Politecnico di Torino

Politecnico di Torino

Corso di Laurea Magistrale in Ingegneria Energetica e Nucleare
A.A. 2021/2022
Marzo 2022

Study and application of the Enhanced Quasi-Static method with OpenFOAM®

Relatore
Prof.ssa Sandra Dulla

Candidato
Michele Fatti

A mio padre

Contents

Acknowledgments

Ringrazio i miei genitori per aver materialmente permesso questo percorso. Da mia madre ho ricevuto tutto l'affetto, il conforto e l'amore che un figlio possa ricevere che sono ciò che contano e che vanno al di là delle naturali divergenze quotidiane: le prime durano, queste no. Questa tesi è dedicata a mio padre e per lui scritta in Latex programma di cui rimase affascinato. Usarlo e scrivere, mi ha fatto sentire un po' più vicino a lui. Devo poi ringraziare mia sorella che posso solo guardare dal basso. La mia riconoscenza e stima non saranno mai abbastanza per ciò che ha fatto e fa per me. Ringrazio il nonno Masino per essere stato quella certezza di cui tutti abbiamo bisogno. Spero anche di rendere orgogliosi il nonno Nazzareno, la nonna Graziella e Luciana. Ringrazio Richi che più di tutti può capire le difficoltà che ho incontrato in questo percorso e presso il quale ho trovato e trovo da tempo un mutuo affetto fraterno. Un grazie all'Ale per i suoi preziosi e sinceri consigli il cui unico scopo era ed è il mio bene. Un grazie ai miei amici del liceo: la Giulia, il Biscio, la Fra, la Jenni, la Vale, la Marti, la Sara, la Mari e il Sabo. Lo scorrere del tempo non fa arrugginire il mio affetto per voi. Un pensiero va anche a Sirchi, Tiberio e Miozza ai quali mai riesco a dimostrare il bene che nutro per loro. Ringrazio l'Elena che mi ha fatto vedere la luce quando per me c'era solo buio. Ringrazio chi in questo percorso si è inserito solo all'ultimo ma che è riuscito comunque a dimostrarmi il suo affetto: Tommy, Giulio, Filippo e Paolo. Se poi ho resistito per tre anni e mezzo a Torino lo devo a tutti quelli che hanno reso quotidianamente piacevole il mio percorso universitario: Fabio, Beatrice, Andrea, Cristina, Alessandro, Alessandro "barba", Cecilia e Giacomo, senza il quale forse sarei ancora ai primi esami di magistrale. A loro devo aggiungere tutte le persone che hanno avuto la sfortuna di vivere con me alle quali rimango molto affezionato e Marco poiché senza il suo aiuto non avrei superato l'ultimo esame. Sarei poi ipocrita a non ringraziare la prof. Dulla e il dottor Abrate per la loro pazienza nella compilazione di questa tesi. Ringrazio il buon Dio che forse ha scritto la trama di tutta questa storia a volte ingarbugliata. Infine le centinaia di persone che non ho citato direttamente ma che anche in piccolissima parte hanno contribuito alla riuscita di questo percorso. Grazie!

Chapter 1

Introduction

1.1 OpenFOAM®

OpenFOAM® is the acronym for Open-source Field Operation And Manipulation. It is a collection of libraries [1] written in C++ and it is conceived for the solution of continuum mechanic problems. It is remarkable that is an open-source program, released under the general public licence (GPL).The final user has the possibility to run the software and to modify it [2] and this determines the great success of the tool that was created for computational fluid-dynamics (CFD). First works on this were made during nineties by Charlie Hill at Imperial College but only as post-processing software. Henry Wellers started the creation of software as computational program and with Hrvoje Jasak built it, sailing it. In 2004, there was the decision to give it as free and it got its actual name OpenFOAM®. Today the official version of OpenFOAM® is released every six months by ESI-OpenCFD Ltd but some forks (a software developed by an already present one by other programmers) are present. It is one of the biggest programm in C++ and one of the most known CFD tool [4].

OpenFOAM® contains many libraries that can be divided in two: *solvers* and *utilities*. The first ones are codes that solve the problems while the other ones are dedicated to the manipulation of data. So the pre-processing and post-processing can be considered as utilities while the part of code dedicated to solution is the solver. By its nature of free programme the user can create its own utilities and solvers, added to the ones already present in the package given by developers. These cover a wide range of utilization. For pre-processing OpenFOAM® allows the creation of mesh using the already present tool like *blockMesh* or *snappyHexMesh* but it is given the possibility to create it by others programmes. Obviously also data must be declared, specifying in case they are the computed variables the boundary and initial condition even if the problem is in steady-state [1]. Solvers cover a wide range of applicability. They are mainly conceived for CFD: there are methods that retrieve the basic equations,

turbulence models like RANS or DNS and also solvers for multiphase problems are present. To these it has to be added the possibility to perform also other problems, for example combustion, molecular dynamic, finance, electromagnetism, stress analysis and even the Monte Carlo method.

The powerful of OpenFOAM® is the already implemented capability to perform discretization of time, space and equation. The first one is quite easy because it consists in the break of the period in many time steps Δt. Regarding the space, it is written above the capability to create a mesh so to divide the space in small control volumes where data are stored. They are points, faces, cells and boundaries and they will be used by OpenFOAM® in order to create the discretization. Data are stored in centres of cells or at faces (internal or external) and finally also on vertices. The equations discretization occurs as classic formula $Ax = b$ where x is the vector that contains the variable, b the source vector and A the square matrix of coefficients. The first two contain the just cited values so the values of data at centre of cells in case of finite volume discretization. The matrix has in itself the coefficients to be multiplied by the variables inheriting inside the information by the mesh. The FV discretization occurs by the integration of volume or, in case of derivative, exploiting Gauss theorem [1]. This terms are then linearized and this is another advantage of OpenFOAM®. Indeed it allows the possibility to user to decide the scheme of linearization of the differential term. To make an example, for the time it is possible to use implicit or explicit Euler method or Crank Nicolson.

OpenFOAM®, as an opensource programme and for giving broad versatility, has some lacks. First of all it does not have a graphic interface user (GUI) from which the problem can be set. Mesh has to be defined by text or created by other software; variables need a text file where initial and boundary conditions are decided and the reading of the results, written in files, is in separated folders, needs another programme. It does not exist a complete manual that can illustrate meaning of function or applications: tutorials, fully access to all the codes and online forums can help to find the solution to problems but this requires more time. It is created to run in Linux so for Windows or Mac that are very popular it needs the installation of a virtual machine that in case of GUI slows the operations. Moreover all the commands must be given by the prompt.

So OpenFOAM® is a very powerful tool that from begin of two thousands become more and more important since its big versatility that allows user to set a wide range of problems and the free philosophy that is behind the concept. However the absence of a GUI and of a complete manual makes slower the utilization for the person.

1.2 Usage of OpenFOAM® in nuclear field as numerical tool

As it is written above OpenFOAM® was designed for fluid-dynamic computation and also now most of the solvers are dedicated to this. However the open access to all settings, so possibility to change the equations make it usable not only in CFD. Some applications have found space also in nuclear field such that on April 2019 a technical committee was established [5]. So in order to understand the capability and the possibility of OpenFOAM® some works about usage of it in nuclear reactor physics are reported.

The first one that has to be reported is the one by Fiorina et al. [6]. They present a multi-physics solver for 2D/3D reactors with a special application on fast reactor in particular on the sodium-cooled fast reactor (SFR). The solver is called Generalized Nuclear Foam (GeN-Foam) due to the capability to resolve different reactor configurations (for example also LWR), due to different physics involvement and possibility to work with fine or coarse mesh. The physics involved are: (I) solver for compressible and incompressible fluid motion based on k-ε equation for turbulence model with the insertion of a porous media to simulate the structures in the circuit; (II) solver for thermal-mechanic motion due to the increase of temperature; (III) multi-group diffusion sub-solver for neutrons able to exchange data with Monte Carlo Serpent-2 code; (IV) a finite difference solver for the representation of temperature in 1D geometry. Focusing on the neutron sub-solver the GeN-Foam is based on multi-group diffusion equations including also delayed neutrons. It is solved after the thermal mechanics solution taking into account that displacement of structure affects the neutron motion. It is considered a non-homogeneous medium so cross section depend on space and also effects of temperature and displacement of the structure are considered. Gen-Foam was also compared to Serpent-2 and to analytic results (for simplified configurations) to understand if it works properly and it is so. Good agreement was observed for the retrievement of k effective and for evaluation of feedback coefficients: the only exception is the coolant density effect that has a deep impact on the diffusion coefficients. Further comparisons are done with tool TRACE, simulating steady state and transient effects showing discrepancies in the power developed (so a quantity that can be directly connected to the neutron equation). These can be reduced to the different solutions method, point kinetic for TRACE and multi-group energy diffusion equation for Gen-Foam. So this application shows that OpenFOAM® is able to work also considering multiphysics, overcoming the problem to interface programs focused on different physical aspects.

