e.g., in Table 4-10) with 295 groups in INSTANT, the errors are located inside the interval [-0, 94;1, 77] whereas with the MOC path (Table 4-7), the errors are inside the interval [-0, 98;1, 83]. So the width of the errors becomes a bit smaller with the SYBILT path. It is difficult to explain, but the magnitude of the difference is not significant. The conclusion is that regarding the single-block model and the resulting fission rate maps in INSTANT, the MOC and the SYBILT solvers are quite equivalent. On the Eigenvalues, the differences and the advantage of the MOC in DRAGON were more apparent.

Table 4-10. Single-block, case SYB DH295: fission rate study.

No. of Groups in INSTANT	SB SYB DH295 P$_3$ scat 1				
	Max	Min	Av. μ	Std σ	AbsD
295	1.77	-0.94	-0.13	0.78	0.66
26	1.73	-0.96	-0.13	0.82	0.72
23	1.76	-1.00	-0.13	0.86	0.77
12	1.20	-0.87	-0.11	0.70	0.65
10	1.24	-0.91	-0.09	0.73	0.68
9	0.37	-0.57	-0.09	0.31	0.28
6	0.69	-0.66	-0.05	0.44	0.41
4	11.2	-14.5	0.99	9.45	9.00
2	10.8	-14.3	1.01	9.30	8.85

Table 4-11. Single block, case SYB HOM295: fission rate study.

No. of Groups in INSTANT	SB SYB HOM295 P$_3$ scat 1				
	Max	Min	Av. μ	Std σ	AbsD
295	1.83	-0.97	-0.14	0.81	0.68
26	1.79	-0.99	-0.13	0.84	0.74
23	1.83	-1.04	-0.14	0.89	0.80
12	1.26	-0.90	-0.10	0.72	0.67
10	1.30	-0.95	-0.10	0.76	0.71
9	0.36	-0.55	-0.009	0.29	0.26
6	0.75	-0.69	-0.06	0.47	0.44
4	11.2	-14.5	0.99	9.46	9.01
2	10.8	-14.4	1.01	9.32	8.86

Table 4-12. Single-block, case SYB HOM26: fission rate study.

No. of Groups in INSTANT	SB SYB HOM26 P_3 scat 1				
	Max	Min	Av. μ	Std σ	AbsD
26	2.04	-1.02	-0.15	0.87	0.71
10	1.46	-0.90	-0.11	0.70	0.59
6	1.12	-0.68	-0.08	0.54	0.47
4	9.90	-12.7	-0.85	8.27	7.88
2	10.6	-14.3	-0.98	9.19	8.75

4.2.4   Conclusion on the Single-block Calculation Scheme

It has been shown that the single-block path provides fairly good results so long as more than 10 groups are used in the whole-core calculation. The benefit of increasing the number of groups in the core calculation above 12 groups is not obvious from these calculation. In particular, keeping 295 groups is not leading to better results compared to the MCNP reference. The error made on the K_{eff} when using 10 or more groups is typically about 50 pcm. When the cross sections are prepared in DRAGON starting with 26 groups, it increases this error to approximately 100 pcm. Very good agreement was shown between the calculations starting from scratch or starting with homogeneous cells and 295 groups, and this may be explained by the fact that the double-heterogeneity effect is well carried by the first homogenization of the fuel pellet in DRAGON.

The results of whole core calculations show little difference between cross sections generated by the MOC solver or the SYBILT solver. As both methods showed a rather good agreement with MCNP at the lattice level, this point was expected.

4.3   Second Path: Cross Sections from Supercells

The supercell path has been evaluated with the tracking modules SYBILT and EXCELT (MOC). A first calculation was performed on a single block to recover homogeneous properties for each cell, because a direct calculation on a fully-detailed supercell required too much memory. The supercell calculations have been performed starting with 26 and 295 groups in DRAGON.

4.3.1   MOC Path: Comparison of the Eigenvalue

In this section, results are shown for the cross sections generated in DRAGON with the MOC. In INSTANT, the calculations were performed in P3 with linear anisotropic scattering.

Supercells, Homogeneous Hexagonal Cells, 295 Groups, MOC

Table 4-13(a) shows the results for every group-structure when the cross sections were generated in DRAGON using 295 groups. The corresponding results with the single-block path are recalled next to it (Table 4-13[b]).

It can be observed that the agreement is quite good when using 295 groups for either calculation scheme—only -8 pcm and 6 pcm compared to MCNP, which is within the statistical uncertainty of MCNP calculations. When a condensation was performed before the core calculation, the agreement is still within 150 pcm from 26 to 10 groups are used in INSTANT, and then quite poor with 9 groups and fewer.

Table 4-13. Comparison of the eigenvalues between supercells and single-block paths, MOC solver in DRAGON with 295 groups.

(a) Supercells, case MOC HOM295			(b) Single-block, case MOC HOM295		
SC MOC HOM295			SB MOC HOM295		
P_3 scat 1			P_3 scat 1		
No. of Groups in INSTANT	Core K-eff	Deviation with MCNP (pcm)	No. of Groups	Core K-eff	Deviation with MCNP (pcm)
295	1.24953	-8	295	1.24971	6
26	1.24984	17	26	1.24977	11
23	1.24980	14	23	1.24947	-13
12	1.25099	109	12	1.24926	-30
10	1.25126	130	10	1.24947	-13
9	1.25198	188	9	1.24625	-270
6	1.25231	214	6	1.24716	-198
4	1.25939	781	4	1.20861	-3283
2	1.26236	1019	2	1.21094	-3096

Compared to the same type of calculation with the first path (MOC, HOM295), it is observed that the values for 23, 26, and 295 groups are in the same range. No clear advantage can be found for the supercells at this stage regarding the K_{eff}. With 10 and 12 groups, the advantage stands for the single-block path. It seems difficult to explain this observation, but it can most likely be attributed to cancellations of errors.

Supercells, Homogeneous Hexagons, 26 Groups, MOC

Table 4-14(a) gives the results for every group-structure when the supercell calculations were performed in DRAGON starting with 26 groups. As usual, the corresponding results are recalled from the single-block path shown next to it in Table 4-14(b).

Normally, the results are quite good when 10 or 26 groups are kept in INSTANT.

Table 4-14. Comparison of the eigenvalues between supercells and single-block paths, MOC solver in DRAGON with 26 groups.

(a) Supercells, case MOC HOM26			(b) Single-block, case MOC HOM26		
SC MOC HOM26			SB MOC HOM26		
P_3 scat 1			P_3 scat 1		
No. of Groups in INSTANT	Core K-eff	Deviation with MCNP (pcm)	No. of Groups in INSTANT	Core K-eff	Deviation with MCNP (pcm)
26	1.25003	32	26	1.25060	78
10	1.25118	124	10	1.25070	86
6	1.25273	248	6	1.24743	-176
4	1.25832	695	4	1.21207	-3006
2	1.26230	1014	2	1.21211	-3002

With fewer than 10 groups, INSTANT gives poor results.

With 10 or 26 groups, approximately the same Eigenvalues are obtained as when performing this calculation with cross sections generated with 295 groups in DRAGON (see Section 4.3.1). This suggests that the lattice calculation may be done with 26 groups for the supercells instead of carrying 295 groups to perform this calculation. From a computation time point of view, this is very interesting. However, it might be difficult to handle other calculations, such as depletion.

4.3.2   SYBILT Path: Comparison of the Eigenvalue

In this section, results are shown for the cross sections generated in DRAGON with the method of CPs and interface currents. In INSTANT, calculations were performed in P_3 with linear anisotropic scattering in both cases.

Supercells, Homogeneous Hexagons, 295 Groups, SYB

Table 4-15(a) summarizes the results for every group-structure used in INSTANT, when the cross sections were generated in DRAGON with 295 groups. The results from the single-block path are shown next to it (Table 4-15[b]).

The eigenvalues show a good agreement with MCNP until four groups are used in INSTANT. The results are within ±40 pcm with 23 to 295 groups, and with 6, 9, 10, or12 groups they are still at less than 90 pcm from the MCNP reference. This is curious, because 6 or 9 groups seemed to be not enough when the cross sections were computed in DRAGON using the MOC solver. It may be a coincidence resulting from cancellation of errors.

Table 4-15. Comparison of eigenvalues between the supercell and single-block paths, SYBILT solver in DRAGON starting with 295 groups.

(a) Supercells, case SYB HOM295			(b) Single-block, case SYB HOM295		
	SC SYB HOM295			SB SYB HOM295	
	P_3 scat 1			P_3 scat 1	
No. of Groups in INSTANT	Core K-eff	Deviation with MCNP (pcm)	No. of Groups in INSTANT	Core K-eff	Deviation with MCNP (pcm)
295	1.24918	-36	295	1.25024	49
26	1.24932	-25	26	1.25031	54
23	1.24932	-25	23	1.24994	25
12	1.25028	52	12	1.24993	24
10	1.25053	72	10	1.25011	38
9	1.25029	53	9	1.24691	-218
6	1.25069	85	6	1.24776	-150
4	1.24582	-305	4	1.20897	-3254
2	1.24884	-63	2	1.21184	-3024

Those results are interesting because recall that the agreement between MCNP and DRAGON was rather bad at the lattice level. Therefore, it means there is some cancellation of errors leading to such a good agreement. One may also infer that the flux calculated at the lattice level might be quite good in the middle of the domain, but worse at the periphery. It would explain that the cross sections coming from the supercells are more or less accurate enough to give a good eigenvalue in INSTANT with 26 groups. This agreement still has to be also evaluated using the fission rate to make complete statements.

Supercells, Homogeneous Hexagons, 26 Groups, SYB

Table 4-16(a) summarizes the results for every group structure with cross sections generated in DRAGON starting with 26 groups in the supercell path. The corresponding K_{eff} values obtained with the single-block path are recalled next to it in Table 4-16(b).

Again, the agreement is good with 26 groups, and becomes poor with two or four groups. The eigenvalues are comparable both to the corresponding results of the paths using MOC and supercells, but also to the path using supercells, SYBILT, and 295 groups in DRAGON. It would appear that the SYBILT solver may be useful even if the agreement with MCNP was poor at the lattice level.

Table 4-16. Comparison of the eigenvalues between supercells and single-block paths, SYBILT solver in DRAGON with 26 groups.

(a) Supercells, case SYB HOM26			(b) Single block, case SYB HOM26		
	SC SYB HOM26			SB SYB HOM26	
	P_3 scat 1			P_3 scat 1	
No. of Groups in INSTANT	Core K-eff	Deviation with MCNP (pcm)	No. of Groups in INSTANT	Core K-eff	Deviation with MCNP (pcm)
26	1.24951	-10	26	1.25074	89
10	1.25051	70	10	1.25099	109
6	1.25080	94	6	1.24765	-158
4	1.24631	-266	4	1.21241	-2978
2	1.24893	-56	2	1.21243	-2977

To make more complete remarks on the different ways of generating cross sections using supercell, we will now look at the fission maps and compare it to the MCNP reference as we did for the calculations of the first path.

4.3.3   Fission Rate Maps

General Overview of the Fission Maps for the Second Path

Figure 4-11 presents the block averaged fission rates for the case SC MOC HOM295, 295 groups in INSTANT, and the differences observed with the MCNP reference.

The fission rate maps look quite the same as for the first path. The two thermal peaks can be observed in the fuel regions adjacent to the reflector, with a higher peak in the inner part. The highest errors are located in the peripheral fuel blocks with an over-prediction of the fission rates at the inner interface with the reflector and an under-prediction at the outer interface. On the contrary, the middle of the annular fuel region agrees well with the reference.

(a) Block-averaged fission rates.

(b) Percent error on block-averaged fission rate.
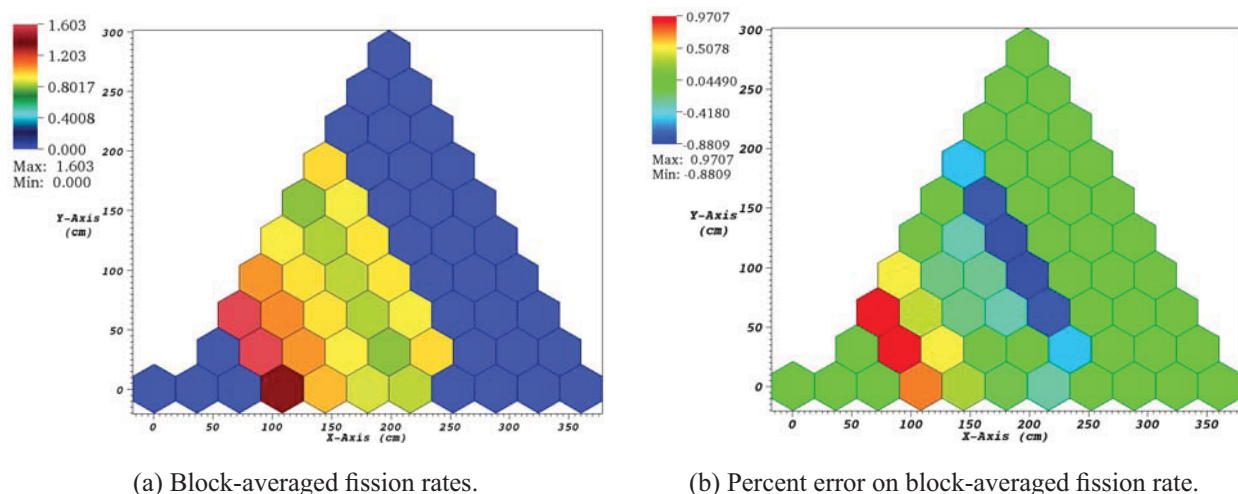
Figure 4-11. Supercell path (SC MOC HOM295, 295 groups in INSATNRT). Block-averaged fission rates and associated error compared to the MCNP reference calculation.

Extensive Study of the Fission Maps

The following tables summarize the results with maximal positive errors (Max), maximal negative errors (Min), average of the errors (Av. μ), its standard deviation of errors (Std σ) and the absolute deviation of errors (AbsD.).

Supercells, MOC, HOM295. Table 4-17(a) presents the results obtained when the supercell calculations were performed in 295 groups in DRAGON and using the MOC. Intuitively, this calculation should provide the best results because the cross sections have to some extent been generated with a spectrum more representative of the real environment in the core. This is, after all, the idea of using supercells. The MOC solver should also be more accurate than SYBILT. It was also demonstrated on the single block that the pre-calculation needed to perform the supercell calculation is accurately carrying the double-heterogeneity effect of the TRISO, so that information is not lost.

Next to Table 4-17(a), Table 4-17(b) recalls the maximum, minimum, and averages obtained with the single-block path. The two other parameters can be found in Table 4-8. To be consistent in the comparison, the path where the same set of initial homogenized cross sections were used in DRAGON was selected, but it does not differ significantly from the results where the DRAGON calculation was directly performed with the TRISO particles.