From Aufiero et al. [7] there is a publication that concerns the calculation of the effective fraction of neutrons by different methods including deterministic one by the usage of OpenFOAM®. The beta effective computed is less than the physical one due to the role of importance that may decrease the value and, considering the case of study, this value differs also due to motion of fuel. Indeed the application is quite specific since it is treated a molten salt reactor (MSR): this gives a very impressive idea on the capability of OpenFOAM® and simulation of neutron behaviour. The equations implemented are multi-group diffusion equations in steady state with the adjoint of the displacing fuel since it is a molten salt reactor. So they slightly differ from the conventional ones due to displacement of precursors. This is the reason for OpenFOAM® choice since the current tools are not able to solve this equation that has the addition of a convective transport object. Moreover it could give another way to compute β_{eff} that is a quantity that actually is calculated with hypothesis that does not consider correctly the physical phenomena

Ma et al. created ntkFoam [8] that is a specific solver for neutron equation. Their work dates to 2020 and it can be considered the most modern. Differently from GeN-Foam this solves only the equation for neutrons and no other physic is involved. Moreover the equation implemented is the transport one that gives more accurate results. Conversely it leads to a more difficult implementation due to need of discretization of space, angle, energy and time. Implementation is in 2D as well as 3D and different geometries of reactors can be studied. The solver considers both steady state and transient looking for k effective solution and also the flux for different groups. Discretization implemented for space is the finite volume method while for angles it is S-n method or control angle discrete ordinate method (CADOM) and backward Euler method for time. For some studied cases, results are compared to reference solution showing a very good agreement for some kind of data. It is remarkable that the transport equation for neutrons is very difficult to implement in code and most of the time some assumptions are required to be done. The versatility and complete access of OpenFOAM® allowed Ma and his mates to produce this code, implementing the full transport equation.

Using OpenFOAM® Cosgrove and Shwageraus created MoCha-Foam [9] that is solver that implements the method of characteristic (MOC). The solver created works only on neutron equation so no multiphysic is available but equally to other solvers showed above is able to solve multi-group energy cases. Since the equation is independent from geometry it can be applied to different types of reactors as it is demonstrated in the paper. The tool relies on method of characteristics and division in energy groups; scattering and fission are considered isotropic allowing the implementation of the analytical solution (however

there is the possibility to drop this). Then since the solution is for angular flux there is integration on direction as well as on the characteristic length. It is showed that the results obtained are very similar to the ones obtained by other numerical tools, in particular for k effective.

It is remarkable the presence of other work that validates GeN-FOAM tool [10]. In 2016, Fiorina et al. published a paper on the development and verification of diffusion part of GeN-Foam. Firstly he indicates the powerful of the already present solver (it dates back to 2014): the possibility to implement the albedo boundary conditions and so also extrapolated ones; the versatility of cross sections that can be the standard ones but that can be modified by any perturbation (power, temperature, mechanical movement, etc.); possibility of parallel computing to speed up calculations (this is already present in OpenFOAM®). To this Fiorina adds new features like computations speeding up with the insertion of three different techniques. Indeed the equations for different energy flux can not be solved all together but in a segregated way. Then he even implements discontinuity factors also for the cross sections, since in code they were already implemented for the flux (discontinuity factors are parameters able to solve the problems related to homogenization that occurs). There is an entire work of PhD thesis by Jeffrey Stewart [11] that enlarges the actual solver to work with the graphite moderator: he added to normal displacement due to thermal field the expansion due to the neutron flux. It is remarkable the work by Cervi et al. in 2019 [12] that to the normal solver already present in GeN-Foam includes the possibility to use the SP3 approximation model. This implies not only the change of normal neutronic equations but also the boundary conditions implemented.

Some work thesis are developed for usage of OpenFOAM® in nuclear reactor physics. It is remarkable that Clifford [13] made a MsC thesis in 2007 on the creation of solver, called diffusionSolver, for solution of multi-group diffusion equation. He is a co-author of GeN-Foam and his name appears also in Aufiero work as well as a member of nuclear technical committee, like Aufiero and Fiorina. The solver considers non steady-state case as well as concentration of precursors and the results obtained agree well with other compared solutions however time-dependent problem can be solved only in case of one group. His work, that is a master thesis, is pioneering in the usage of OpenFOAM® since it is dated back to 2007.

It is reported also the thesis of Di Lecce [14] that focuses, with usage of OpenFOAM®, on emergency case in which in a MSR reactor the salt slowly drains. Since the MSR reactor is based on salt solution there must be a coupling between fluid and neutronic equations: moreover due to the condition studied, also increasing presence of air has to be considered. This is the reason

for which the solver for the fluid motion is a two-phase solver and strongly coupled to this it is the equation for neutrons since the quite realistic assumption that only in the salt there is the presence of these. Concerning the neutronic equations these are monoenergetic diffusive ones, for prompt neutrons and precursors concentration, with cross sections and diffusive coefficients depending on temperature. Since it is studied an accident transient the main focus is on reactivity that depends on neutrons that rely on motion of fluid and its characteristics. So to take into account this, the solver is able for each time instant to compute the reactivity by the power method.

Finally it is reported that even also application for other type of studies are developed for nuclear field, as OFFBEAT. This solver is dedicated to the study of thermal and mechanic behaviour of a nuclear pin inside a reactor. Moreover as it is written, OpenFOAM® is born for CFD evaluation: there are several applications that use it for the study of coolant inside a reactor.

The works cited up to now show the big versatility and efficiency of OpenFOAM®, ensuring that also the application that in this work thesis is desired to describe can be done.

1.3 Objective of thesis

The aim of the thesis is to study the Enhanced quasi-Static method retracing the work already made by Dulla and Nervo [15]. So the thesis proceeds with the description of the method, its implementation and finally the discussion of the results. It will be showed that the peculiarity of equations and their complexity can be only tackled by the versatility of OpenFOAM®.

Chapter 2

Enhanced quasi-static method

2.1 Introduction to the time problems in the reactor

The transport equation for neutrons fully describe their behaviour but it is not resolvable as it is. Most of the time to reach an analytical solution some assumptions are done like the steady state flux. The steady state is very important to study in order to characterize the reactor during its normal operation. Due to its characteristics, it is not available to make rapid change like other energy systems and long periods of activity are associated. Despite this, it is vital to understand the reactor behaviour in time especially when some perturbations occur. This is crucial for example in case of an accident and the time involved can be very small, as it is showed in the past events (Chernobyl). Moreover it is important to state, even if it is obvious, that the core is the first and main part of the plant that must be observed. Indeed the source of all the heat produced in this site and problems observed in other parts of the plant can reverse on the integrity of core (Three mile Island). So even time problems require a furthered research since they can predict power trend and so the necessary counter measures in order to avoid accidents.

An easy method to predict the power trend inside the reactor is the point kinetic method (PK). In experiments it was noted that the time changes of neutrons are different: the shape of the flux changes slower respect to the power associated. This allows to factorize the flux in two different quantities the shape of the flux and the power; the first depends on space while the second on time. Doing this it is assumed that the shape of flux can be kept constant while the power not. So this method assumes that the reactor can be studied as zero dimensional system and only power and concentration of delayed neutrons are the studied quantities. This approximation is very easy to implement since the final formula is a system of ordinary differential equation but it may fail a lot in the prediction of results. The basic assumption indeed can be quite wrong.

So in sixties Henry designed the quasi-static model in order to better study

the trend of power. As it is written above it can be assumed that the flux is factorized by two variables. The main difference with point kinetic is the dependence of the shape of the flux on space and also on time (however in a period larger than the power) and the studied equation is the transport one. Such formulation is more accurate since the flux can change on space and time and it is no more considered fixed. On the other side it is quite complex to implement and it leads to a very slow computation. Basically the method computes the amplitude in many time steps, quite small. After a certain time, after many little time steps, the flux is updated considering the new amplitude value.

These two methods are the most famous for the time computation of reactor. The first one is fast and easy to implement but it can lead to coarse results; the second one is accurate but it requests more computations.

2.2 Enhanced quasi static method

The existence of two different and possibly opposite methods encourages the research on new ones that are easy to implement, fast in computation and accurate in prediction.