An improvement is clearly observed with the supercells. For example, in 26 groups, the maximal error has been divided by two using supercells, and the minimal error is reduced by 25%. The averaged error is divided by two and the standard deviation is also reduced by 40%. The supercell path appears to be better for all group structures except the 9-group structure. This may further the argument that the good results obtained with the single-block calculations collapsed to 9-groups was due to fortuitous cancellation of errors.

The supercell path leads to fission rates with errors located in general within $\pm1\%$ compared to MCNP. This is considered a very good agreement. The results are good with 6 to 295 groups in INSTANT. Again, there is curiously very good agreement on the fission rates with the 9-group and 6-group calculations while their Eigenvalues were much farther away from the MCNP reference than the results with finer group structures.

Table 4-17. Fission map study, single block versus supercell paths.

(a) Supercells, case MOC HOM295						(b) Single block, case MOC HOM295			
No. of Groups in INSTANT	SC MOC HOM295 P$_3$ scat 1					No. of Groups in INSTANT	SB MOC HOM295 P$_3$ scat 1		
	Max	Min	Av. μ	Std σ	AbsD.		Max	Min	Av. μ
295	1.03	-0.73	-0.08	0.51	0.44	295	1.86	-0.99	-0.14
26	0.89	-0.75	-0.07	0.49	0.40	26	1.82	-1.01	-0.14
23	1.12	-0.61	-0.09	0.49	0.43	23	1.85	-1.05	-0.14
12	0.64	-0.69	-0.05	0.41	0.34	12	1.29	-0.93	-0.10
10	0.67	-0.64	-0.05	0.39	0.34	10	1.33	-0.97	-0.10
9	0.48	-0.39	-0.01	0.22	0.18	9	0.35	-0.51	-0.007
6	0.51	-0.58	-0.02	0.31	0.26	6	0.78	-0.71	-0.06
4	5.50	-5.89	0.53	3.74	3.35	4	11.2	-14.5	0.99
2	6.86	-5.89	0.54	4.37	3.94	2	10.8	-14.4	1.01

Supercells, MOC, HOM26. Table 4-18(a) presents the results obtained from starting the supercell calculations in DRAGON with 26 groups. The corresponding results obtained by the single-block path are recalled in Table 4-18(b). Both DRAGON calculations were performed with the MOC.

Table 4-18. Supercell fission rate study and comparison with the single-block path, DRAGON calculation done in 26 groups.

(a) Supercells, case MOC HOM26.						(b) Single block, case MOC HOM26.			
No. of Groups in INSTANT	SC MOC HOM26 P$_3$ scat 1					No. of Groups in INSTANT	SB MOC HOM26 P$_3$ scat 1		
	Max	Min	Av. μ	Std σ	AbsD.		Max	Min	Av. μ
295	1.18	-0.90	-0.10	0.62	0.53	295	2.05	-1.03	-0.16
10	0.88	-0.85	-0.07	0.54	0.45	10	1.47	-0.91	-0.11
6	0.62	-0.63	-0.02	0.39	0.35	6	1.13	-0.69	-0.09
4	4.53	-5.08	0.46	3.20	2.90	4	9.90	-12.7	-0.85
2	6.58	-5.70	0.53	4.2	3.81	2	10.6	-14.2	-0.98

Here, an improvement is again observed when supercells are used. When the INSTANT calculation is performed in 26 groups, the maximal error is divided by 1.7 and the minimal error is lowered slightly. The decrease is more impressive in the 10-group calculation, but this might be a coincidence. Also note that compared to the supercell path performed in 295 groups in DRAGON, the loss of accuracy is not very important. For example, the maximal error goes from 0.89 to 1.18 (i.e., 32% of increase). Given the calculation times, it might be interesting to accept this compromise because performing a supercell calculation in DRAGON with 295 groups is time consuming and requires a lot of data exchange.

Supercell, SYB, HOM295. These calculations were repeated using cross sections calculated from SYBILT. Table 4-19(a) describes the results obtained when the supercell calculations were done in 295 groups. The corresponding results from the single-block path (Table 4-11) are partially recalled here in Table 4-19(b).

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

90   March 2011

Table 4-19. Supercells and single-block paths comparison of the fission rate, SYBILT solver in DRAGON.

(a) Supercells, case SYB HOM295.						(b) Single block, case SYB HOM295.			
No. of Groups in INSTANT	SB SYB HOM295 P_3 scat 1					No. of Groups in INSTANT	SB SYB HOM295 P_3 scat 1		
	Max	Min	Av. μ	Std σ	AbsD.		Max	Min	Av. μ
295	1.00	-1.49	-0.046	0.81	0.68	295	1.83	-0.97	-0.14
26	1.01	-1.53	-0.039	0.83	0.69	26	1.79	-0.99	-0.13
23	0.89	-1.29	-0.053	0.72	0.59	23	1.83	-1.04	-0.14
12	0.96	-1.44	-0.01	0.77	0.64	12	1.26	-0.90	-0.10
10	0.92	-1.38	-0.014	0.74	0.61	10	1.30	-0.95	-0.10
9	0.79	-1.20	0.039	0.69	0.63	9	0.36	-0.55	-0.009
6	0.78	-1.11	0.015	0.62	0.53	6	0.75	-0.69	-0.06
4	4.38	-8.29	0.69	3.40	2.62	4	11.2	-14.5	0.99
2	4.75	-8.45	0.70	3.45	2.66	2	10.8	-14.4	1.01

This shows that the group structures from 6 to 295 groups give almost the same results. Here the best group structure regarding the fission map seems to be the 6-group structure. Again, this was not expected because the K_{eff} is not the best one for this group structure.

Here the advantage of the supercells is less evident. For example, with 26 groups in INSTANT, the over-prediction is reduced by 79%, but at the same time the under-prediction is increased to -1.53% compared to -0.99%. The standard and absolute deviations are about the same in both paths. The average of the error is a bit better for the supercell path, which may indicate that the fission rates are better centered on the reference's values, but with the same deviations. Therefore, no clear gain is obtained by the supercells here.

Recalling that the agreement was quite poor with MCNP at the supercell level using SYBILT, perhaps the flux calculation is not accurate enough at this stage to produce cross sections, which would increase the accuracy of the full-core calculation as observed with the MOC path. Compared to this path, it is observed that the loss of accuracy is essentially located on the maximal negative error.

Supercell, SYB, HOM26. Table 4-20(a) shows the results when the DRAGON calculation was performed in 26 groups using SYBILT. Table 4-20(b) recalls some of the corresponding results with the single-block path.

Table 4-20. Supercells and single-block paths comparison on fission rates with SYBILT and DRAGON calculation starting with 26 groups.

(a) Supercells, case SYB HOM26						(b) Single block, case SYB HOM26			
No. of Groups in INSTANT	SB SYB HOM26 P_3 scat 1					No. of Groups in INSTANT	SB SYB HOM295 P_3 scat 1		
	Max	Min	Av. μ	Std σ	AbsD.		Max	Min	Av. μ
26	2.04	-1.02	-0.065	0.88	0.71	26	2.04	-1.02	-0.15
10	1.46	-0.90	-0.036	0.87	0.70	10	1.46	-0.90	-0.11
6	1.12	-0.68	0.011	0.74	0.61	6	1.12	-0.68	-0.08
4	9.90	-12.7	0.59	2.95	2.24	4	9.90	-12.7	-0.85
2	10.6	-14.3	0.67	3.30	2.55	2	10.6	-14.3	-0.98

Again, the results with supercells are good with 6, 10, and 26 groups with a best agreement with 6 groups. In two and four groups, the fission rates are poor. One does not see any difference on the width of the error committed in the two paths compared to MCNP. The average is a bit better for supercells, but the standard deviations are not improved, which indicates that the error is not much improved.

Compared to the equivalent calculations where 295 groups were kept in DRAGON, the loss of accuracy is not totally clear. The maximal error is multiplied by 2 when 26 groups are used in DRAGON instead of 295, but at the same time, the minimal error goes from -1.53% in 295 groups to -1.02%. Note that the average error is lower when DRAGON was using 295 groups.

4.3.4   Conclusion of the Second Path: Supercells

Contrary to the single-block path, the interest of supercells differs depending on the solver, which was employed in DRAGON to produce the homogenized cross sections. The MOC solver gives clearly better results on the shape of the fission rate compared to the single-block model. This is especially true when the supercells were calculated in DRAGON with 295 groups. A reduction of about 62% of the interval width of the errors compared to MCNP is observed in this case. Most of the improvement is seen in the over-predictions of the fission rates in the fuel blocks located next to the inner reflector, which is almost divided by a factor of 2.

When DRAGON is using 26 groups and the MOC, the errors are increased but remain acceptable and better than the corresponding calculations with the single-block model. Thus, a consistent trend of improvement is seen due to the use of supercells. The results of the supercell path with 26 groups in DRAGON are even better than the results of the single-block path with 295 groups in DRAGON.

The prediction of the eigenvalue is also better in the supercell path with MOC when the same number of groups was used in the single-block model in DRAGON. The advantage is here clearer for keeping 295 groups—we obtain a deviation to MCNP equal to only -8 pcm, which is within the statistical uncertainty of the MCNP calculations.

On the contrary, the use of SYBILT to perform the calculations in DRAGON does not lead to a clear improvement of the INSTANT results. The fission rate map is not centered on the same value so that the maximal error is decreased but the minimal error is increased. The K_{eff} values are also farther from the reference.

4.4   Calculation Durations

An INSTANT calculation with the linear anisotropic scattering matrix in 295 groups takes 123 seconds in P_1 and 635 seconds in P_3. With only 26 groups, it takes 5 seconds in P_1 and 37 seconds in P_3. Those values are rather independent of the way the cross sections are prepared because the number of iterations is generally the same whatever the path chosen for cross section generation.

In DRAGON, the condensation has an influence of about 1 or 2 minutes, depending on the amount of data the code has to write in ASCII files. It is clearly not significant compared to the rest of the calculation. Therefore, only the calculation durations are given for cases with a condensation to 26 groups at the end. Table 4-21 concerns DRAGON calculations made on a single block, whereas Table 4-22 concerns supercell calculations in DRAGON.

As can be seen from those tables, the calculations with the MOC are very time consuming, especially for the supercells when using 295 groups. Not surprising is the fact that this method is also the most accurate. The SYBILT solver is fast, but it does not give the same accuracy and cannot serve as a reference calculation.

Table 4-21. Single block: Summary of calculation durations in DRAGON.

	Calculation Durations in DRAGON, Single Block		
	MOC DH 295	MOC HOM295	MOC HOM26
Duration	46 min	8 min	31 s
	SYB DH295	SYB HOM295	SYB HOM26
Duration	6 min	19 s	2 s

Table 4-22. Supercells: Summary of calculation durations in DRAGON.

	Calculation Durations in DRAGON, Supercells		
	MOC DH 295	MOC HOM295	MOC HOM26
Duration	—	82 h 10 min	2 h 39 min
	SYB DH295	SYB HOM295	SYB HOM26
Duration	—	7 h 58 min	7 min

The use of supercells is difficult due to the time constraint. The parallelization of the MOC solver may be of great value if the supercells are determined to be required for calculations in this core.

4.5   Conclusion on the Cross Sections Generation

Considering only a steady-state calculation with fresh fuel, there is a clear advantage to generating microscopic homogenized cross sections from supercell models in DRAGON if the MOC solver is used. However, regarding the calculation durations and the complexity of the use of supercells, it is not clear that the gain of accuracy is significant enough to justify their use. The single-block model already provides a good solution on the condition that enough groups are used in INSTANT, typically 12 or more. The reader should remember here that the interest of supercells may be higher when burnable poisons and control rods are added in the core models.

A partial explanation for the fact that supercells do not make a significant difference compared to the single-block path can be found by the study of the neutron energy spectra in both cases. The word spectrum describes the flux as a function of the energy, normalized such as the integral of this curve is equal to one. Because a multi-group approximation was used, the flux in 295 groups is first divided by a lethargy interval:

$$\frac{\phi(E_i)}{log_{10}(E_{i+1}) - log_{10}(E_i)}, i \in [\![1,295]\!] \tag{4-3}$$

This term is plotted as a function of the base 10 logarithm of the energy (in eV): $log_{10}(E_{central,i})$. $E_{central,i}$ is calculated using the upper and lower boundaries of the group:

$$E_{central,i} = 10^{\left(\frac{log_{10}(E_{i+1}) + log_{10}(E_i)}{2}\right)} \tag{4-4}$$

Notice that what is called "lethargy" in the figures is not equal to the usual definition of the lethargy. It has been used because the fluxes are plotted on a logarithmic scale. It is known that the wider is a group, the more important will be the flux, so that to be able to compare the level of the flux in each group, it is necessary to divide the fluxes by the logarithmic widths of their groups.

Before plotting it, a normalization is performed so that the integral of the function on a logarithmic scale is equal to one. Therefore the fluxes are normalized by:

$$\tilde{\phi}(E_i) = \frac{1}{\sum_{i=1}^{295} \frac{\phi(E_i)}{log_{10}(E_{i+1}) - log_{10}(E_i)} \cdot (log_{10}(E_{i+1}) - log_{10}(E_i))} \times \frac{(E_i)}{log_{10}(E_{i+1}) - log_{10}(E_i)}$$

$$= \frac{1}{\sum_{i=1}^{295} \frac{1}{\phi(E_i)}} \times \frac{\phi(E_i)}{log_{10}(E_{i+1}) - log_{10}(E_i)} \tag{4-5}$$

This normalization enables comparison of several spectra together so that neutron thermalization can be observed in each model. All the spectra have an integral equal to one on a logarithmic scale.