In this way the Enhanced quasi static method (EQS) tries to satisfies the aims just written. This method was firstly designed by Ravetto et al. [16] in 1996. As the two previous method is based on a physical fact that is noted inside reactors during the transients. Most of the time the power changes inside reactor faster than the shape of neutrons but it makes mainly on one direction. A typical example of this is during loss of coolant accident (LOCA). In the core there is presence of water, for the cooling of the pins, and above there is steam due to vaporization of fluid. It is known that water is a good moderator for neutrons, slowing them and making them suitable for the fission; conversely steam is not good, it can be compared to air. The result is that profile of power on the axial side has the typical shape of cosine until the level of water: here it drops dramatically since neutrons are too fast to make fission and to produce power (this must not lead to think that in case of LOCA the core is automatically safe because there is still the residual heating by the fission product and no water to cool). The main characteristic of the method is based on this: split the total flux as function of the shape and then on amplitude that relies on time and a particular value, important for the study of the event.

The PK and the QS methods consider power as only function of time so they do not consider the shape of the power. So here it is the first important advantage: it is possible to understand how it is distributed the power generation. These informations can be quite important during the LOCA accident also to

understand how to distribute the emergency fluid. In the work by Ravetto et al. [16], some tests case are done showing important results that gives more motivation for the usage of the method. The EQS method offers very good results using bigger time step for the recalculation of the shape of flux and the number of floating points operations involved are less respect to QS. So it is remarkable the importance of the method that not only allows a better physical vision of what is occurring inside the reactor but it is also more convenient computationally speaking. In this sense it is a very good trade-off of the two previous showed methods.

In 2013 this method was retrieved by Dulla and Nervo [15]. They amply the previous knowledge offering a general mathematical formulation and studying some test cases observing the difference between the methods. In particular following the paper, the mathematical passages that take to the final formulation of the method are reported and they will be also discussed.

2.3 Mathematical formulation

Setting the equations as general as possible they become:

$$\frac{1}{v}\frac{\partial\phi(\mathbf{x},t)}{\partial t} = \mathcal{L}\phi(\mathbf{x},t) + S_p\phi(\mathbf{x},t) + \sum_i \lambda_i C_i(\mathbf{x},t) \tag{2.1}$$

$$\frac{\partial C_i(\mathbf{x},t)}{\partial t} = -\lambda_i C_i(\mathbf{x},t) + S_{d,i}\phi_i(\mathbf{x},t) \qquad (i = 1, ..., G) \tag{2.2}$$

Where ϕ and C_i are the flux and concentration of precursors of group i, $\mathbf{x}$ is the phase space, $\mathcal{L}$, S_p, S_d are operators that depend on the model chosen (diffusion or transport) to study neutrons.

As it is already written, the amplitude changes only trough a preferential direction lowering the computational burden related to kinetic theory. The factorization becomes:

$$\phi(\mathbf{x},t) = \varphi(\mathbf{x};t)A(\mathbf{x_s},t) \tag{2.3}$$

where $\mathbf{x_s}$ is the privileged quantity. Before it was referred to space but it can be one of the phase space so position, energy or direction but this last one is not so important. In space it is:

$$\phi(\vec{r},E,\vec{\Omega},t) = \varphi(\vec{r},E,\vec{\Omega},t)A(y,t) \tag{2.4}$$

While for energy is:

$$\phi(\vec{r}, E, \vec{\Omega}, t) = \varphi_g(\vec{r}, \vec{\Omega}, t) A_g(t) \tag{2.5}$$

For a better discretization, the amplitude can be discretized on space and simultaneously on energy:

$$\phi(\vec{r}, E, \vec{\Omega}, t) = \varphi_g(\vec{r}, \vec{\Omega}, t) A_g(y, t) \tag{2.6}$$

Finally it is possible also to consider more groups but one amplitude:

$$\phi(\vec{r}, E, \vec{\Omega}, t) = \varphi_g(\vec{r}, \vec{\Omega}, t) A(y, t) \tag{2.7}$$

Consequently also the equations change in their form.

After factorization there is the projection on weight function w integrating in all phase space except the variable of interest. Since the different expressions that can be achieved, here implementations are illustrated separately. Moreover for the sake of simplicity the equations studied are the one of diffusion without the presence of delayed that will be added later.

2.3.1 Factorization in space for one group

Considering the diffusion equation for one group:

$$\frac{1}{v}\frac{\partial \phi(\vec{r}, t)}{\partial t} = \nabla \cdot \Big(D(\vec{r}, t) \nabla \phi(\vec{r}, t) \Big) - \Sigma_a(\vec{r}, t)\phi(\vec{r}, t) + \nu\Sigma_f(\vec{r}, t)\phi(\vec{r}, t) + S(\vec{r}, t) \tag{2.8}$$

it is applied a simplified formula of the factorization in space as in 2.4, remembering that one update of shape flux returns already good results [16]:

$$\phi(\vec{r}, t) \approx \varphi_0(\vec{r}) A(y, t) \tag{2.9}$$

where $\varphi_0(\vec{r})$ is the flux shape of the reference solution, $\vec{r}$ is the position in space and y the space variable over which it is assumed amplitude changes. Formula 2.9 is inserted in 2.8 to obtain:

$$\begin{aligned} \frac{1}{v}\frac{\partial (\varphi_0(\vec{r}) A(y, t))}{\partial t} &= \nabla \cdot \Big[D(\vec{r}, t) \nabla \Big(\varphi_0(\vec{r}) A(y, t) \Big) \Big] - \Sigma_a(\vec{r}, t)\varphi_0(\vec{r}) A(y, t) \\ &+ \nu\Sigma_f(\vec{r}, t)\varphi_0(\vec{r}) A(y, t) + S(\vec{r}, t) \end{aligned} \tag{2.10}$$

This has to be developed in order to get the formulation for the amplitude:

$$\frac{1}{v}\varphi_0(\vec{r})\frac{\partial A(y,t)}{\partial t} = \nabla\cdot\left[D(\vec{r},t)\left(A(y,t)\nabla\varphi_0(\vec{r}) + \varphi_0(\vec{r})\frac{\partial A(y,t)}{\partial y}\right)\right]$$
$$-\Sigma_a(\vec{r},t)\varphi_0(\vec{r})A(y,t) + \nu\Sigma_f(\vec{r},t)\varphi_0(\vec{r})A(y,t) + S(\vec{r},t)$$

$$\frac{1}{v}\varphi_0(\vec{r})\frac{\partial A(y,t)}{\partial t} = D(\vec{r},t)\varphi_0(\vec{r})\frac{\partial^2 A(y,t)}{\partial y^2} + \left[D(\vec{r},t)\frac{\partial\varphi_0(\vec{r})}{\partial y} + \frac{\partial(D(\vec{r},t)\varphi_0(\vec{r}))}{\partial y}\right]\frac{\partial A(y,t)}{\partial y}$$
$$+ A(y,t)\nabla\cdot\Big(D(\vec{r},t)\nabla\varphi_0(\vec{r})\Big) - \Sigma_a(\vec{r},t)\varphi_0(\vec{r})A(y,t) + \nu\Sigma_f(\vec{r},t)\varphi_0(\vec{r})A(y,t) + S(\vec{r},t)$$

There are still not the source of transients: perturbations of source and properties are introduced: $D(\vec{r},t) \rightarrow D(\vec{r}) + \delta D(\vec{r},t)$, $S(\vec{r},t) \rightarrow S(\vec{r}) + \delta S(\vec{r},t)$, $\Sigma_a(\vec{r},t) \rightarrow \Sigma_a(\vec{r}) + \delta\Sigma_a(\vec{r},t)$ and $\Sigma_f(\vec{r},t) \rightarrow \Sigma_f(\vec{r}) + \delta\Sigma_f(\vec{r},t)$. By this the equation is:

$$\frac{1}{v}\varphi_0(\vec{r})\frac{\partial A(y,t)}{\partial t} = \Big(D(\vec{r}) + \delta D(\vec{r},t)\Big)\varphi_0(\vec{r})\frac{\partial^2 A(y,t)}{\partial y^2}$$
$$+\left[\Big(D(\vec{r}) + \delta D(\vec{r},t)\Big)\frac{\partial\varphi_0(\vec{r})}{\partial y} + \frac{\partial\Big((D(\vec{r}) + \delta D(\vec{r},t))\varphi_0(\vec{r})\Big)}{\partial y}\right]\frac{\partial A(y,t)}{\partial y}$$
$$+ A(y,t)\nabla\cdot\Big((D(\vec{r}) + \delta D(\vec{r},t))\nabla\varphi_0(\vec{r})\Big) - \Big(\Sigma_a(\vec{r}) + \delta\Sigma_a(\vec{r},t)\Big)\varphi_0(\vec{r})A(y,t)$$
$$+\Big(\nu\Sigma_f(\vec{r}) + \delta\nu\Sigma_f(\vec{r},t)\Big)\varphi_0(\vec{r})A(y,t) + S(\vec{r}) + \delta S(\vec{r},t)$$

Exploiting the reference equation, so the one valid at the initial instant, in case of steady state, there can be simplification.

$$0 = \nabla\cdot\Big(D(\vec{r})\nabla\varphi_0(\vec{r})\Big) - \Sigma_a(\vec{r})\varphi_0(\vec{r}) + \nu\Sigma_f(\vec{r})\varphi_0(\vec{r}) + S(\vec{r}) \qquad (2.11)$$

By this it can be noted that some terms that multiplies $A(y,t)$ disappear, since their sum is equal to zero. In this way the final equation becomes:

$$\begin{aligned}
&\frac{1}{v}\varphi_0(\vec{r})\frac{\partial A(y,t)}{\partial t} = \Big(D(\vec{r}) + \delta D(\vec{r},t)\Big)\varphi_0(\vec{r})\frac{\partial^2 A(y,t)}{\partial^2 y} \\
&+ \left[\Big(D(\vec{r}) + \delta D(\vec{r},t)\Big)\frac{\partial \varphi_0(\vec{r})}{\partial y} + \frac{\partial\Big((D(\vec{r}) + \delta D(\vec{r},t))\varphi_0(\vec{r})\Big)}{\partial y}\right]\frac{\partial A(y,t)}{\partial y} \\
&+ \Big[\nabla\cdot\delta D(\vec{r},t)\nabla\varphi_0(\vec{r}) - S(\vec{r}) + \Big(\delta\nu\Sigma_f(\vec{r},t) - \delta\Sigma_a(\vec{r},t)\Big)\varphi_0(\vec{r})\Big]A(y,t) \\
&+ S(\vec{r}) + \delta S(\vec{r},t)
\end{aligned} \tag{2.12}$$

Then it is performed a projection on weights, integrating in all except on the variable of interest. The choose of weight is free but it is recommended the use of importance [15] [16].

$$\begin{aligned}
&\Big\langle w(\vec{r})\Big|\frac{1}{v}\varphi_0(\vec{r})\Big\rangle_{\vec{r}-y}\frac{\partial A(y,t)}{\partial t} = \Big\langle w(\vec{r})\Big|\Big(D(\vec{r}) + \delta D(\vec{r},t)\Big)\varphi_0(\vec{r})\Big\rangle_{\vec{r}-y}\frac{\partial^2 A(y,t)}{\partial y^2} \\
&\Big\langle w(\vec{r})\Big|\Big(D(\vec{r}) + \delta D(\vec{r},t)\Big)\frac{\partial \varphi_0(\vec{r})}{\partial y} + \frac{\partial\Big((D(\vec{r}) + \delta D(\vec{r},t))\varphi_0(\vec{r})\Big)}{\partial y}\Big\rangle_{\vec{r}-y}\frac{\partial A(y,t)}{\partial y} \\
&+ \Big\langle w(\vec{r})\Big|\nabla\cdot\delta D(\vec{r},t)\nabla\varphi_0(\vec{r}) - S(\vec{r}) + \Big(\delta\nu\Sigma_f(\vec{r},t) - \delta\Sigma_a(\vec{r},t)\Big)\varphi_0(\vec{r})\Big\rangle_{\vec{r}-y} \\
&+ \langle w(\vec{r})|S(\vec{r}) + \delta S(\vec{r},t)\rangle_{\vec{r}-y}
\end{aligned} \tag{2.13}$$

In order to simplify the expression the above equation can be rewritten with greek letters, result of projections:

$$\alpha(y,t)\frac{\partial A(y,t)}{\partial t} = \delta(y,t)\frac{\partial^2 A(y,t)}{\partial y^2} + \gamma(y,t)\frac{\partial A(y,t)}{\partial y} + \eta(y,t)A(y,t) + Q(y,t) \tag{2.14}$$

It must be remarked that the amplitude equation is very particular for the nuclear field. Even if Laplacian can be considered familiar for diffusion equation, this is not for the first derivative term. This has effects on the possibility to use one or another tool and with this formulation no one dedicated to nuclear diffusion can be exploited since they do not consider this term. However it seems to retrace the classic scalar-transport equation, quite familiar to other

numerical tools like the CFD ones so also OpenFOAM® [17]. Despite this it is remarkable that the transport equation is the product of physical passages: the Laplacian term is due to the motion of the dependent variable by the gradients while the convection term is due to its motion on time (velocity). Finally it is remarkable that in $\eta(y,t)$ it appears the source term and an increase of it leads to a decreasing of the amplitude. Certainly this will not due to the presence of $Q(y,t)$.

2.3.2 Factorization for energy

For energy similar passages are performed. For a simpler version the factorization occurs only for energy groups and it is:

$$\phi(\vec{r},E,t) \approx \varphi_{g,0}(\vec{r})A_g(t) \tag{2.15}$$

that is inserted in multigroup diffusion equation that is:

$$\frac{1}{v_g}\frac{\partial\phi_g(\vec{r},t)}{\partial t} = \nabla\cdot\Big(D_g(\vec{r},t)\nabla\phi_g(\vec{r},t)\Big) - \Sigma_{r,g}(\vec{r},t)\phi_g(\vec{r},t) + \sum_{g'\neq g}\Sigma_{g'\to g}(\vec{r},t)\phi_{g'}(\vec{r},t)$$

$$+\sum_{k=1}^{G}\chi_k\nu\Sigma_{f,k}(\vec{r},t)\phi_k(\vec{r},t) + S_g(\vec{r},t) \qquad g=1,...,G \tag{2.16}$$

where G is the number of groups. The insertion of factorization produces (Even if it is not reported below it is considered that $g=1,...,G$):

$$\frac{1}{v_g}\varphi_{g,0}(\vec{r})\frac{dA_g(t)}{dt} = A_g(t)\nabla\cdot\Big(D_g(\vec{r},t)\nabla\varphi_{g,0}(\vec{r})\Big) - \Sigma_{r,g}(\vec{r},t)\varphi_{g,0}(\vec{r})(\vec{r})A_g(t)$$

$$+\sum_{g'\neq g}\Sigma_{g'\to g}(\vec{r},t)\varphi_{g',0}(\vec{r})A_{g'}(t) + \sum_{k=1}^{G}\chi_k\nu\Sigma_{f,k}(\vec{r},t)\varphi_{k,0}(\vec{r})A_k(t) + S_g(\vec{r},t) \tag{2.17}$$

As before it is introduced the perturbation on cross sections and source. These terms can be then simplified but considering the reference equation, the one in

steady state:

$$0 = \nabla \cdot \Big(D_g(\vec{r}) \nabla \varphi_{g,0}(\vec{r}) \Big) - \Sigma_{r,g}(\vec{r}) \varphi_{g,0}(\vec{r}) + \sum_{g' \neq g} \Sigma_{g' \to g}(\vec{r}) \varphi_{g'}(\vec{r})$$

$$+ \sum_{k=1}^{G} \chi_k \nu \Sigma_{f,k}(\vec{r}) \varphi_k(\vec{r}) + S_g(\vec{r}) \qquad g = 1, ..., G \tag{2.18}$$

By this equation it can be rewritten the diffusive and the absorption part as function of the two sums and the source.

$$\frac{1}{v_g} \varphi_{g,0}(\vec{r}) \frac{dA_g(t)}{dt} = \Big[\nabla \cdot \delta D_g(\vec{r}, t) \nabla \varphi_{g,0} - \sum_{k \neq 1}^{G-1} \chi_k \nu \Sigma_{f,k}(\vec{r}) \varphi_{k,0}(\vec{r})$$

$$- \sum_{g' \neq g} \Sigma_{g' \to g}(\vec{r}) \varphi_{g',0}(\vec{r}) - S_g(\vec{r}, t)$$

$$+ \Big(\chi_g \delta \nu \Sigma_{f,g}(\vec{r}, t) - \delta \Sigma_{r,g}(\vec{r}, t) \Big) \varphi_{g,0}(\vec{r}) \Big] A_g(t)$$

$$+ \sum_{g' \neq g} \Big(\Sigma_{g' \to g}(\vec{r}) + \delta \Sigma_{g' \to g}(\vec{r}, t) \Big) \varphi_{g',0}(\vec{r}) A_{g'}(t)$$

$$+ \sum_{k=1}^{G} \chi_k \Big(\nu \Sigma_{f,k}(\vec{r}) + \delta \nu \Sigma_{f,k}(\vec{r}, t) \Big) \varphi_{k,0}(\vec{r}) A_k(t) + S_{g,0}(\vec{r}) + \delta S_g(\vec{r}, t) \tag{2.19}$$