Figure 4-12 contains the spectra for the single block (in blue) and the supercell path (in red).
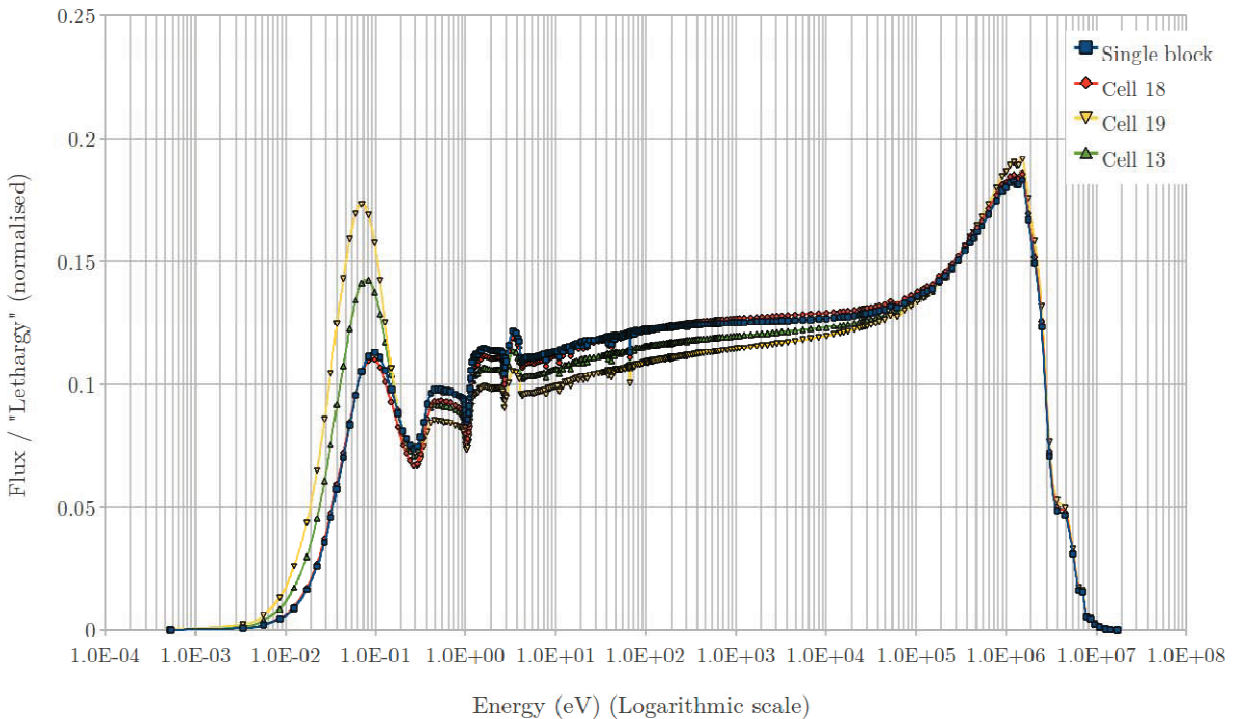
Figure 4-12. Comparison of the spectra averaged over the block of interest between the single-block and the three types of supercells (Cells 13, 18, 19). 295 groups, homogeneous cells, SYBILT.

This shows that the spectra of the single block and of the Supercell 18 look almost the same. The Supercell 18 is made of a block surrounded by three reflector blocks and three fuel blocks. This result was unexpected. It shows that on average, the flux looks the same over the block. The thermal peaks that appear on the right of the supercell next to the reflector are compensated for by the higher fast neutron flux coming from the left side, so that on average the spectrum of the single block is approximately the same. Therefore, the homogenization of the cross sections will not differ much between the two models.

Notice that this is not the case for the two other types of supercells that are surrounded by two blocks of reflector (Cell 19) or one block of reflector (Cell 13). For them, there is a clear thermal peak that is higher when there are two blocks of reflector compared to one block. The thermal peaks are also higher than the one observed for the single block or the supercell surrounded by three blocks of reflector.

As seen in the detailed fission rate map, the blocks next to the reflectors show a high variation of their fission rates in their halves in contact with the reflector blocks. This indicates that in these regions, the spectrum may change a great deal. This is also confirmed by the spectral plots that will be shown for the depletion study (see Section 4.6 and Figure 4-20 later in the text). Considering that the two parts seems to behave very differently, using an averaged of the cross sections over the entire block of interest may not be acceptable because this information on the spectral variation is lost.

Therefore, one recommendation would be to homogenize the two parts separately using, for example, six triangular sectors per block and annular regions in the block. It would then be possible to employ different sets of cross sections for a block located near the reflector in INSTANT. This way of generating cross sections for the core calculation may lead to more accurate results than those presented here because it would then fully utilize the information provided by a supercell model. This may be particularly true when burnable poisons or control rods are added. This additional level of detail was beyond the scope of this work.

4.6   Depletion Study

As all of the machinery to deal with supercell creation had been automated, it was possible to rather quickly perform an initial study of the depletion methods, which may be used in the future for this reactor. A more complete study should be performed to improve understanding of this since the gradients in the inner and outer ring of fuel introduce a very complicated flux shape and perhaps some difficult challenges for the depletion procedures.

4.6.1   Overview of the Problems

In LWRs, the depletion is generally performed in the lattice code. It consists of:

- Performing a calculation over an assembly at different burnup steps and boron histories.

- Creating a reactor database where the homogenized cross sections tabulated as a function of the burnup are stored.

- Passing these to the full-core calculation. The first full-core calculation creates a power map. A time interval is declared in the full-core solver to allow calculation of the burnup of each assembly knowing its power and its composition. Therefor, each assembly potentially has a different burnup.

- The new macroscopic cross sections for each assembly are recovered from the database depending on the particular burnup of each assembly. The full-core calculation can then be performed for burnup Step 1. It provides a new power map for the core, and the process can be started again for next burnup iteration.

It works particularly well for LWRs (at least in a first approximation) because the spectra do not differ between the assemblies. This spectrum is well described by a lattice calculation in the fundamental mode because of the small mean free paths. Notice that discretizing an assembly into four regions of depletion is usually sufficient to ensure a reasonable accuracy because the fluxes remain relatively flat over an assembly. However, this procedure raises two questions when applied to the VHTR.

The first question can be formulated this way: given the fact that in this reactor the flux is peaked in the peripheral assemblies, the concentrations in the blocks may change in a very different manner while burning. Therefore, is it still possible to consider averaged number densities over each block at the core level, or should the method keep track of the detailed variation of the isotopic composition in the blocks to ensure the accuracy of the calculation? According to Henry's 1975 article,[27] the size of the depletion regions is "chosen so that within them, the group parameters at start of life are constant (possibly as the result of some homogenization procedure) and *the flux shapes are reasonably flat.*"[27] Therefore, a finer discretization of the peripheral blocks may be needed for depletion in VHTRs. However, the introduction

of burnable poisons and of burnt assemblies may also sufficiently flatten the peaks to enable the consideration of homogeneous properties over any block in the core. To provide some insights, the influence of the flux peaking on the depletion is studied in Section 4.6 using lattice calculations. A study over a more realistic core should be done to completely answer this question, but it is beyond the scope of this report.

The second question, perhaps more difficult, is to determine whether for a given burnup of a fuel block or of a fuel cell the depletion will be the same whatever the spectrum that has been imposed on the region so that it can reach this burnup. This is probably true for LWRs because the neutron energy spectrum is quite similar for each assembly, whatever the position of the assembly in the core. But is it still the case in this reactor, especially for the peripheral blocks, which are subjected to a strong variation of the flux and to a very different spectrum compared to the middle of the annular core? The following sections present some results which, along with the results already shown, can be used to begin answering this question. More studies on realistic cores should be done as well in further studies to provide a more complete answer.

4.6.2   Possible Depletion Procedures

The interest of doing depletion in the lattice code is that it provides a fine spatial resolution of the flux, and also to perform, for example, a detailed pin power reconstruction once the average power on an assembly has been calculated by the full-core software. However, as was shown in the previous sections (see Figures 4-1 and 4-2[a]), the flux can have a very peaked shape in the peripheral assemblies due to the reflector effect. Considering this, the isotopic concentration will not evolve equally everywhere in the core. Therefore, two solutions may be considered:

- The depletion may be partially performed in the core to calculate the new number densities using the flux coming from the full-core calculation and therefore taking the entire environment into account. The microscopic cross section update would still be performed somehow by the lattice code.

- The depletion is performed in the lattice code, which provides macroscopic cross sections for the full-core calculation. The full-core calculation would only give the burnup of each assembly at each burnup iteration, based on the full-core power map, so that the solver can recover the corresponding cross sections in the reactor database. This solution, adopted for LWRs, could work for this reactor only if a way is found to obtain the correct shape of the flux at the lattice level for each type of block in the core. Otherwise, it is likely that the spectral heterogeneity will not be taken into account, and thus there would be unacceptable discrepancies after a number of burnup steps.

In the first case, the in-core depletion would provide a very accurate isotopic concentration at each burnup step, especially if the discretization in the peripheral fuel blocks is finer than the typical distance of variation of the flux. This is a very interesting technique regarding the current capabilities employed at INL because it only requires the implementation of a depletion module in INSTANT and a way to associate its own set of cross sections for every core region that will also be considered as a homogeneous mixture and depleted. There would still be the issue of generating cross sections because the changes in the concentrations in the core will affect the microscopic cross sections.

A typical scheme may consist of performing a single-block calculation in DRAGON with its own burnup loop. The code would then generate sets of cross sections tabulated as a function of the burnup of the block (for example) and stored in a library that INSTANT can access. Then in INSTANT, each burnup iteration would be followed by a call to the cross section storage so that microscopic cross sections corresponding to the local burnup of each block are provided in INSTANT. An interpolation module is needed to calculate microscopic cross sections for burnup step that may not fall on the burnup points of the library. The macroscopic cross section would then be calculated using the number densities of each region in the core.

This solution may be accurate as long as a sufficient number of groups are used so that it compensates for the fact that:

- The microscopic cross sections may not have been calculated with the exact composition of the blocks at a given burnup step.

- The homogenization is not performed with a representative flux for peripheral blocks, as has been shown in the preceding sections. This difference concerns both flux magnitude and flux spectrum. A trial with supercells may alleviate this concern if the calculation time remains manageable.

Another possible solution is a complete depletion in the lattice code for both densities and microscopic cross sections. To do this and produce meaningful results would mean recreating the spectral environment of every block. One idea to achieve that is to impose the net currents calculated in the full-core calculation at the boundaries in the lattice code. This would then avoid the problem mentioned above; the cross sections would be calculated both with the right shape of the flux and the right composition. It may lead to a gain in accuracy, but the implementation might be difficult and some theoretical work has to be pursued on the issues it presents (e.g., the compatibility with the fundamental mode assumption and the necessity of flux unfolding in angle and energy). Such a task is naturally beyond the scope of this study.

The goal here is to assess the differences between performing depletion calculations on a single block and on a supercell in DRAGON. This will indicate how the reflector affects the number densities in the block of interest, and how the neutron spectrum changes during depletion.

4.6.3   Settings for Depletion

The depletion was performed with following settings for DRAGON: SYBILT calculation in 295 groups in both cases (single-block calculation and supercell calculation) using the same sets of homogenized cross sections as before for the initial calculation, along with the same settings for the spatial discretization of the geometry.

The following time step scheme was adopted for the depletion calculations. Detailed analysis was not performed to evaluate these, as the aim was to evaluate the differences between domain size only:

- 10 day increment from 0 to 50 days

- 50 day increment from 50 to 500 days

- 100 day increment from 500 to 1000 days.

The power was constant and fixed at 600 MW.tonne^{-1} of initial heavy metal. This corresponds to the specification of this reactor. The fuel is meant to pass three times inside the reactor, so that the very high burnup of 600 GW day.tonne^{-1} is attained.

4.6.4   Method of Investigation

To analyze the results of the two methods of depleting the fuel, number densities along a line across the fuel block of interest were examined. This tells whether the shape of the flux has a significant effect on the isotopic variation. Blocks were also homogenized to get the average number densities for each isotope of interest. Finally, the neutron spectra were examined in different locations in a supercell and its change with depletion.

The isotopes selected are ^{238}Pu, ^{239}Pu, ^{240}Pu, ^{241}Pu, ^{242}Pu, ^{242}Am, ^{241}Am, and ^{243}Am. These isotopes are present in the initial loading.

An initial investigation was performed to determine the importance of performing self-shielding calculations at each burnup step in the depletion. Ordinarily, the self-shielding calculations would be

repeated at each burnup step, but this was not possible in the supercell due to the necessity to use pre-homogenized cross sections, as described in the previous sections.

A way to remain consistent between both paths and to circumvent this problem is to suppress the self-shielding in all but the zero-burnup step in both supercells and single-block models. The error may not be so important because luckily, the fuel already contains the most important plutonium isotopes, so that their cross sections will already be self-shielded. The variation of the number densities might not greatly affect this self-shielding. The importance of this approximation has been studied over one single cell.

4.6.5   Study Over One Cell: Importance of Self-shielding During Depletion

To quantify the error committed by not calculating the self-shielding after each burnup iteration, a simple test was performed over one single hexagonal fuel cell. In the trial, the calculation starts with a regular library of 295 groups containing the usual nuclides of the fuel. One cell was burned with a self-shielding calculation after each burnup step, whereas the other was burnt keeping the original self-shielding. The K_{eff} obtained in DRAGON at each time step with and without self-shielding were compared and the final isotopic concentration of several isotopes was evaluated.

Figure 4-13 shows the differences on the K_{eff} as a function of time. The relative difference is calculated by the formula:

$$\text{Difference} = \frac{K_{eff \, without \, self-shielding} - K_{eff \, with \, self-shielding}}{K_{eff \, with \, self-shielding}} \times 10^5 \, (pcm) \tag{4-6}$$
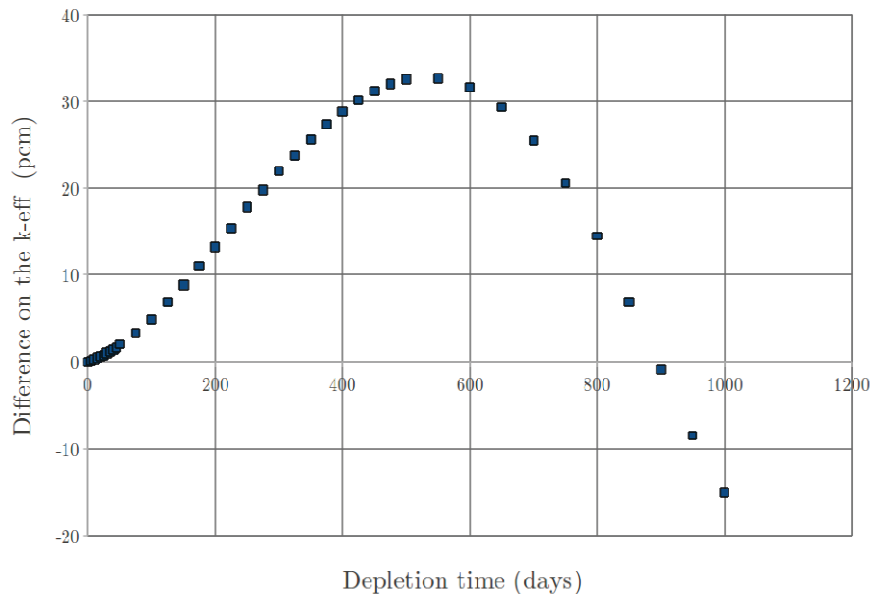
Figure 4-13. Difference of K_{eff} obtained with and without self-shielding over one fuel.

This trial shows that the presence or absence of self-shielding during the burnup iteration is responsible for differences of around ± 30 pcm on the K_{eff} in DRAGON and a maximum of $\pm 0.4\%$ of differences on the isotopic concentration after 1000 days of depletion. Those relatively good results are due to the plutonium and americium isotopes already being present at the beginning and thus underwent a self-shielding before entering the burnup loop. Therefore, the isotopic variation may not greatly affect the self-shielding.