Then the equation has to be projected on a specific function $w_g(\vec{r})$ and since the factorization is on energy the integration is performed in all space:

$$\left\langle w_g(\vec{r})\middle|\frac{1}{v_g}\varphi_{g,0}(\vec{r})\right\rangle \frac{dA_g(t)}{dt} = \left\langle w_g(\vec{r})\middle|\nabla\cdot\delta D_g(\vec{r},t)\nabla\varphi_{g,0} - \sum_{k\neq 1}^{G-1}\chi_k\nu\Sigma_{f,k}(\vec{r})\varphi_{k,0}(\vec{r})\right.$$
$$- \sum_{g'\neq g}\Sigma_{g'\to g}(\vec{r})\varphi_{g',0}(\vec{r}) - S_g(\vec{r})$$
$$\left.+\Big(\chi_g\delta\nu\Sigma_{f,g}(\vec{r},t) - \delta\Sigma_{r,g}(\vec{r},t)\Big)\varphi_{g,0}(\vec{r})\right\rangle A_g(t)$$
$$+ \sum_{g'\neq g}\left\langle w_g(\vec{r})\middle|\Big(\Sigma_{g'\to g}(\vec{r}) + \delta\Sigma_{g'\to g}(\vec{r},t)\Big)\varphi_{g',0}(\vec{r})\right\rangle A_{g'}(t)$$
$$+ \sum_{k=1}^{G-1}\left\langle w_g(\vec{r})\middle|\chi_k\Big(\nu\Sigma_{f,k}(\vec{r}) + \delta\nu\Sigma_{f,k}(\vec{r},t)\Big)\varphi_{k,0}(\vec{r})\right\rangle A_k(t)$$
$$+ \langle w(\vec{r})_g|S_g(\vec{r}) + \delta S_g(\vec{r},t)\rangle \tag{2.20}$$

This can be simplified as:

$$\alpha_g(t)\frac{dA_g(t)}{dt} = \gamma_g(t)A_g(t) + \sum_{g'\neq g}\beta_{g'}(t)A_{g'}(t) + \sum_{k=1}^{G-1}\delta_k(t)A_k(t) + Q_g(t) \tag{2.21}$$

where Greek letters represent the value of the projection.

Differently from the case with factorization in space now the equation has only a differentiable term that is the one on time for the amplitude. However this is one of G equations depending on number of groups in which energy is divided. So the final expression is a system of ordinary differentiable equations. This is the only complication to a system of equation that can be easily implemented in many solvers. It is also remarkable that the weights are only dependent on time, differently from the other ones, and this further simplify the implementation. In their definition no differentiable terms appear that request a deeper computational effort. It is finally remarkable how the fission or scattering by other groups have a direct effect on the amplitude of the studied group.

2.3.3 Factorization for space-energy

It is possible a third factorization that is the one in space as well as in energy. The passages to achieve the final equations are very similar to the ones above and the final form is in some ways equivalent to the one of space. The

factorization is:

$$\phi_{g,0}(\vec{r},t) \approx \varphi_{g,0}(\vec{r})A_g(y,t) \tag{2.22}$$

It is introduced in equation 2.16 giving as final result:

$$\frac{1}{v_g}\frac{\partial(\varphi_{g,0}(\vec{r})A_g(y,t))}{\partial t} = \nabla\cdot\Big[D_g(\vec{r},t)\nabla\Big(\varphi_{g,0}(\vec{r})A_g(y,t)\Big)\Big] - \Sigma_{r,g}(\vec{r},t)\varphi_{g,0}(\vec{r})A_g(y,t)$$

$$+\sum_{g'\neq g}\Sigma_{g'\to g}(\vec{r},t)\varphi_{g',0}(\vec{r})A_{g'}(y,t) + \sum_{k=1}^{G}\chi_k\nu\Sigma_{f,k}(\vec{r},t)\varphi_{k,0}(\vec{r})A_k(y,t) + S(\vec{r},t) \tag{2.23}$$

For sake of simplicity the passages that lead to the differentiable terms in space are not reported since they are very similar to the ones already done in space. Again the reference equation 2.16 is used in order to substitute the Laplacian as it is done in section 2.3.2, finding similar weights as in energy discretization.

The final equation is:

$$\left\langle w_g(\vec{r})\middle|\frac{1}{v_g}\varphi_{g,0}(\vec{r})\right\rangle_{\vec{r}-y}\frac{\partial A_g(y,t))}{\partial t}=\left\langle w_g(\vec{r})\middle|\Big(D_g(\vec{r})+\delta D_g(\vec{r},t)\Big)\varphi_{g,0}(\vec{r})\right\rangle_{\vec{r}-y}\frac{\partial^2 A_g(y,t)}{\partial y^2}$$

$$+\left\langle w_g(\vec{r})\middle|\left[\Big(D_g(\vec{r})+\delta D_g(\vec{r},t)\Big)\frac{\partial\varphi_{g,0}(\vec{r})}{\partial y}\right.\right.$$

$$\left.\left.+\frac{\partial\Big(\big(D_g(\vec{r})+\delta D_g(\vec{r},t)\Big)\varphi_{g,0}(\vec{r})\big)}{\partial y}\right]\right\rangle_{\vec{r}-y}\frac{\partial A_g(y,t)}{\partial y}$$

$$\left\langle w_g(\vec{r})\middle|\left[\nabla\cdot\delta D_g(\vec{r},t)\nabla\varphi_{g,0}-\sum_{k\neq g}^{G-1}\chi_k\nu\Sigma_{f,k}(\vec{r})\varphi_{k,0}(\vec{r})\right.\right.$$

$$-\sum_{g'\neq g}\Sigma_{g'\to g}(\vec{r})\varphi_{g',0}(\vec{r})-S_g(\vec{r},t)$$

$$\left.\left.+\Big(\chi_g\delta\nu\Sigma_{f,g}(\vec{r},t)-\delta\Sigma_{r,g}(\vec{r},t)\Big)\varphi_{g,0}(\vec{r})\right]\right\rangle_{\vec{r}-y}A_g(y,t)$$

$$+\left\langle w_g(\vec{r})\middle|\sum_{g'\neq g}\Big(\Sigma_{g'\to g}(\vec{r})+\delta\Sigma_{g'\to g}(\vec{r},t)\Big)\varphi_{g',0}(\vec{r})\right\rangle_{\vec{r}-y}A_{g'}(y,t)$$

$$+\left\langle w_g(\vec{r})\middle|\sum_{k\neq g}^{G-1}\chi_g\Big(\nu\Sigma_{f,k}(\vec{r})+\delta\nu\Sigma_{f,k}(\vec{r},t)\Big)\varphi_{k,0}(\vec{r})\right\rangle_{\vec{r}-y}A_k(y,t)$$

$$+\left\langle w_g(\vec{r})\middle|S_g(\vec{r})+\delta S_g(\vec{r},t)\right\rangle_{\vec{r}-y}\tag{2.24}$$

After the projection on the specific weight, this can be rewritten by greek letters as:

$$\alpha_g(y,t)\frac{\partial A_g(y,t)}{\partial t}=\delta_g(y,t)\frac{\partial^2 A_g(y,t)}{\partial y^2}+\gamma_g(y,t)\frac{\partial A_g(y,t)}{\partial y}$$

$$+\eta_g(y,t)A_g(y,t)+\sum_{g'\neq g}\beta_{g'}(y,t)A_{g'}(y,t)+\sum_{k\neq g}^{G-1}\kappa_k(y,t)A_k(y,t)+Q_g(y,t)\tag{2.25}$$

As it can be observed the result is a reasonable union between the space and energy discretization. The same applications problem regarding the space can be extended also to this version with the adjoint of number of equation at least

as the number of groups, coupled between them. This widens the difference with the scalar-transport equation. It is remarkable that in $\eta_g(y,t)$ new terms related to other groups appear.