Notice that in the case where some new heavy, transuranic isotopes are created while burning, this assumption may not be valid anymore. The calculation was repeated starting only with ^{235}U and ^{238}U in the fuel instead of the usual Pu-Am fuel, and the results are much less attractive: ± 200 pcm of differences on the K_{eff} (see Figure 4-14) are recorded depending whether or not the self-shielding was repeated at each depletion step. The isotopic concentrations differ by $\pm 20\%$. It clearly shows that the newly-created fissile isotopes need to be self-shielded at each time step. After the new nuclides are built up, the self-shielding update with the burnup may become less important.
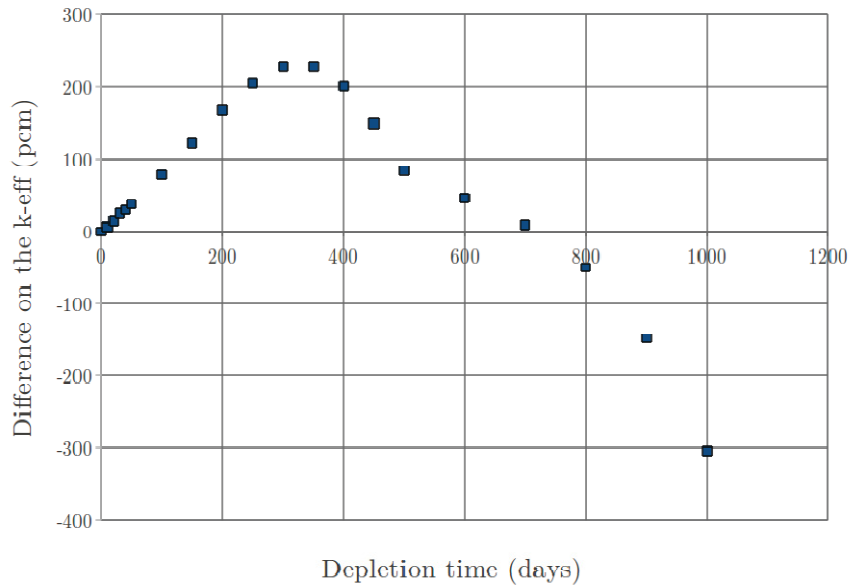
Figure 4-14. Difference of Keff obtained with and without self-shielding over one fuel cell with a conventional fuel (uranium only).

4.6.6   Isotopic Depletion across the Block of Interest

Single-block Path

Figure 4-15 shows the evolution of the K_{eff} during the depletion from t = 0 days to t = 1000 days.

At the beginning, there is the typical steep gradient on the K_{eff}. Then the evolution is smoother.
Table 4-23 gives the correspondence between the time and the burnup for the block. They are related by a linear relation because the specific power is assumed to be constant during all the cycle. As the specific power is the same in the supercells, the correspondence will also be the same.
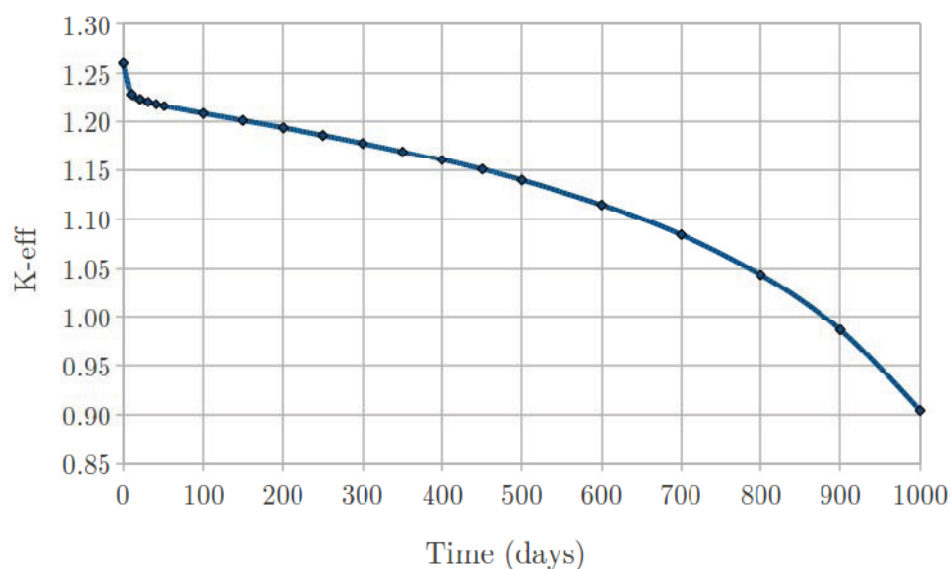
Figure 4-15. Evolution of the Keff of the single block during depletion.

Table 4-23. Single block – time-burnup equivalence.

Time (Days)	0	10	20	30	40	50	100
Burnup (GW day.tonne^{-1})	0	6	12	18	24	30	60
Time (Days)	150	200	250	300	350	400	450
Burnup (GW day.tonne^{-1})	90	120	150	180	210	240	270
Time (Days)	500	600	700	800	900	1000	
Burnup (GW day.tonne^{-1})	300	360	420	480	540	600	

The final burnup (600 GW day.tonne^{-1}) is much higher than what is currently reached in LWRs, but the TRISO particles may allow such a deep burn. It is one of the major interests of this reactor.

The number densities for key heavy isotopes were reported at 0 days, 50 days, 500 days, and 1000 days, calculated in various locations. Table 4-24 describes how the fuel is depleting in a fuel cell near the center of the block, next to the central graphite cells.

Table 4-24. Single block – variation of the concentration at different time-steps near the center.

	Relative Variation Compared to the Initial Value (%)			
Isotopes	$t = 0$ days	$t = 50$ days	$t = 500$ days	$t = 1000$ days
^{237}Np	0.0	-2.9	-28.2	-59.5
^{238}Pu	0.0	4.7	59.6	51.5
^{239}Pu	0.0	-7.9	-65.0	-94.8
^{240}Pu	0.0	-1.3	-25.8	-74.8
^{241}Pu	0.0	10.4	48.7	-37.8
^{242}Pu	0.0	1.9	41.1	116.1
^{241}Am	0.0	-5.8	-45.1	-84.5
^{242m}Am	0.0	70.9	72.0	-66.9
^{243}Am	0.0	4.4	49.3	124.9

It can be observed that all fissile products initially present are disappearing except the ^{242}Pu whose number density is increasing. Of course, minor actinides, like curium, are created as well, some of which may have significant impact on the fuel cycle, but these have not been reported here. The ^{239}Pu is well consumed. Almost 95% of the initial mass has been fissioned or transmuted after 1000 days. The reactor seems to be quite efficient for burning ^{239}Pu, which may be of interest for non-proliferation concerns as well as to reduce the amount of waste.

The question is now to determine how homogeneous is the depletion in the single block. For the work on the single block, a fuel cell near the center of the block and a fuel cell in the middle between the center and the edge were considered. This choice is dictated by the fact that the graphite cells located in the center tend to thermalize the neutrons, so that the flux will be different in a fuel cell next to a graphite cell than in a location where the fuel cell is surrounded by fuel cells and coolant cells. This fact is well shown on Figure 4-17, which represents the flux condensed to one group, plotted across the block from the center to the edge. It will be discussed further in Section 4.6.7.

Table 4-25 shows the differences in the number densities at the same time between the two locations of interest inside the single block. Small differences appear between the two locations, which is explained by differences in the flux due to the central graphite cells. The location near the graphite in the center is taken as a reference from which relative differences are calculated.

Table 4-25. Single block – variation of the concentration at different time-steps in two locations: near the center (reference) and between the center and the edge.

Isotopes	Variation of the Concentration Compared to the Center (%)		
	t = 50 days	t = 500 days	t = 1000 days
^{237}Np	0.05	0.49	1.15
^{238}Pu	-0.06	-0.34	0.54
^{239}Pu	0.32	3.49	4.16
^{240}Pu	0.07	2.18	8.77
^{241}Pu	-0.46	-0.99	2.83
^{242}Pu	-0.13	-1.49	-0.91
^{241}Am	0.24	1.89	3.72
^{242m}Am	-0.93	1.80	3.61
Am^{243}	-0.07	-0.94	-1.68

From this, it can be concluded that even in the single block, the depletion is not totally homogeneous. The maximal difference is of 8.77% on the ^{240}Pu after 1000 days, which is significant. The density of ^{239}Pu is varying by more than 3.5% after 500 days and this density is quite important as this is the major fissile isotope in the core, at least during the first part of the irradiation.

Supercell Path

Here the same study is performed, but considering a supercell model with three reflector blocks and three fuel blocks surrounding the block of interest, described previously (see Figure 3-2[c]). Figure 4-16 shows the evolution of the K$_{eff}$ during the depletion from t = 0 days to t = 1000 days.

The number densities have been recovered at different locations across the block of interest. Values are presented for a fuel cell located at the extreme right, next to the reflector block. The flux spectrum in this region is very different due to the thermalization of the neutrons in the reflector.[28] This is confirmed in Section 4.6.7. Therefore, differences are expected in the depletion. Table 4-26 gives the variation of the number densities in this right cell compared to the initial values at 0 days, 50 days, 500 days, and 1000 days.
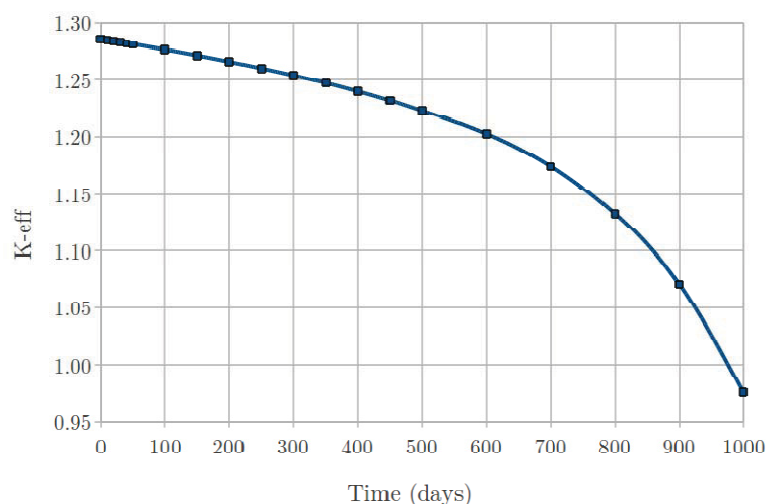
Figure 4-16. Evolution of the K_{eff} of the supercell model during depletion.

Table 4-26. Supercell – isotope densities at various times in a fuel cell located next to the reflector.

Isotopes	Variation of the Concentration Compared to the Center (%)			
	$t = 0$ days	$t = 50$ days	$t = 500$ days	$t = 1000$ days
^{237}Np	0.0	-2.7	-27.2	-58.9
^{238}Pu	0.0	7.4	38.5	-7.3
^{239}Pu	0.0	-11.6	-76.3	-96.5
^{240}Pu	0.0	2.1	-10.9	-60.9
^{241}Pu	0.0	0.1	-11.6	-63.4
^{242}Pu	0.0	6.0	67.9	140.6
^{241}Am	0.0	-6.6	-53.7	-89.6
^{242m}Am	0.0	34.1	-23.5	-84.7
^{243}Am	0.0	2.4	34.1	93.4

Initially, Table 4-26 shows that the evolution is similar to what was observed with the single-block model. For example, the ^{239}Pu is burnt in this cell as well as in the cell studied on the single-block model. Table 4-27 provides a comparison of the depletion at time $t = 1000$ days between the cell taken in the single block and the cell taken in the supercell, so that the comparison is more straightforward.

This shows that some of the isotopes, the ^{238}Pu and to a lesser extent ^{241}Pu, do not behave the same in the two models. In the fuel cell of the single-block model, the ^{238}Pu had been only produced during the cycle, while in the supercell, it is produced during the first 500 days but then consumed, ending the irradiation with less than was initially present.

Again, it is of interest to assess whether the depletion is the same between a fuel cell located next to the reflector on the right, and a fuel cell located next to a fuel block (on the left). Therefore, isotopic number densities were reported in these two locations at time $t = 1000$ days. Table 4-28 gives the relative density variations compared to the initial number densities in the two locations of interest.

Table 4-27. Comparison between the fuel cells taken from the single-block model and the supercell model after 1000 days irradiation.

	Variation of the Concentration Compared to the Center (%)		
	Single Block (% of the Initial Density)	Supercell (% of the Initial Density)	Difference SC-SB (% of the Initial Density)
^{237}Np	-59.5	-58.9	0.6
^{238}Pu	51.5	-7.3	-58.8
^{239}Pu	-94.8	-96.5	-1.7
^{240}Pu	-74.8	-60.9	13.9
^{241}Pu	-37.8	-63.4	-25.5
^{242}Pu	116.1	140.6	24.6
^{241}Am	-84.5	-89.6	-5.1
^{242m}Am	-66.9	-84.7	-17.8
^{243}Am	124.9	93.4	-31.6

Table 4-28. Supercell model – evolution of the number densities in two locations between $t = 0$ and $t = 1000$ days.

Location Near a Fuel Block (Left)		Location Near a Reflector Block (Right)	
Isotope	Density Relative Variation (%)	Isotope	Density Relative Variation (%)
^{237}Np	-57.8	^{237}Np	-58.9
^{238}Pu	42.4	^{238}Pu	-7.3
^{239}Pu	-94.4	^{239}Pu	-96.5
^{240}Pu	-64.0	^{240}Pu	-60.9
^{241}Pu	-46.6	^{241}Pu	-63.4
^{242}Pu	113.8	^{242}Pu	140.6
^{241}Am	-84.6	^{241}Am	-89.6
^{242m}Am	-72.2	^{242m}Am	-84.7
^{243}Am	120.0	^{243}Am	93.4

While the simulations are started with exactly the same concentration in each fuel cell at the beginning, the depletion differs depending on the position inside the block. The effect is amplified compared to the single-block model because the amount of graphite seen by the fuel cell on the right is much higher than what is seen on the left. A cell located near a fuel block will not have the same spectrum as a cell located near the reflector, whose presence is drastically affecting the depletion.

The left part seems to behave more or less like what was observed on the single-block model. It is not completely equivalent; it might be a bit affected by the reflector, but more probably, the different burning state on the right part of the block may lead to different conditions for the left side. An increase or decrease of the fission rates and absorption rates on the right may affect cells on the left by altering the neutron spectrum.

The right side of the block of interest displays very different behavior than the left. For example, ^{238}Pu is produced on the left, increasing the initial concentration by 42.4%; however, it is already consumed on the right, reaching a concentration below the initial one. The number densities of the isotopes ^{242}Pu and

[243]Am are also particularly different between the two sides, which indicate a high dependence of those isotopes on the neutron spectrum in this reactor.

Table 4-29 gives the variation of number densities for every isotope at different dates between the left and the right side of the block, the reference being the left side. The discrepancy is becoming larger during the depletion, especially in the middle of the depletion at t = 500 days. The [239]Pu number density differs by 40% between the two cells, which is rather important and much higher than what was observed when the lattice calculation was performed on a single block.

Table 4-29. Supercell model -variation of the concentration between locations at different time-steps (%).