2.3.4 Factorization for space from more than one group

In the last two sections it is showed the discretization in energy groups and space. Here it is showed that it is possible to discretize only on space starting from a number G of equations. The discretization indeed wants only one amplitude:

$$\phi_g(\vec{r},t) \approx \varphi_{g,0}(\vec{r})A(y,t) \tag{2.26}$$

This one has to be inserted in the equation 2.16, leading to this:

$$\frac{1}{v_g}\frac{\partial(\varphi_{g,0}(\vec{r})A(y,t))}{\partial t} = \nabla\cdot\Big[D_g(\vec{r},t)\nabla\Big(\varphi_{g,0}(\vec{r})A(y,t)\Big)\Big] - \Sigma_{r,g}(\vec{r},t)\varphi_{g,0}(\vec{r})A(y,t)$$
$$+\sum_{g'\neq g}\Sigma_{g'\to g}(\vec{r},t)\varphi_{g',0}(\vec{r})A(y,t) + \sum_{k=1}^{G-1}\chi_k\nu\Sigma_{f,k}(\vec{r},t)\varphi_{k,0}(\vec{r})A(y,t) + S(\vec{r},t) \tag{2.27}$$

However this is just one equation of system of G linear equations. The introduction of perturbations and the projection on importance does not undermine this property. So for a simpler treatment the equations are summed. In this way there is only one equation to solve where the contribute of each groups is

considered.

$$
\begin{aligned}
&\frac{\partial A(y,t)}{\partial t}\sum_{g=1}^{G}\left\langle w_g(\vec{r})\middle|\frac{1}{v_g}\varphi_{g,0}(\vec{r})\right\rangle_{\vec{r}-y}\\
&=\frac{\partial^2 A(y,t)}{\partial y^2}\sum_{g=1}^{G}\left\langle w_g(\vec{r})\middle|\Big(D_g(\vec{r})+\delta D_g(\vec{r},t)\Big)\varphi_{g,0}(\vec{r})\right\rangle_{\vec{r}-y}\\
&+\frac{\partial A(y,t)}{\partial y}\sum_{g=1}^{G}\left\langle w_g(\vec{r})\middle|\Big(D_g(\vec{r})+\delta D_g(\vec{r},t)\Big)\frac{\partial\varphi_{g,0}(\vec{r})}{\partial y}\right.\\
&\left.+\frac{\partial\Big((D_g(\vec{r})+\delta D_g(\vec{r},t))\varphi_{g,0}(\vec{r})\Big)}{\partial y}\right\rangle_{\vec{r}-y}\\
&A(y,t)\sum_{g=1}^{G}\left\langle w_g(\vec{r})\middle|\nabla\delta D_g(\vec{r},t)\nabla\varphi_{g,0}(\vec{r})-\delta\Sigma_{r,g}(\vec{r},t)\varphi_{g,0}(\vec{r})\right.\\
&\left.+\sum_{k=1}^{G}\chi_g\delta\nu\Sigma_{f,k}(\vec{r},t)\varphi_{k,0}(\vec{r})+\sum_{g'\neq g}\delta\Sigma_{g'\to g}(\vec{r},t)\varphi_{g',0}(\vec{r})-S_g(\vec{r})\right\rangle_{\vec{r}-y}\\
&\sum_{g=1}^{G}\left\langle w_g(\vec{r})\middle|S_g(\vec{r})+\delta S_g(\vec{r},t)\right\rangle_{\vec{r}-y}
\end{aligned}
\tag{2.28}
$$

As it can be seen by the coefficients many of the terms that refer to the reference equation disappear. The final version is equivalent to 2.14.

2.4 Boundary and initial conditions

In this section the boundary and the initial conditions will be showed. It is considered as reference case the one of space since it can be seen that the space discretization is the g equation of the system.

2.4.1 Boundary conditions

To get these, as it is done by Dulla and Nervo, it is important to recall that the method is created by a factorization of the flux. So in the typical boundary conditions for the flux it is inserted the desired factorization. General boundary conditions are the extrapolated ones:

$$
\phi(\partial\vec{r},t)+d\frac{\partial\phi(\vec{r},t)}{\partial\vec{n}}\bigg|_{\partial\vec{r}}=0 \tag{2.29}
$$

where $\partial\vec{r}$ is a point of boundary. In the above equation the factorization 2.4 is inserted giving consequently the boundary condition for the amplitude:

$$\varphi_0(y=0)A(y=0,t) - d\frac{\partial\varphi_0(\vec{r})}{\partial y}\bigg|_{y=0} A(y=0,t) - \varphi_0(y=0)d\frac{\partial A}{\partial y}\bigg|_{y=0} = 0$$

$$\varphi_0(y=H)A(y=H,t) + d\frac{\partial\varphi_0}{\partial y}\bigg|_{y=H} A(y=H,t) + \varphi_0(y=H)d\frac{\partial A}{\partial y}\bigg|_{y=H} = 0$$

y and H are the edges of the system in y direction. It can be observed that the first two terms of both equations satisfy the condition imposed by the boundary condition 2.29 so they cancel themselves and since $\varphi_0(\partial\vec{r},t)$ should be diverse from 0 for non trivial results:

$$\frac{\partial A}{\partial y}\bigg|_{y=0,H} = 0 \tag{2.30}$$

where subscripts represent the edges of the studied system.

2.4.2 Initial condition

The initial condition is quite straightforward. If it is considered that before the transient the reactor is in steady state, releasing the reference power, the amplitude is:

$$A(y,t=0) = 1 \tag{2.31}$$

This can be also deduced by the reference equation.

2.5 Insertion of delayed

Actually the implementation presented does not contain the delayed neutrons, neither in the work made by Dulla and Nervo. Maybe it is present in the work by Ravetto but a complete and concrete mathematical formulation not. Consider them should lead to more physical results, even if it creates more problems from computational point of view since everywhere another equation appears.

2.5.1 Factorization for space for one group

Exploiting again the diffusion version, there is a system (R families of delayed):

$$\begin{cases} \frac{1}{v}\frac{\partial\phi(\vec{r},t)}{\partial t} = \nabla\cdot\Big(D(\vec{r},t)\nabla\phi(\vec{r},t)\Big) - \Sigma_a(\vec{r},t)\phi(\vec{r},t) \\ \qquad\qquad + (1-\beta)\nu\Sigma_f(\vec{r},t)\phi(\vec{r},t) + \sum_{i=0}^{R}\lambda_i C_i(\vec{r},t) + S(\vec{r},t) \end{cases} \tag{2.32}$$

$$\frac{\partial C_i(\vec{r},t)}{\partial t} = -\lambda_i C_i(\vec{r},t) + \beta\nu\Sigma_f(\vec{r},t)\phi(\vec{r},t) \qquad i = 1,...,R \tag{2.33}$$

As before the part $\phi(\vec{r},t)$ is substituted by the product between the reference solution flux and the amplitude, so using equation 2.9. Again the equations are developed (passages are not shown because they are quite similar to what is written above) and they are projected on adjoint:

$$\begin{cases} \Big\langle w(\vec{r})\Big|\frac{1}{v}\varphi_0(\vec{r})\Big\rangle_{\vec{r}-y}\frac{\partial A(y,t)}{\partial t} = \Big\langle w(\vec{r})\Big|\Big(D(\vec{r}) + \delta D(\vec{r},t)\Big)\varphi_0(\vec{r})\Big\rangle_{\vec{r}-y}\frac{\partial^2 A(y,t)}{\partial y^2} \\ \qquad + \Big\langle w(\vec{r})\Big|\Big[\Big(D(\vec{r}) + \delta D(\vec{r},t)\Big)\frac{\partial\varphi_0(\vec{r})}{\partial y} \\ \qquad + \frac{\partial\Big((D(\vec{r}) + \delta D(\vec{r},t))\varphi_0(\vec{r})\Big)}{\partial y}\Big]\Big\rangle_{\vec{r}-y}\frac{\partial A(y,t)}{\partial y} \\ \qquad + \Big\langle w(\vec{r})\Big|\nabla\cdot\delta D(\vec{r},t)\nabla\varphi_0(\vec{r}) - S(\vec{r},t) \\ \qquad + \Big(\delta\nu\Sigma_f(\vec{r},t) - \delta\Sigma_a(\vec{r},t)\Big)\varphi_0(\vec{r})\Big\rangle_{\vec{r}-y}A(y,t) \\ \qquad + \sum_{i=1}^{R}\lambda_i\left\langle w(\vec{r})|C_i(\vec{r},t)\right\rangle_{\vec{r}-y} + \left\langle w(\vec{r})|S(\vec{r}) + \delta S(\vec{r},t)\right\rangle_{\vec{r}-y} \end{cases} \tag{2.34}$$

$$\begin{cases} \Big\langle w(\vec{r})\Big|\frac{\partial C_i(\vec{r},t)}{\partial t}\Big\rangle_{\vec{r}-y} = -\lambda_i\Big\langle w(\vec{r})\Big|C_i(\vec{r},t)\Big\rangle_{\vec{r}-y} \\ \qquad + \sum_{g=1}^{G}\Big\langle w(\vec{r})\Big|\beta\chi_g(\nu\Sigma_f(\vec{r},t) + \delta\nu\Sigma_f(\vec{r},t))\varphi_0(\vec{r})\Big\rangle_{\vec{r}-y}A(y,t) \end{cases} \tag{2.35}$$

That can be easily rewritten as:

$$\begin{cases} \alpha(y,t)\dfrac{\partial A(y,t)}{\partial t} = \delta(y,t)\dfrac{\partial^2 A(y,t)}{\partial y^2} + \gamma(y,t)\dfrac{\partial A(y,t)}{\partial y} + \eta(y,t)A(y,t) \\ \qquad\qquad + \sum_{i=1}^{R} \lambda_i g_i(y,t) + Q(y,t) \end{cases} \tag{2.36}$$

$$\frac{\partial g_i(y,t)}{\partial t} = -\lambda_i g_i(y,t) + \zeta_i(y,t)A(y,t) \tag{2.37}$$

The equation for amplitude is basically equal to 2.14 with the adjoint of the concentration of delayed neutrons and different coefficients. It can be noted that the perturbation in the fission is reflected directly also in delay concentration. On the other side the effect of delay neutrons is immediately reflected on amplitude. The terms in the brackets in front of the amplitude are diverse from equation 2.14. Firstly the perturbation of fission cross section is decreased by the term $(1-\beta)$ the one that considers that not all neutrons are prompt. Secondly still in these brackets by the last term is considered that at the begin of the transient not all neutrons can be considered. It is remarkable that the decay constant is not in the projections of the weights: if this was considered inside the coefficient it would give problems in the time derivative of delayed equations.

2.5.2 Factorization in energy

The energy factorization is similar to previous cases. The presence of delayed adds R equations to the system. For simplicity it is assumed that the delay appears only in the equation of group g. Considering the normal diffusion equa-

tions for G groups:

$$\begin{cases} \frac{1}{v}\frac{\partial\phi(\vec{r},t)}{\partial t} = \nabla\cdot\Big(D_g(\vec{r},t)\nabla\phi_g(\vec{r},t)\Big) - \Sigma_{r,g}(\vec{r},t)\phi_g(\vec{r},t) \\ \qquad\qquad + (1-\beta)\sum_{k=1}^{G}\chi_g\nu\Sigma_{f,k}(\vec{r},t)\phi_k(\vec{r},t) + \sum_{g'\neq g}^{G-1}\Sigma_{g'\to g}(\vec{r},t)\phi_{g'}(\vec{r},t) \\ \qquad\qquad + \sum_{i=1}^{R}\lambda_i C_i(\vec{r},t) + S_g(\vec{r},t) \end{cases} \tag{2.38}$$

$$\frac{\partial C_i(\vec{r},t)}{\partial t} = -\lambda_i C_i(\vec{r},t) + \beta\sum_{g=1}^{G}\nu\Sigma_{f,g}(\vec{r},t)\phi_g(\vec{r},t) \tag{2.39}$$

Inserting the factorization number 2.15 in the equation 2.38 and 2.39, then projecting on the adjoint and considering that the reference equations are 2.18 so there is not difference between including or not delayed, it becomes:

$$\begin{cases} \Big\langle w_g(\vec{r})\Big|\frac{1}{v_g}\varphi_{g,0}(\vec{r})\Big\rangle\frac{dA_g(t)}{dt} = \Big\langle w_g(\vec{r})\Big|\nabla\cdot\delta D_g(\vec{r},t)\nabla\varphi_{g,0}(\vec{r}) - \sum_{k\neq g}^{G-1}\chi_k\nu\Sigma_{f,k}(\vec{r})\varphi_{k,0}(\vec{r}) \\ \qquad\qquad - \sum_{g\neq g'}\Sigma_{g'\to g}(\vec{r})\varphi_{g',0}(\vec{r}) - S_g(\vec{r}) \qquad (2.41) \\ \qquad\qquad - \Big((1-\beta)\delta\nu\Sigma_{f,g}(\vec{r}) - \delta\Sigma_{r,g}\Big)\varphi_{g,0}(\vec{r}) - \beta\nu\Sigma_{f,g}(\vec{r})\varphi_{g,0}(\vec{r})\Big\rangle A_g(t) \\ \qquad\qquad + \sum_{k\neq g}^{G-1}\Big\langle w_g(\vec{r})\Big|\chi_k(1-\beta)(\nu\Sigma_{f,k}(\vec{r}) + \delta\chi_g\nu\Sigma_{f,k}(\vec{r},t))\varphi_{k,0}(\vec{r})\Big\rangle A_k(t) \\ \qquad\qquad + \sum_{g\neq g'}\Big\langle w_g(\vec{r})\Big|\Big(\Sigma_{g'\to g}(\vec{r}) + \delta\Sigma_{g'\to g}(\vec{r},t)\Big)\varphi_{g',0}(\vec{r})\Big\rangle A_{g'}(t) \\ \qquad\qquad + \langle w_g(\vec{r})|S_g(\vec{r}) + \delta S_g(\vec{r},t)\rangle \\ \Big\langle w_1(\vec{r})\Big|\frac{\partial C_i(\vec{r},t)}{\partial t}\Big\rangle = -\lambda_i\langle w_g(\vec{r})|C_i(\vec{r},t)\rangle \qquad (2.42) \\ \qquad\qquad + \sum_{g=1}^{G}\Big\langle\beta_i\chi_g(\nu\Sigma_{f,g}(\vec{r}) + \delta\nu\Sigma_{f,g}(\vec{r},t))\varphi_{g,0}\Big\rangle A_g(t) \end{cases}$$

This can be easily seen as:

$$\begin{cases} \alpha(t)\dfrac{dA_g(t)}{dt} = \gamma(t)A_g(t) + \sum\limits_{g' \neq g} \beta_1(t)A_{g'}(t) + \sum\limits_{k \neq g}^{G-1} \beta_2(t)A_k(t) \\ \qquad\qquad + \sum\limits_{i=1}^{R} \lambda_i g(t) + Q_g(t) \end{cases} \tag{2.42}$$

$$\frac{dg_i(t)}{dt} = -\lambda_i g_i(t) + \sum_{g=1}^{G} \zeta_{g,i}(t)A_g(t) \tag{2.43}$$

The equation does not differ so much from the previous one and same consideration done in previous section can be extended to this. The coefficient $\zeta(y,t)$ depends on the energy group as well as on the delay family.

2.5.3 Space energy discretization

As before the space-energy discretization is between the energy and space discretization. The assumption that delay appears only in the g equation is repeated. Starting from 2.38 and 2.39 but inserting the discretization 2.22 and

immediately projecting gives:

$$
\begin{cases}
\Big\langle w_g(\vec{r})\Big|\dfrac{1}{v_g}\varphi_{g,0}(\vec{r})\Big\rangle_{\vec{r}-y}\dfrac{\partial A_g(y,t)}{\partial t} = \Big\langle w_g(\vec{r})\Big|\Big(D_g(\vec{r})+\delta D_g(\vec{r},t)\Big)\varphi_{g,0}(\vec{r})\Big\rangle_{\vec{r}-y}\dfrac{\partial^2 A_g(y,t)}{\partial y^2} \\
\qquad + \Big\langle w_g(\vec{r})\Big|\Big(D_g(\vec{r})+\delta D(\vec{r},t)\Big)\dfrac{\partial\varphi_{g,0}(\vec{r})}{\partial y} \\
\qquad + \dfrac{\partial((D(\vec{r})+\delta D(\vec{r},t))\varphi_{g,0}(\vec{r}))}{\partial y}\Big\rangle_{\vec{r}-y}\dfrac{\partial A(y,t)}{\partial y} \\
\qquad + \Big\langle w_g(\vec{r})\Big|\Big[\nabla\cdot\delta D_g(\vec{r},t)\nabla\varphi_{g,0}(\vec{r}) - \sum_{k\neq g}^{G-1}\chi_k\nu\Sigma_{f,k}(\vec{r})\varphi_{k,0}(\vec{r}) \\
\qquad - \sum_{g'\neq g}^{G-1}\Sigma_{g'\to g}(\vec{r})\varphi_{g',0}(\vec{r}) - S(\vec{r}) + \Big(1-\beta\Big)\Big(\delta\nu\Sigma_{f,g}(\vec{r},t) \\
\qquad - \delta\Sigma_{r,g}(\vec{r},t)\Big)\varphi_g(\vec{r}) - \beta\nu\Sigma_{f,g}(\vec{r})\varphi_{g,0}(\vec{r})\Big]\Big\rangle_{\vec{r}-y}A_g(y,t) \\
\qquad + \sum_{k\neq g}^{G-1}\Big\langle w_g(\vec{r})\Big|\chi_k\Big(\nu\Sigma_{f,k}(\vec{r})+\delta\nu\Sigma_{f,k}(\vec{r},t)\Big)\varphi_{k,0}(\vec{r})\Big\rangle_{\vec{r}-y}A_k(y,t) \\
\qquad + \sum_{g\neq g'}\Big\langle w_g(\vec{r})\Big|\Big(\Sigma_{g'\to g}(\vec{r})+\delta\Sigma_{g'\to g}(\vec{r},t)\Big)\varphi_{g',0}(\vec{r})\Big\rangle_{\vec{r}-y}A_{g'}(y,t) \\
\qquad + \langle w_g(\vec{r})|S_g(\vec{r})+\delta S_g(\vec{r},t)\rangle_{\vec{r}-y}
\end{cases}
\tag{2.45}
$$

$$
\begin{aligned}
\Big\langle w_g(\vec{r})\Big|\frac{\partial C_i(\vec{r},t)}{\partial t}\Big\rangle_{\vec{r}-y} &= -\lambda\Big\langle w_g(\vec{r})\Big|C_i(\vec{r},t)\Big\rangle_{\vec{r}-y} \\
&\quad + \sum_{g=1}^{G}\Big\langle w_g(\vec{r})\Big|\beta_i\chi_g\Big(\nu\Sigma_{f,g}(\vec{r})+\delta\nu\Sigma_{f,g}(\vec{r},t)\Big)\varphi_{g,0}(\vec{r})\Big\rangle_{\vec{r}-y}
\end{aligned}
\tag{2.46}
$$