Isotopes	Concentration's Variation between Left and Right (%)		
	$t = 50$ days	$t = 500$ days	$t = 1000$ days
^{237}Np	-0.4	-3.3	-2.6
^{238}Pu	-0.8	-17.1	-35.9
^{239}Pu	-5.2	-40.3	-38.1
^{240}Pu	3.0	8.8	8.6
^{241}Pu	-7.3	-35.2	-31.4
^{242}Pu	4.3	22.7	12.5
^{241}Am	-2.3	-23.4	-32.8
^{242}Am	-15.5	-53.1	-45.0
^{243}Am	-1.2	-5.8	-12.1

4.6.7   Spectrum and Flux Study

Finally, fluxes were recovered in one group and the spectra of various locations were compared as described in Section 4.5 for the single block and the supercells at t = 0 and t = 1000 days.

Single Block

As would be expected, the flux over the single block is more or less flat, as shown on Figure 4-17. This explains why the depletion is quite homogeneous inside the block.

Figure 4-18 shows the spectral changes between t = 0 days and t = 1000 days. Here the spectrum comes from the flux in 295 groups homogenized over the block. There is no significant change in spectrum across the block, explaining why the densities do not differ much between two fuel regions inside the block. While depleting, the spectrum is becoming more thermal, as the thermal cross sections are reduced and neutrons spend more time thermal before being absorbed.
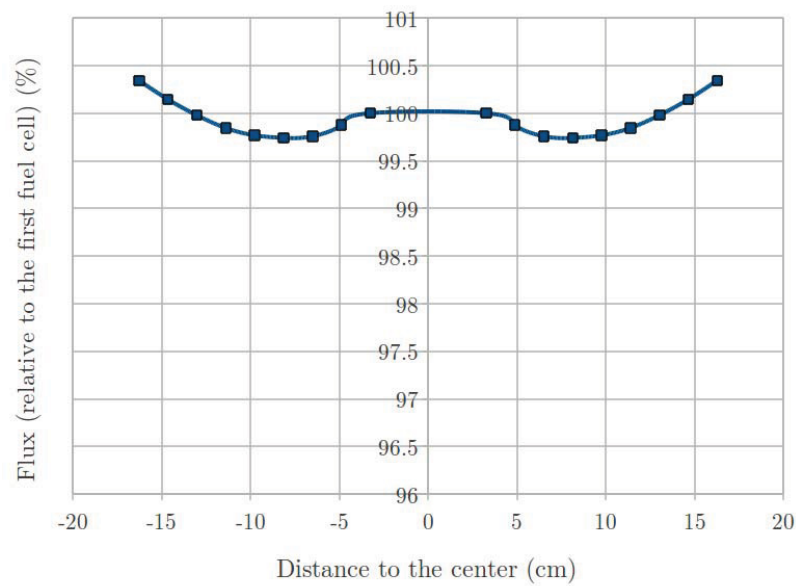
Figure 4-17. Single block: relative flux in one group across the single block, x = 0 being the center.
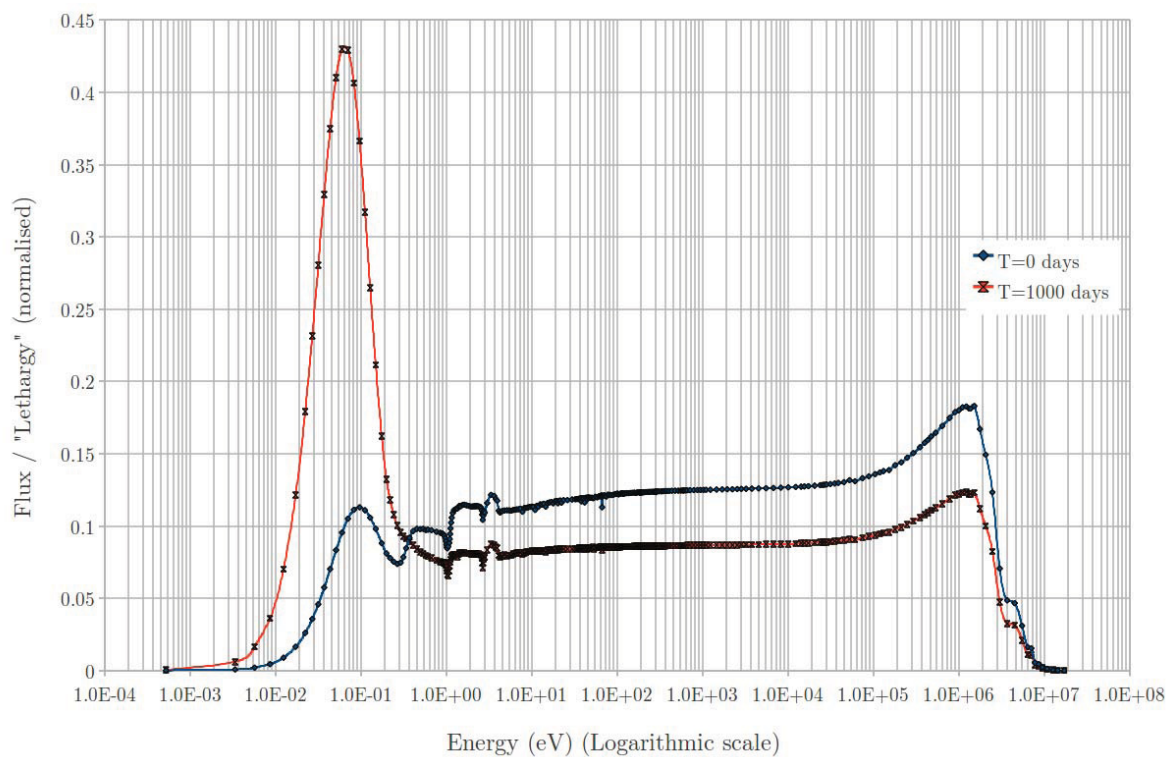
Figure 4-18. Single block: spectral evolution between t = 0 days and t = 1000 days.

Supercell

For this study, the same supercell with three blocks of reflector is considered.

The plot of the flux in one group across the block of interest reveals that it cannot be considered as flat (see Figure 4-19). With more than one group, more variations may be observed.
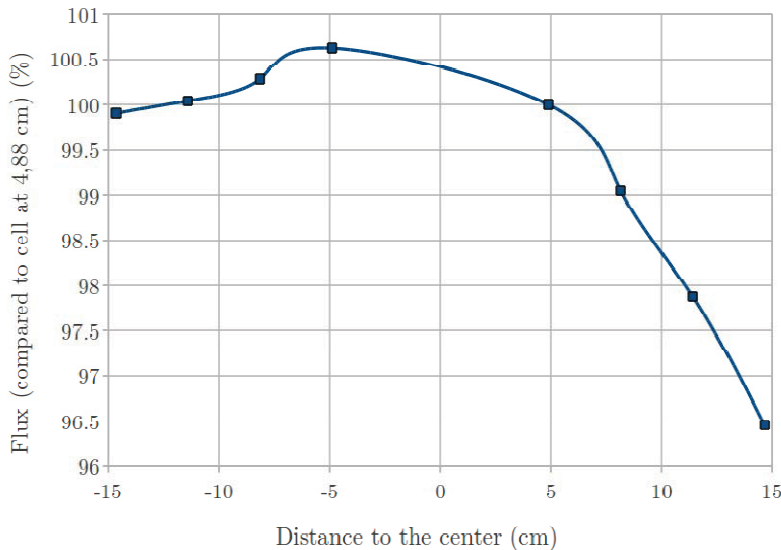
Figure 4-19. Supercell: flux in one group across the block, $x = 0$ being the center.

The magnitude is not the only parameter changing, the spectrum also exhibits strong variations. Figure 4-20 shows the spectra obtained in different fuel cells on a line crossing the center of the block from the left side (next to a fuel block) to the right side (next to a reflector block). The cells were numbered F_1 to F_8, F_1 being the closest cell to the fuel block on the left and F_8 being the closest cell to the reflector block on the right.

The spectra in cells F_1 to F_4, that is to say on the left part of the block of interest, look quite similar. Therefore, they were removed from the plot to clarify the chart. Entering the right part of the block, closer to the reflector, it is observed that the thermal peak's magnitude is increasing. The highest magnitude is reached in fuel cell F_8 next to the reflector. On the contrary, the epithermal flux is higher in the region next to the fuel block because the neutrons coming from the neighboring fuel blocks are less thermalized. The fast flux is higher in cells located next to the reflector because the thermal neutrons coming back from the reflector are well thermalized and directly absorbed in the first fuel regions, leading to fissions and thus to the release of fast neutrons. A good part of those neutrons may enter the reflector, become scattered, and come back with thermal energies to the fuel block, so that the thermal flux is increased.

This spectral variation will strongly affect the depletion because the thermal neutrons coming from the reflector will have a higher probability of being captured. For example, this explains why the ^{239}Pu is more burned near the reflector; its absorption cross section is higher at low energies, and this region will receive more neutrons in this range of energies, so that the absorption reaction rate will be higher.

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

106   March 2011
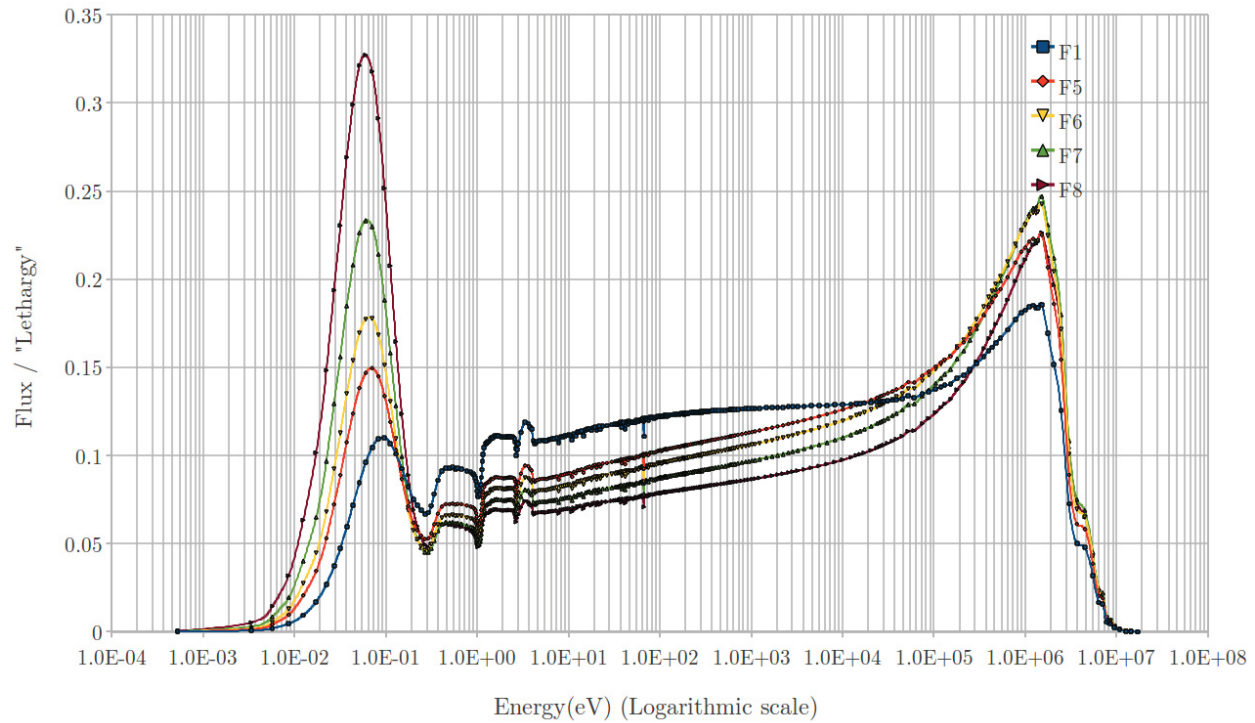
Figure 4-20. Supercell: neutron energy spectra in different regions of the block of interest, t = 0 days.

At the end of the depletion, the same pattern is observed (see Figure 4-21). The right half of the block exhibits a higher thermal peak. Globally, this peak is higher at the end of the depletion than at the beginning, which is explained by the fact that there are less fissile isotopes, and neutrons are thermal longer before being absorbed. However, note that the difference is less pronounced: at t = 0 days, the thermal peak is more than three times higher in F_8 compared to F_1; at t = 1000 days, it is just 1.4 times higher.
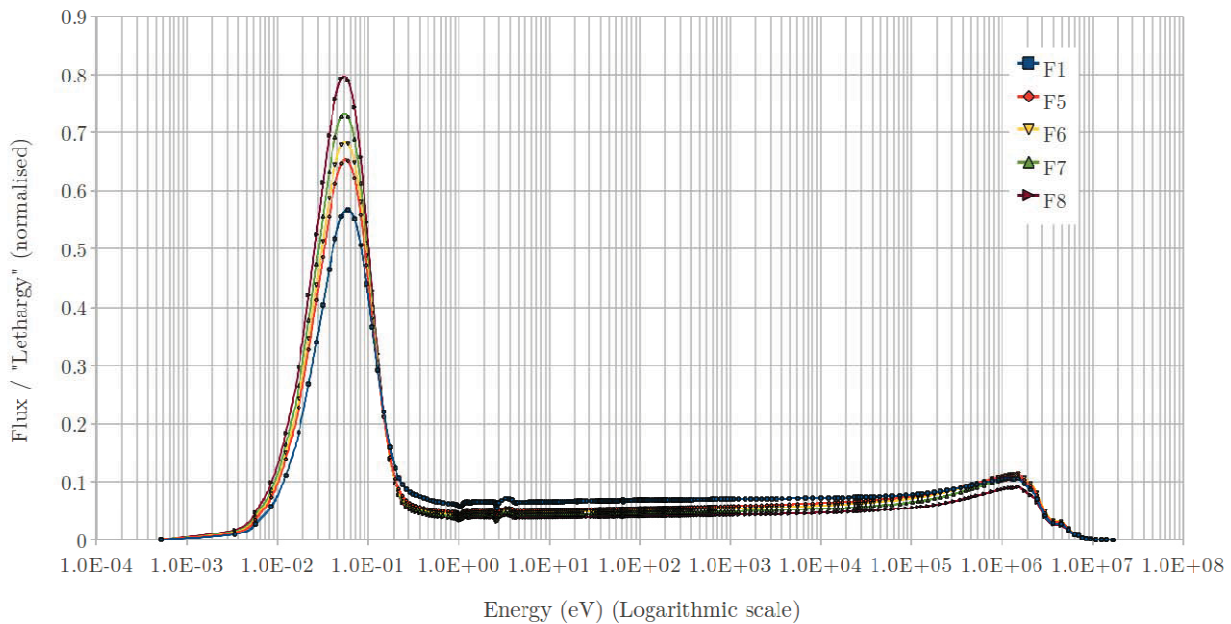
Figure 4-21. Supercell: neutron energy spectra in different regions of the block of interest, t = 1000 days.

4.6.8   Conclusion of the Depletion Study

Several conclusions can be drawn from this study. First, a block surrounded by reflector and fuel blocks exhibits a strong variation in its neutron energy spectrum with a thermal peak, which is 1.5 to more than three times higher on the reflector side than near the neighboring fuel block. As a consequence, the fuel will not burn in a homogeneous manner. After 1000 days, large discrepancies can be observed regarding the densities between regions next to a fuel block or next to a reflector block. This heterogeneity is much less severe if the depletion is performed over a single-block model. Thus, the use of a traditional scheme where the concentrations are calculated in the lattice code seems to be difficult to reconcile with the heterogeneity of the number density evolution. Therefore, if an in-core depletion model is used, it would be favorable that the block near the reflector is meshed into several small regions were the number densities are calculated using the local flux, to allow calculation of different macroscopic cross sections for each region. Otherwise, a macroscopic cross section applied to the entire block and based on an average of the densities may not take into account the high heterogeneity of those blocks. The typical size of an element of the mesh may have to be smaller than the typical distance of variation of the flux to allow one to consider a constant flux over the element while still preserving the overall density changes across a peripheral block.

Besides the heterogeneity of burning inside the core, one should also pay attention to the fact that the burnup of an assembly may not necessarily be a sufficient prediction of nuclides densities or possibly cross sections. It is very likely that because the peripheral blocks exhibit a different spectrum than the one of the central fuel blocks, there may be differences in the composition at the same burnup. This remark tends to imply the use of supercells to correct the spectrum while the depletion takes place at the lattice level, too.

The choice of not using the supercells may be less attractive once burnable poisons and control rods are added. Moreover, if the blocks are divided into small regions in the core calculation, then it might be interesting to recover microscopic cross sections corresponding to each small domain using supercells. Such a calculation scheme would certainly require an improvement of the speed of the MOC solver in DRAGON to be considered a viable option.

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

108   March 2011

Given the previous remarks, the solution of coupling the lattice calculation and the core calculation with net currents at the interfaces of each block may greatly improve the fidelity of the simulation and facilitate both depletion analysis and cross-section generation for this reactor by directly solving the two problems. This may be a project worth investigating for future work.

5.   CONCLUSION

5.1   Summary of Results

The project of a Deep Burn high-temperature reactor pursued by INL requires accurate neutronic simulations of the core behavior in all situations to meet licensing requirements from the safety authorities, with control rods, burnable poisons, depletion of the fuel, and also in accidental conditions where transient analysis must be performed. Within this framework, the influence of the reflector was evaluated in the calculation scheme. The core has been simplified by removing burnable poisons and control rods. Only steady-state calculations were performed and in cold conditions.

Compared to the usual procedures used for light water reactor analysis, two major differences have been identified:

- The annular design of the core leads to a high-peaked shape of the flux at both interfaces between fuel and graphite reflectors

- The graphite reflector thermalizes the spectrum, and this spectral variation is noticeable on the first half of the peripheral fuel blocks.

The spectral change as well as the peaked shape of the flux in the half of the fuel adjacent to the reflector raise issues for the lattice physics because a priori, a lattice calculation over a single block may not yield a spectrum representative enough of the real spectrum of the peripheral blocks. To assess that, two paths have been studied to prepare the cross sections. The first one considered only a single-block model at the lattice level, while the second introduced supercells to take into account some of the effects of the reflector. Both calculation paths provide homogenized cross sections condensed to several group structures to allow evaluation whether keeping more groups at the core level may compensate for having introduced error in the cross sections for peripheral blocks at the lattice stage.

The results show that with the MOC in DRAGON, both paths lead to good predictions of the eigenvalue and of the averaged fission rates of each fuel block in the core if enough groups are kept in INSTANT (e.g., 26 groups). The supercell path produces the best results compared to the equivalent calculations using the single-block path. A K_{eff} within ±20 pcm from MCNP was obtained in both cases. The fission rate errors are within ±1% with the supercells model, and within [-1%; + 1, 88%] with the single-block path, compared to the values from MCNP. Calculations with the CP and interface currents method proved to be less accurate, especially for supercells. But considering the good agreement reached by the single-block path with MCNP and considering the significant calculation durations, the use of supercells may not appear as attractive for this simplified core.

5.2   Limitations of the Single-block Path

Several concerns remain and may play against a calculation using only the single-block model.

First, the introduction of burnable poisons and control rods may create geometric difficulties and spectral effects that may not be carried to the core model using a single-block model, even with many groups in the core calculation. It is especially important to correctly model the burnable poison effects because they strongly affects the local flux.

Perhaps still the main challenge remains the depletion calculation in the code. Trials on a supercell have shown that the number densities across a block partially surrounded by reflector vary in significant proportions, up to 45% between two sides of the block for certain isotopes. In addition, the neutron spectrum also changes significantly depending on whether the reflector is close or not to the considered fuel cell. Thus, two phenomena are happening in those blocks: a high variation of the fission rates, which tends to create a heterogeneous burning in the block; and a spectral variation, which may lead for the same burnup to a different fuel composition from what is achieved in the center or on a single block.

5.3   Perspectives and Future Research Directions

The burnup and composition dependence to the spectrum seems to be the next important feature to be studied to provide scientific basis for deciding what depletion scheme can be used on the VHTR. Adding burnable poisons and control rods to the calculation will allow completion of the study and provide a basis for developing a complete calculation scheme for this reactor. Incidentally, the influence of temperature on cross sections still has to be captured, even if it is not expected to change the fundamental conclusions of this work.

The path forward toward a complete calculation scheme may require some improvements in DRAGON. The exact modeling of the geometry without the jagged boundary at the periphery of the blocks may be necessary. This implementation may be completed in the next years in the department. The self-shielding on homogenized cross sections would be helpful for performing depletion on supercells. More generally, the memory issues and long computation times are limitations of DRAGON for use in further studies for the prismatic reactor, especially if supercells prove to be required once the complete calculation scheme is defined. Some improvement of the acceleration techniques in the MOC as well as the parallelization of this method in the code may resolve these issues. Finally, INSTANT is currently missing depletion capabilities and easy transfer of cross sections with DRAGON. The development of a combined code with shared meshes may be an interesting project as well.

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

March 2011 111

6. REFERENCES

1. Wang, Y. *INSTANT: User Manual*, INL, Idaho Falls, USA, August 2010.

2. Hébert, A., *Applied Reactor Physics*. Presses Internationales Polytechnique, 2009.

3. Reuss, P., *Précis de neutronique*. L'Editeur: EDP Sciences, 2003.

4. Massimo, L., *Physics of high-temperature reactors*. Pergamon Press, January 1976.

5. Basdevant, J. and J. Dalibard, *Mécanique quantique*. Éditions de l'École Polytechnique, 2002.

6. Sanz, M. N. and B. Salamito, *Physique tout-en-un PC-PC**. Dunod, 2009.

7. Le Tellier, R., "Développement de la méthode des caractéristiques pour le calcul de réseau," Ph.D. dissertation, École Polytechnique de Montréal, December 2006.

8. Rozon, D., *Introduction à la cinétique des réacteurs nucléaires*. Presses Internationales Polytechnique, avr 1992.

9. Marleau, G., A. Hébert, and R. Roy, "A user guide for DRAGON Version4," École Polytechnique de Montréal, Tech. Rep. IGE-294, October 2009.

10. Bell, G. I. and S. Glasstone, *Nuclear reactor theory*. Van Nostrand Reinhold Co., 1970.

11. Hébert, A., "Development of the Subgroup Projection Method for Resonance Self-Shielding Calculations," *Nuclear Science and Engineering*, vol. 162, no. 1, pp. 56–75, 2009.

12. Livolant, M. and F. Jeanpierre, "Auto-protection des résonnances dans les réacteurs nucléaires. Application aux isotopes lourds." Commissariat à l'Énergie Atomique, France, Tech. Rep. CEA-R-4533, 1974.

13. Cullen, D. E., "Application of the probability table method to multigroup calculations of neutron transport," *Nuclear Science and Engineering*, vol. 55, no. 4, pp. 387–400, 1974.

14. Ribon, P. and J. M. Maillard, "Les tables de probabilité. Application au traitement des sections efficaces pour la neutronique." 1986, note CEA-N-2485.

15. Smith, K. S., "Spatial Homogenization Methods for Light Water Reactor Analysis," Ph.D. dissertation, Massachusetts Institute of Technology, June 1980.

16. Marleau, G., "Calcul neutronique des reacteurs," 2010, cours de maîtrise ENE6103.

17. Leppännen, J., "Development of a New Monte Carlo Reactor Physics Code," Ph.D. dissertation, Helsinki University of Technology, 2007.

18. Hfaiedh, N. and A. Santamarina, "Determination of the Optimized SHEM Mesh for Neutron Transport Calculations," in *Proc. Topl. Mtg. Mathematics and Computations, Supercomputing, Reactor Physics and Nuclear and Biological Applications*, September 2005.

19. Hfaiedh, N., "Nouvelle Méthodologie de Calcul de l'Absorption Résonnante," Ph.D. dissertation, Université Louis Pasteur, September 2006.

20. Hfaiedh, N. and A. Santamarina, "Refinement of the Santamarina-Hfaiedh Energy Mesh between 22.5 eV and 11.4 keV," in *Proc. Int. Conf. Physics of Reactors*, September 2008.

21. Wang, Y., *Equations of PN-hybrid-FEM for the Multigroup Transport Equation*, INL, Idaho Falls, USA, July 2010.

22. Lee, C. H., Z. Zhong, T. A. Taiwo, W. S. Yang, M. A. Smith, and G. Palmiotti, "Status of Reactor Physics Activities on Cross Section Generation and Functionalization for the Prismatic Very High Temperature Reactor, and Developpement of Spatially-Heterogeneous Codes," Argonne National Laboratory, Argonne, USA, Tech. Rep. ANL-GenIV-075, August 2006.

23. Lee, C. H., Z. Zhong, T. A. Taiwo, W. S. Yang, H. S. Khalil, and M. A. Smith, "Enhancement of REBUS-3/DIF3D for Whole-Core Neutronics Analysis of Prismatic Very High Temperature Reactor (VHTR)," Argonne National Laboratory, Argonne, USA, Tech. Rep. ANL-GenIV-076, September 2006.

24. Taiwo, T. A. and T. K. Kim, "Evaluation of the DRAGON Code for VHTR Design Analysis," Argonne National Laboratory, Argonne, USA, Tech. Rep. ANLGenIV-060, September 2005.

25. Ortensi, J., J. Cogliati, M. Pope, J. Bess, R. Ferrer, A. Bingham, and A. Ougouag, "Deterministic Modeling of the High Temperature Test Reactor," Idaho National Laboratory (INL), Tech. Rep., 2010.

26. Hébert, A., "A Collision Probability Analysis of the Double-Heterogeneity Problem," *Nuclear Science and Engineering*, vol. 115, no. 2, p. 177, 1993.

27. Henry, A., *Nuclear-reactor analysis*. MIT press Cambridge, MA, 1975.

28. Pope, M.A., J. Ortensi, and A.M. Ougouag, "Investigation of Supercells for Preparation of Homogenized Cross-Sections for Prismatic Deep Burn VHTR Calculations," In *Proc. of HTR 2010*, Prague, Czech Republic, October 18-20, 2010.

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

March 2011 113

Appendix A

Geometric Relations in a Hexagon

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

114   March 2011

Appendix A

Geometric Relations in a Hexagon

Useful geometric relations for hexagons follow:[c]
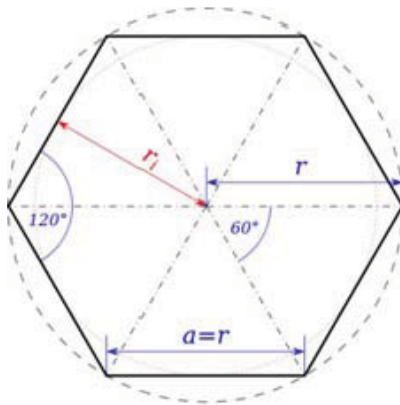
Figure A-1. Geometric relations in a hexagon.

$$\hspace{3cm} \text{(A-1)}$$

$$\overline{} \hspace{3cm} \text{(A-2)}$$

$$\hspace{3cm} \text{(A-3)}$$

$$\overline{} \hspace{3cm} \text{(A-4)}$$

---

c.   The figure is taken form Wikipedia: http://fr.wikipedia.org/wiki/Hexagone

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

116

March 2011

Appendix B

Formula to Calculate the Number of Hexagons in a 1/12th Core

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

118

March 2011

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

March 2011

Appendix B

Formula to Calculate the Number of Hexagons in a 1/12th Core

Say that N is the number of hexagons on a row from the center to the periphery of a hexagonal array of hexagons and T is the total number of hexagons on the one-twelfth geometry.

First case: suppose N odd:

$\exists p \in \mathbb{N} \,/N = 2p + 1$

$$\Longrightarrow T = \sum_{i=0}^{p}(2i + 1) = \sum_{i=o}^{p} 1 + 2\sum_{i=0}^{p} i = p + 1 + 2\frac{p(p+1)}{2} = p^2 + 2p + 1 = (p + 1)^2 \qquad \text{(B-1)}$$

Then replacing p in the previous formula using $p = \frac{N-1}{2}$, we find:

$$T = \left(\frac{N - 1}{2} + \frac{2}{2}\right)$$

$$\Longleftrightarrow \; T = \frac{(N+1)^2}{4} \text{ with N odd.} \qquad \text{(B-2)}$$

Second case: suppose N even:

$\exists p \in \mathbb{N} \,/N = 2p$

$$\Longrightarrow \; T = \sum_{i=o}^{p}(2i) = p(p + 1) \qquad \text{(B-3)}$$

Then replacing p in the previous formula using $p = \frac{n}{2}$, we find:

$$T = \frac{N}{2}\left(\frac{N}{2} + 1\right) = \frac{N^2}{4} + \frac{N}{2} = \frac{N}{4}(N + 2)$$

$$\Longleftrightarrow T = \frac{N(N+2)}{4} \text{ with N even.} \qquad \text{(B-4)}$$

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

120

March 2011

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

March 2011   121

Appendix C

Homogenized Graphite Density Calculation over a Single Block

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

122   March 2011

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

March 2011   123

Appendix C

Homogenized Graphite Density Calculation over a Single Block

This appendix explains the calculation of the graphite average densities over the real block and over the DRAGON single-block model. This is necessary to determine the density of the graphite in the J-cells, which alleviates the lack of a proper geometry in DRAGON, but also to build correct macroscopic cross sections for the graphite in INSTANT.

Thanks to the symmetry, the calculation is performed for a $1/12^{th}$ block. This allows comparison of some parts of the calculation with what DRAGON is calculating by itself. Thus, the geometry contains:

- 17.5 F cells

- 8.5 C cells

- $\frac{1}{2}$ K cell

- $1 + \frac{1}{12}$ G cell

- The boundary area S_b or 5.5 J-cells, depending on the model.

The normal graphite density d_{GRA} is set to $d_{GRA} = 8.774.10^{-2}.10^{24}$ atoms of graphite.cm^{-3}. Basically, the number of atoms and the global area for each geometry are computed, along with the homogenized surface densities in both geometries. The density in the J-cells is then adjusted depending on what need to be conserved: the number of atoms, or the global average graphite density.

C-1. INTERNAL DOMAIN: 10 RINGS

This domain is the same for the two geometries. For a better understanding, it is first supposed that the cells are prismatic (3-D) and not hexagonal (2-D). We define "h", the height of a hexagonal prism.

C-1.1 F Cells

The F cells are composed of an internal tube pellet with a matrix filled with TRISO particles and graphite, an annular region filled with helium and a surrounding region of graphite. The following notations are used:

- R_1 and R_2: the radii of the embedded tubes

- $\mathcal{A}$: the area of the hexagonal basis

- a: the side of the hexagonal basis

- η: the packing fraction

- V_T: the volume occupied by all TRISO particles

- V_R: the volume of the matrix.

The packing fraction corresponds to the volume fraction of TRISO particles present in the matrix volume. We have: $\eta = V_T/V_R$.

A TRISO particle is made up of a sphere with five layers having the following radii and with following graphite densities:

Radius (cm)	Graphite volume density $(.10^{24}$ atoms.cm$^{-3})$
$r_1 = 0.01$ cm	$d_{GRA1} = 0.0$
$r_2 = 0.022$ cm	$d_{GRA2} = 5.265.10^{-2}$
$r_3 = 0.026$ cm	$d_{GRA3} = 9.526.10^{-2}$
$r_4 = 0.0295$ cm	$d_{GRA4} = 0.0$
$r_5 = 0.0335$ cm	$d_{GRA5} = 9.526.10^{-2}$

The surrounding matrix has a density of $d_{GRA-F} = 8,524.10^{-2}.10^{24}$ atoms.cm^{-3}.

Let N be the number of TRISO particles contained in the matrix. The packing fraction is defined by:

$$\eta = \frac{\frac{4}{3}\pi r_5^3 \cdot N}{\pi R_1^2 h}$$

Thus we have:

$$N = \eta \cdot \frac{\pi R_1^2 h}{\frac{4}{3}\pi r_5^3} : \text{ number of TRISO particles in the matrix}$$

Let N_T be the total number of graphite atoms in a single TRISO particle.

$$N_T = \frac{4}{3}\pi \left(r_2^3 - r_1^3\right) d_{GRA2} + \frac{4}{3}\pi \left(r_4^3 - r_3^3\right) d_{GRA3} + \frac{4}{3}\pi \left(r_5^3 - r_4^3\right) d_{GRA5}$$

The total number of graphite atoms N_F contained in an F cell is composed of three terms: the graphite atoms contained in all the TRISO particles, those contained in the matrix, and those located in the external surrounding volume.

$$N_F = \eta \cdot \frac{\pi R_1^2 h}{\frac{4}{3}\pi r_5^3} \cdot N_T + (1 - \eta) \cdot \pi R_1^2 h \cdot d_{GRA-F} + (\mathcal{A} - \pi R_2^2) h \cdot d_{GRA}$$

The height h can be factored out:

$$N_F = h \cdot \left(\eta \cdot \frac{\pi R_1^2}{\frac{4}{3}\pi r_5^3} \cdot N_T + (1 - \eta) \pi R_1^2 \cdot d_{GRA-F} + (\mathcal{A} - \pi R_2^2) \cdot d_{GRA}\right)$$

Finally N_{totF} is the total number of atoms contained in all F cells of the geometry:

$$N_{totF} = 17,.5 \, N_F$$

C-1.2   C Cells

These cells contain a tube filled with helium surrounded by graphite with the normal density d_{GRA}. Calling R the radius of the embedded tube, and N_C the number of graphite atoms in one cell of this type. $R = 0,794$ cm.

$$N_C = h \cdot (\mathcal{A} - \pi R^2) \cdot d_{GRA}$$

Calling N_{totC} the total number of atoms contained in all C cells of our geometry.

$$N_{totC} = 8,.5 \, N_C$$

C-1.3 K Cells

These cells contain a tube filled with helium surrounded by graphite with the normal density d_{GRA}. Calling r the radius of the embedded tube, and N_K the number of graphite atoms in one cell of this type. $r = 0,635$ cm.

$$N_K = h \cdot (\mathcal{A} - \pi r^2) \cdot d_{GRA}$$

Calling N_{totK} the total number of atoms contained in all cells of the geometry:

$$N_{totK} = 8.5 \, N_K$$

C-1.4 G Cells

These cells are simple hexagons with normal graphite density. Calling N_G and N_{totG} the number of graphite atoms in one G cell and in all G cells of our geometry:

$$N_G = h \cdot \mathcal{A} \cdot d_{GRA}$$

$$N_{totG} = \left(1 + \frac{1}{12}\right) N_G$$

It is then possible to compute the total number of graphite atoms located in the internal prism N_{int}:

$$N_{int} = N_{totF} + N_{totC} + N_{totK} + N_{totG}$$

Note that "h" is in factor in front of every term.

C-2.   BOUNDARY REGION AND GLOBAL VOLUME

C-2.1 DRAGON Model

The model use 5.5 J-cells with adjusted graphite density d_J that form the boundary region. The total number of atoms contained in the J-cells is calculated as follows:

$$N_{totJ} = 5.5 \, h \cdot \mathcal{A} \cdot d_J$$

Thus the total number of atoms in the DRAGON model is equal to:

$$N_{DRAGON} = N_{int} + 5.5 \, h \cdot \mathcal{A} \cdot d_J$$

To calculate the global volume in the DRAGON model, the number of cells are summed, and this is multiplied by the area of a hexagon $\mathcal{A}$. Calling S_{DRAG} the area of the base of the prismatic block and V_{DRAG} the volume of the prismatic block, we have:

$$S_{DRAG} = \left(17.5 + 8.5 + 0.5 + 1 + \frac{1}{12} + 5.5\right) * \mathcal{A}$$

$$V_{DRAG} = h \cdot S_{DRAG}$$

Again, "h" is in factor in V_{DRAG}.

It is now possible to calculate the density for the DRAGON model:

$$d_h = \frac{N_{DRAG}}{V_{DRAG}}$$

As shown previously, h is in factor in N_{DRAG} and V_{DRAG}. It will be simplified by the division, which leads to this quite important conclusion: in the case of a tube pellet with spherical particles, the global homogenized density is independent from the height h of the tube.

This is why it is possible to redefine everything in terms of numbers of atoms over a hexagon: one can divide the previous quantities of atoms by h.

$$N_T = \tfrac{4}{3}\pi \left(r_2^3 - r_1^3\right) \cdot d_{GRA2} + \tfrac{4}{3}\pi \left(r_4^3 - r_3^3\right) \cdot d_{GRA3} + \tfrac{4}{3}\pi \left(r_5^3 - r_4^3\right) \cdot d_{GRA5}$$

$$N_{totF} = 17,5 \left(\eta \cdot \frac{\pi R_1^2}{\tfrac{4}{3}\pi r_5^3} \cdot N_T + (1-\eta)\, \pi R_1^2 \cdot d_{GRA-F} + \left(\mathcal{A} - \pi R_2^2\right) \cdot d_{GRA} \right)$$

$$N_{totC} = 8,5 \left(\mathcal{A} - \pi R^2\right) \cdot d_{GRA}$$

$$N_{totK} = \tfrac{1}{2}\left(\mathcal{A} - \pi r^2\right) \cdot d_{GRA}$$

$$N_{totG} = \left(1 + \tfrac{1}{12}\right) \mathcal{A} \cdot d_{GRA}$$

$$N_{int} = N_{totF} + N_{totC} + N_{totK} + N_{totG}$$

$$N_{totJ} = 5.5\, \mathcal{A} \cdot d_J$$

$$N_{DRAG} = N_{int} + N_{totJ}$$

This amounts to say that all densities are meant as surface densities instead of volume densities. Thus the above formulae are homogeneous. N_T becomes the number of atoms of graphite per height in one single TRISO particle because the d_{GRA2}, d_{GRA3} and d_{GRA5} become surface densities. η remains the volumetric packing fraction, without dimension. Its definition is no more straightforward because the volumes of the pellet and of the TRISO spheres are no more defined over a plane surface. In fact, the volume of the TRISO is projected on the plane surface.

The term $\eta \cdot \dfrac{\pi R_1^2}{\tfrac{4}{3}\pi r_5^3} \cdot N_T$ has no dimension and can be interpreted as the average number of graphite atoms contained in the TRISO particles and located on a right-plane section of the tube. It becomes clearer when it is completely written:

$$\eta \cdot \frac{\pi R_1^2}{\tfrac{4}{3}\pi r_5^3} \cdot N_T = \eta \cdot \pi R_1^2 \sum_{i=1}^{5} \left(\underbrace{\frac{\tfrac{4}{3}\pi\left(r_i^3 - r_{i-1}^3\right)}{\tfrac{4}{3}\pi r_5^3} \cdot d_{GRAi}}_{\phi_i} \right)$$

where $\emptyset_i$ is the volumetric fraction of the Layer i in the TRISO particle.

The term in the sum corresponds to the average number of atoms of graphite of a TRISO particle per square centimeter. It can be seen as the average number of atoms counted on any section of a sphere.

Thus, multiplied by πR_1^2, it becomes the maximum number of atoms that could be counted on average on a right-plane section of the pellet if the pellet were totally filled with TRISO particles. This enables one to give a sort of definition of η in two dimensions: η would be the ratio between the number of graphite atoms belonging to TRISO particles located on a section, divided by the maximum number of atoms that could be placed on the same section.

$$\eta = \cfrac{\text{True number of graphite atoms belonging to particles over a section}}{\underbrace{\pi R_1^2}_{\text{Area of the plane right section}} \cdot \underbrace{\sum_{i=1}^{5}\left(\cfrac{\frac{4}{3}\pi(r_i^3 - r_{i-1}^3)}{\frac{4}{3}\pi r_5^3}\cdot d_{GRAi}\right)}_{\text{Average number of graphite atoms.cm}^{-2}\text{ in 1 particle}}}$$

The numerical calculation gives:

$$
\begin{aligned}
N_T &= 9.64977237588257.10^{-6}.10^{24} \text{ graphite atoms in 1 TRISO particle.}\\
N_{totF} &= 4.48140732535334.10^{24} \text{ graphite atoms in all F cells.}\\
N_{totC} &= 0.805688501750402.10^{24} \text{ graphite atoms in all C cells.}\\
N_{totK} &= 0.0787080406765923.10^{24} \text{ graphite atoms in all K cells.}\\
N_{totG} &= 0.290942564134322.10^{24} \text{ graphite atoms in all G cells.}\\
N_{int} &= 5.65674643191466.10^{24} \text{ graphite atoms in the internal area.}\\
S_{DRAG} &= 101,2643982757 \text{ cm}^2.
\end{aligned}
$$

C-2.2   MCNP Model

In the MCNP model, it is easier to compute the boundary area between the internal cells and the frontier by subtracting the internal area from the total area.

Calling S the total area of the large hexagon, $p = 36,0$ the pitch, and s the side of the large hexagon:

$$s = \frac{P}{\sqrt{3}} = 20.78461 \text{ cm}^2$$

$$S = \frac{3\sqrt{3}}{2}s^2 = 1122.3689 \text{ cm}^2$$

There are 331 internal hexagons, which occupy an area equal to:

$$\mathcal{A}_h = 331 \cdot \frac{\sqrt{3}}{2}(1.88)^2 = 1013.151 \; cm^2$$

Thus, if we call S_b the boundary area for $1/12^{th}$ of the block:

$$12\,S_b = S - \mathcal{A}_h = 109.217 \text{ cm}^2$$

Thus the area of the boundary region and the number of atoms in it is found to be

$$S_b = 9.1014167 \text{ cm}^2$$

$$N_b = S_b \cdot d_{GRA} = 0.798542.10^{24} \text{ graphite atoms.}$$

Finally, the total number of graphite atoms in the MCNP $1/12^{th}$ block model and the corresponding area are equal to:

$$N_{MCNP} = N_{int} + N_b = 6.455288.10^{24} \text{ graphite atoms.}$$

$$S_{MCNP} = \frac{1}{12}S = 93.53074 \text{cm}^2.$$

C-3.   CONSERVATION OF THE GLOBAL HOMOGENIZED DENSITY

In this first example, we are searching d_J so that we keep the same global homogenized surface density d_h in both cases:

$$\frac{N_{MCNP}}{S_{MCNP}} = \frac{N_{DRAG}}{S_{DRAG}} \Leftrightarrow d_J = \frac{1}{5,5\,\mathcal{A}} \left(\frac{S_{DRAG}}{S_{MCNP}} N_{MCNP} - Nint \right)$$

The numerical application gives:

$$d_J - 7,.9139.10^{-2}.10^{24} \; graphite \; atoms.cm^{-2} \Leftrightarrow d_J = 0.90198 \cdots d_{GRA}$$

$$d_h = \frac{N_{DRAG}}{S_{DRAG}} = 6.901782.10^{-2}.10^{24} \; graphite \; atoms.cm^{-2}$$

The difference between the global homogenized densities of the MCNP model and the DRAGON model is equal to $1.55.10^{-15}$, which is beyond the precision of DRAGON.

The calculation in DRAGON is consistent with the above calculation: we obtain
$d_h = 6,90178201.10^{-2}.10^{24}$ graphite atoms.cm^{-2} in DRAGON instead of $d_h = 6.901782601.10^{-2}.10^{24}$ graphite atoms.cm^{-2} in the density calculation.

This J-cell density was useful for building directly the right macroscopic cross sections in DRAGON for INSTANT. The hope was that the different boundaries would not appreciably change the neutron energy spectrum compared to what would give a calculation over the real block. But in fact, to provide directly macroscopic cross sections for INSTANT with the correct densities, all isotope's densities should be adjusted in every region by a factor of about 8.2%, corresponding to the ratio of S_{DRAGON}/S_{MCNP}, including the densities of the fissile isotopes. This would significantly change the composition of the fuel and would likely lead to quite wrong cross sections for INSTANT.

Once the capability was developed to change the density during preparation of the macroscopic cross sections for INSTANT, this capability was no more of interest. Indeed, the most logical approximation to mimic the real block calculation is to conserve the number of atoms for each isotope between the two block models, to recover the microscopic cross sections and to establish the new macroscopic cross sections for INSTANT taking the true densities into account.

The resulting J-cell density has been determined hereafter. Other isotope's densities over a real block have also been determined and are given in next appendix.

C-4.   CONSERVATION OF THE NUMBER OF GRAPHITE ATOMS IN THE BOUNDARY REGION

It was preferred to conserve the number of atoms in the boundary regions between the MCNP model and the DRAGON model. This is equivalent to conserving the total number of graphite atoms, because the internal number of atoms is the same in both cases. This choice can be justified by the fact that such a relation would produce a spectrum for the neutrons coming in the boundary fuel rings that should better reflect the spectrum generated in MCNP. This appears to be the most logical approximation to deal with the jagged boundary limitation in DRAGON.

The conservation relation is written as:

$$N_{MCNP} = N_{DRAG} \iff N_{int} + N_b = N_{int} + N_J$$

$$\iff N_b = d_J \cdot V_J$$

$$\iff d_J = \frac{N_b}{V_J}$$

We find:

$d_J = 4.74337.10^{-2}.10^{24}$ graphite atoms.cm^{-2}, or otherwise:

$d_J = 0.54061738 \cdot d_{GRA}$

This leads to a global homogenized density in DRAGON model of:

$d_{h-DRAG} = 6.3746.10^{-2}.10^{24}$ graphite atoms. cm^{-2}

The global homogenized density in MCNP model remains the same:

$d_{h-MCNP} = 6.901782.10^{-2}.10^{24}$ graphite atoms. cm^{-2}

So we have:

$d_{h-DRAG} = 0.923629 \cdot d_{h-MCNP}$

Thus, the macroscopic cross sections passed to INSTANT with this method without any other treatment will have lower density than the expected one. Therefore, it necessary to recover the microscopic cross section from such a calculation, and to calculate the macroscopic cross section for INSTANT with the right density d_{h-MCNP}.

In the supercell models in DRAGON, each J-cell is shared between two blocks, so that the core geometry looks approximately like the real one. Therefore, the J-cell density has to be multiplied by 2 to reflect the right boundary volume. This remark is valid only when trying to conserve the total number of atoms over the whole lattice comparing to MCNP. Therefore, in this case, the following is used:

$d_J = 9.49.10^{-2}.10^{24}$ graphite atoms.cm^{-2}, or otherwise:

$d_J = 1.08 \cdot d_{GRA}$

Naturally, this leads to a different homogenized density comparing to the MCNP one, but this is not important as only the microscopic cross sections are recovered from those calculations:

$d_{h-DRAG} = 7.16.10^{-2}.10^{24}$ graphite atoms.cm^{-2}, that is to say:

$d_{h-DRAG} = 1.037885 \cdot d_{h-MCNP}$

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

130   March 2011

Appendix D

Density Calculation for all Isotopes in INSTANT

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

132

March 2011

Appendix D

Density Calculation for all Isotopes in INSTANT

As seen before, the areas of the blocks in INSTANT and DRAGON are different, which leads to different densities and prevents the direct passage of homogenized macroscopic cross sections computed in DRAGON.

It is necessary to conserve the number of atoms between the two calculations. Only graphite is located in the J-cells. Therefore, for all the other isotopes, the correct way to pass cross sections to INSTANT is to keep the same number of atoms as in DRAGON, because the inside part of the blocks does not change between the two models.

As the J-cells introduce an extra area at the periphery, the densities calculated by DRAGON will be lower than the correct ones. Thus, they have to be corrected before calculating the macroscopic cross sections for INSTANT.

The following can be done:

$$d_{INSTANT}^{isotope} = d_{DRAGON}^{isotope} \cdot \frac{S_{DRAGON}}{S_{INSTANT}}$$

For explanation purposes, the full calculation were repeated as for the graphite above, to verify the densities calculated in DRAGON. The calculation was done over a 1/12th block and the work done for the graphite was reused. This leads naturally to the same results.

The surface in INSTANT is equal to:

$$S_{INSTANT} = 93,.53 \text{cm}^2.$$

The model contains four kind of isotopes:

- The isotopes located in the kernel of the TRISO particles

- The isotopes located in the peripherial shells of the TRISO particles

- The helium located in the annular tube surrounding the fuel pellet and in the coolant channels

- The graphite located in the TRISO particles, in the matrix of the fuel cells, in the graphite cells, and in all areas surrounding coolant channels.

For the isotopes located in the TRISO's kernel, we compute N_T, the number of atoms inside one TRISO particle, then the total number of atoms inside one fuel cell N_{totF}, and finally the homogenized density over the block (see Table D-1).

$$N_T = \frac{4}{3}\pi r_1^3$$

$$N_{totF} = 17,5 \times \left(\eta \cdot \frac{\pi R_1^2}{\frac{4}{3}\pi r_5^3} \cdot N_T \right)$$

$$d_h^{isotope} = \frac{N_{totF}}{S_{INSTANT}} \tag{D-1}$$

Table D-1. Average densities of the fuel isotopes in INSTANT.

Isotope	Density ($.10^{24}$ $atoms.cm^{-3}$)
O16	$4.703781.10^{-5}$
NP27	$1.613876.10^{-6}$
PU38	$6.854202.10^{-7}$
PU39	$1.165342.10^{-5}$
PU40	$5.390898.10^{-6}$
PU41	$2.053928.10^{-6}$
PU42	$1.138833.10^{-6}$
AM41	$6.535032.10^{-7}$
AM42	$4.649703.10^{-9}$
AM43	$3.241537.10^{-7}$

For isotopes located in the shells of the TRISO particles (see Table D-2), the calculation is similar.

$N_T = \frac{4}{3}\pi(r_i^3 - r_{i-1}^3)$ with $i = 4$ for layer 4 and $i = 5$ for layer 5

$$N_{totF} = 17,5 \times \left(\eta \cdot \frac{\pi R_1^2}{\frac{4}{3}\pi r_5^3} \cdot N_T \right)$$

$$d_h^{isotope} = \frac{N_{totF}}{S_{INSTANT}} \tag{D-2}$$

Table D-2. Average densities of the TRISO's shells isotopes in INSTANT.

Isotope	Density ($.10^{24}$ $atoms.cm^{-3}$)
SI28	$3.779168.10^{-4}$
SI29	$1.918774.10^{-5}$
SI30	$1.264588.10^{-5}$
C12	$4.096818.10^{-4}$

Finally, helium (Table D-3) is located in the coolant tubes and in the annular region surrounding the fuel pellet.

$$N_{He} = \sum_{i\epsilon \text{ F,C,K cells}} (\pi R_i^2 - \pi R_{i-1}^2) d_i \text{ and } d_{He} = \frac{N_{He}}{S_{INSTANT}}$$

Table D-3. Average density of helium in INSTANT.

Isotope	Density ($.10^{24}$ $atoms.cm^{-3}$)
HE4	$1.372031.10^{-4}$

For the graphite (Table D-4), the calculation was already done in the previous appendix to calculate the J-cell density. We have:

Table D-4. Average density of graphite in INSTANT.

Isotope	Density ($.10^{24}$ $atoms.cm^{-3}$)
GRAB	$6.901782.10^{-2}$

Appendix E

Comparison between P_1, P_3, P_5

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

136

March 2011

Appendix E

Comparison between P_1, P_3, P_5

In INSTANT, it is possible to choose how many angles are treated. This appendix shows a comparison between P_1, P_3 and P_5 to complete the explanations of Section 3.2.2.

The first two tables concern P_1 and P_3 calculations. Table E-1 compares the K_{eff} and Table E-2 the fission map parameters obtained in INSTANT. The variability is significant. On the contrary, the same K_{eff} are calculated between P_3 and P_5 (see Table E-3). This demonstrates the convergence of the INSTANT model in P_3.

Table E-1. Single block path: INSTANT calculations in P_1 and P_3, eigenvalues.

No. of Groups in INSTANT	SB MOC DH295			
	P_1 scat 1		P_3 scat 1	
	Core K-eff	Deviation with MCNP (pcm)	Core K-eff	Deviation with MCNP (pcm)
295	1.25033	58	1.25010	40
26	1.25046	69	1.25023	50
23	1.24994	27	1.24970	8
12	1.24990	24	1.24968	6
10	1.25011	41	1.24987	22
9	1.24683	-221	1.24660	-240
6	1.24783	-141	1.24758	-162
4	1.20891	-3256	1.20891	-3256
2	1.21163	-3039	1.21162	-3039

Table E-2. Single block path: INSTANT calculations in P_1 and P_3, fission map comparison.

No. of Groups in INSTANT	SB MOC DH295									
	P_1 scat 1					P_3 scat 1				
	Max	Min	Av. μ	Std σ	AbsD	Max	Min	Av. μ	Std σ	AbsD
295	2.06	-1.13	-0.15	0.94	0.82	1.83	-0.98	-0.14	0.81	0.69
26	2.02	-1.14	-0.15	0.98	0.88	1.79	-0.99	-0.13	0.85	0.74
23	2.05	-1.18	-0.15	1.03	0.93	1.83	-1.04	-0.14	0.90	0.80
13	1.49	-1.01	-0.11	0.87	0.81	1.27	-0.91	-0.10	0.73	0.68
10	1.53	-1.05	-0.11	0.90	0.84	1.30	-0.96	-0.10	0.76	0.71
9	0.24	-0.30	-0.007	0.18	0.16	0.35	-0.53	-0.009	0.48	0.25
6	0.98	-0.80	-0.07	0.61	0.58	0.76	-0.70	-0.06	0.29	0.45
4	11.2	-14.3	0.98	9.36	8.92	11.2	-14.5	0.99	9.36	9.01
2	10.7	-14.2	1.00	9.22	8.77	10.8	-14.4	1.01	9.22	8.86

Table E-3 shows the K_{eff} obtained with P_3 and P_5 calculations in INSTANT.

Table E-3. Single block path: INSTANT calculations in P_3 and P_5, eigenvalues.

| No. of Groups | SB MOC DH295 | | Difference (pcm) |
	P_3 scat 1 Core K-eff	P_5 scat 1 Core K-eff	
295	1.25010	1.25010	0
26	1.25023	1.25023	0
23	1.24970	1.24971	-0.8
12	1.24968	1.24967	0.8
10	1.24987	1.24987	0
9	1.24660	1.24660	0
6	1.24758	1.24758	0
4	1.20891	1.20891	0
2	1.21162	1.21161	0.8

Appendix F

Group Structures

Final Report on Utilization of TRU TRISO Fuel as Applied to HTR Systems Part II: Prismatic Reactor Cross Section Generation

140   March 2011

Appendix F

Group Structures

Tables F-1, F-2, F-3, F-4, F-5, F-6, F-7, and F-8 give the boundaries of the intermediate group structures as they were determined by DRAGON. The boundaries are kept as close as possible to well-known group structures. The lowest boundary is fixed at $1.100027.10^{-4}$ eV for every group structure.

All of these group structures are based on boundaries given by publications, except the 10-group structure, which has been built based on the 12-group structure by removing two boundaries that were located inside one larger interval of the 26-group structure. This enables condensation from 26 groups to 10 groups, while it was not possible to go from 26 groups to 12 groups with their existing boundaries.

Group structures with 2, 4, 6, 12, and 26 groups are given in the IAEA report[F-1] and have been used by the French CEA or the German Research Center of Jülich. Note that the 12-group structure is in fact based on the 13-group structure proposed in the report, but the last group was automatically removed by DRAGON because it was too low in energy. The group structure with 23 groups comes from the technical report[23] of Argonne National Laboratory. The 9-group structure has been determined by General Atomics and is cited in Reference F-2. Once boundaries are given to DRAGON, the code finds the closest match with the internal boundaries of the 295-group structure. It explains the differences one may observe between the boundaries cited in those publications and the final boundaries chosen by the code and given in Tables F-1 to F-8.

Table F-1. 26-group structure, energy limits.

26-group Structure				
Group	1	2	3	4	5
E_{max} (eV)	1.9640E+07	6.7032E+06	3.3287E+06	5.7844E+05	9.4665E+04
Group	6	7	8	9	10
E_{max} (eV)	1.8585E+04	2.9962E+03	1.3436E+03	6.7729E+02	2.6830E+02
Group	11	12	13	14	15
E_{max} (eV)	1.2623E+02	5.9925E+01	2.7885E+01	1.3573E+01	7.9653E+00
Group	16	17	18	19	20
E_{max} (eV)	4.9585E+00	2.3301E+00	1.2509E+00	6.2500E-01	3.2501E-01
Group	21	22	23	24	25
E_{max} (eV)	1.9000E-01	1.2000E-01	7.6497E-02	4.7302E-02	1.4830E-02
Group	26				
E_{max} (eV)	7.1453E-03				

Table F-2. 23-group structure, energy limits.

23-group Structure					
Group number	1	2	3	4	5
E_{max} (eV)	1.9640E+07	3.3287E+06	1.3369E+06	4.9400E+05	9.4665E+04
Group number	6	7	8	9	10
E_{max} (eV)	6.7379E+04	7.4658E+03	3.5357E+02	4.0000E+00	1.4440E+00
Group number	11	12	13	14	15
E_{max} (eV)	1.0920E+00	1.0350E+00	9.6396E-01	8.2004E-01	4.7502E-01
Group number	16	17	18	19	20
E_{max} (eV)	3.9000E-01	3.2501E-01	2.7999E-01	2.3119E-01	1.6190E-01
Group	21	22	23		
E_{max} (eV)	1.3800E-01	8.9797E-02	4.7302E-02		

Table F-3. 12-group structure, energy limits.

12-group Structure					
Group	1	2	3	4	5
E_{max} (eV)	1.9640E+07	1.6507E+05	9.8249E+02	1.6562E+01	4.0000E+00
Group	6	7	8	9	10
E_{max} (eV)	2.0701E+00	1.2930E+00	7.2000E-01	6.2500E-01	3.9000E-01
Group	11	12			
E_{max} (eV)	3.0501E-01	1.0430E-01			

Table F-4. 10-group structure, energy limits.

10-group Structure					
Group	1	2	3	4	5
E_{max} (eV)	1.9640E+07	1.6507E+05	9.8249E+02	1.6562E+01	4.0000E+00
Group	6	7	8	9	10
E_{max} (eV)	2.0701E+00	7.2000E-01	3.9000E-01	3.0501E-01	1.0430E-01

Table F-5. 9-group structure, energy limits.

9-group Structure					
Group	1	2	3	4	5
E_{max} (eV)	1.9640E+07	1.6507E+05	9.0968E+02	1.7376E+01	3.8822E+00
Group	6	7	8	9	
E_{max} (eV)	2.3301E+00	1.2509E+00	8.2004E-01	1.19999E-01	

Table F-6. 6-group structure, energy limits.

6-group Structure					
Group number	1	2	3	4	5
E_{max} (eV)	1.9640E+07	1.6507E+05	1.6562E+01	2.0701E+00	6.2500E-01
Group	6				
E_{max} (eV)	1.0430E-01				

Table F-7. 4-group structure, energy limits.

4-group Structure				
Group number	1	2	3	4
E_{max} (eV)	1.9640E+07	9.4665E+04	2.7885E+01	1.7800E+00

Table F-8. 2-group structure, energy limits.

2-group Structure		
Group number	1	2
E_{max} (eV)	1.9640E+07	2.3301E+00

References

F-1. IAEA, "Evaluation of high temperature gas cooled reactor performance: Benchmark analysis related to initial testing of the HTTR and HTR-10," IAEA, Tech. Rep. IAEA-TECDOC-1382, November 2003.

F-2. Schmidt-Hoenow, M. and R. Rucker, "Analysis of VHTRC Physics Benchmarks," Argonne National Laboratory, Tech. Rep. DOE-HTGR-90363, Rev. 0, May 1993.