The equation represents a mix of the space and the energy equations with the adjoint of delayed neutrons. It can be rewritten with simply:

$$\begin{cases} \alpha_g(y,t)\dfrac{\partial A_g(y,t)}{\partial t} = \delta_g(y,t)\dfrac{\partial^2 A_g(y,t)}{\partial y^2} + \gamma_g(y,t)\dfrac{\partial A_g(y,t)}{\partial y}] + \eta_g(y,t)A_g(y,t) \\ \qquad + \displaystyle\sum_{g\neq g'} \beta_{g'}(y,t)A_{g'}(y,t) + \sum_{k=1}^{G} \beta_k(y,t)A_k(y,t) + \sum_{i=1}^{R} \lambda_i g_i(y,t) + Q_g(y,t) \end{cases} \tag{2.46}$$

$$\frac{\partial g_i(y,t)}{\partial t} = -\lambda_i g(y,t) + \sum_{g=1}^{G} \zeta_g(y,t)A_g(y,t) \tag{2.47}$$

2.5.4 Space discretization for more than one group

Finally it is showed the discretization procedure in case of more than one group but considering only one amplitude.

$$
\begin{cases}
\dfrac{\partial A(y,t)}{\partial t}\displaystyle\sum_{g=1}^{G}\left\langle w_g(\vec{r})\middle|\frac{1}{v_g}\varphi_{g,0}(\vec{r})\right\rangle_{\vec{r}-y} \\
\quad =\dfrac{\partial^2 A(y,t)}{\partial y^2}\displaystyle\sum_{g=1}^{G}\left\langle w_g(\vec{r})\middle|\Big(D_g(\vec{r})+\delta D_g(\vec{r},t)\Big)\varphi_{g,0}(\vec{r})\right\rangle_{\vec{r}-y} \\
\qquad +\dfrac{\partial A(y,t)}{\partial y}\displaystyle\sum_{g=1}^{G}\left\langle w_g(\vec{r})\middle|\Big(D_g(\vec{r})+\delta D_g(\vec{r},t)\Big)\frac{\partial\varphi_{g,0}(\vec{r})}{\partial y}\right. \\
\qquad \left.+\dfrac{\partial((D_g(\vec{r})+\delta D_g(\vec{r}),t)\varphi_{g,0}(\vec{r}))}{\partial y}\right\rangle_{\vec{r}-y} \qquad (2.49) \\
\qquad A(y,t)\displaystyle\sum_{g=1}^{G}\left\langle w_g(\vec{r})\middle|\nabla\delta D_g(\vec{r},t)\nabla\varphi_{g,0}(\vec{r})-\delta\Sigma_{r,g}(\vec{r},t)\varphi_{g,0}(\vec{r})\right. \\
\qquad \left.+\displaystyle\sum_{k=1}^{G}\delta\nu\Sigma_{f,k}(\vec{r},t)\varphi_{k,0}(\vec{r})+\sum_{g'\neq g}\delta\Sigma_{g'\to g}(\vec{r},t)\varphi_{g',0}(\vec{r})-S_g(\vec{r})\right\rangle_{\vec{r}-y} \\
\qquad +\displaystyle\sum_{i=1}^{R}\lambda_i\Big\langle w_g(\vec{r})\Big|C_i(\vec{r},t)\Big\rangle_{\vec{r}-y}+\sum_{g=1}^{G}\left\langle w_g(\vec{r})\middle|S_g(\vec{r})+\delta S_g(\vec{r},t)\right\rangle_{\vec{r}-y} \\
\left\langle w_g(\vec{r})\middle|\dfrac{\partial C_i(\vec{r},t)}{\partial t}\right\rangle_{\vec{r}-y}=-\lambda\Big\langle w_g(\vec{r})\Big|C_i(\vec{r},t)\Big\rangle_{\vec{r}-y} \qquad (2.50) \\
\qquad +\left\langle w_g(\vec{r})\middle|\beta_i\displaystyle\sum_{g=1}^{G}\chi_i\Big(\nu\Sigma_{f,g}(\vec{r})+\delta\nu\Sigma_{f,g}(\vec{r},t)\Big)\varphi_{g,0}(\vec{r})\right\rangle_{\vec{r}-y}A(y,t)
\end{cases}
$$

The equation is equivalent to the one for only one group but there is the contribute of all the other groups. This system is easier to solve respect to the one for the space-energy factorization.

2.5.5 Initial condition for delayed

Due to the insertion of delayed neutrons, so of a new variable, initial conditions have to be defined, since it is an equation on time. This has to be derived from the reference equation. The starting point is the one that realizes the steady state, so that the left part of equations 2.5.1 is zero. By equation two of the

just cited system it comes from a simple equation that gives:

$$C_i(\vec{r},0) = \frac{\chi\beta_i\nu\Sigma_f(\vec{r},0)\phi(\vec{r},0)}{\lambda_i} \tag{2.50}$$

Considering this the normal condition for the initial state of precursor and that the amplitude is one at $t = 0$, for variable g_i it results that:

$$g_i(\vec{r},0) = \left\langle w(\vec{r}) \middle| \frac{\chi\beta_i\nu\Sigma(\vec{r},0)\varphi(\vec{r},0)}{\lambda_i} \right\rangle_{\vec{r}-y} \tag{2.51}$$

This condition is true in every point of space, depending on it.

Chapter 3

Implementation in OpenFOAM®

As it can be observed in the previous chapters the final equations retrieved are quite strange: the space one is similar to the scalar transport equation and if there is the presence of the delayed, a system has to be treated. Considering also the particularity of the method there is not any numerical tool that has inside it, already implemented, the EQS. Moreover the structure of equations make difficult the analytical solution that maybe could be found. To get rid of all these things, the solution is to pass trough discretization procedure. Solver like Matlab can be easily used for numerical solution like this but further problems related to correctness of the solution and on the scheme adopted for discretization will go beyond the scope of the thesis. So usage of OpenFOAM® seems to be the best solution to get rid of the equations written in previous chapters. The versatility of OpenFOAM® and the complete accessibility seem to be the best accessories to solve the problems related to the EQS implementation. However there are also disadvantages: the biggest one encountered is that the open source programme does not give a complete manual user, slowing down the implementation procedure. So in order to simplify this process some assumptions are done:

- system should be a simple geometry like a cube or a parallelepiped;
- the mesh should be a structured mesh;
- there is not source;
- the energy groups are two;
- in case of delayed, it is considered one family;

Moreover it has to be considered that for a complete working code the starting data is the flux.

3.1 How OpenFOAM® works

In chapter one it is described how OpenFOAM® works generally but not specifically. Indeed since it is a union of classes and it does not have a GUI, the data have to be given in specific way. First of all, the solver has to be run inside the main folder that contains data for the simulation. These are organized in three different directories: `0`,`constant` and `system`. In the first one there are the data field that will be used for the run: in the specific case there are the fluxes, the amplitudes, the properties, the weights and some dummy data for the correct function of the tool (like the $Q(\vec{r})$ function). These files contain for each quantities the boundary and the initial conditions. In `constant` there should be the properties of the system like cross sections or diffusion properties but in successive section will be reported how and which decisions are done in order to account them. In this directory there is another one called `polymesh`: it contains all files necessary for the production of mesh. To make an example it contains `points` that is a file with all the points of the mesh organized like a matrix and ordered in increasing way from the horizontal coordinate. It will be used later for the transformation of mesh. Finally there is the `system` that maybe is the most important since there are the files that drive the simulation. Here there is the document for the creation of the mesh, `blockMeshDict`, the two where set the schemes and the algorithms for the work of the simulation, `fvSchemes` and `fvSolution`, the `setFieldsDict` that allows the set of initial value for the quantities and finally `controlDict` where are reported all the informations for the run of the simulation like the starting time, the timestep, precision of results, etc.. These are organized in folders named like the time instants (this is the reason for the `0` folder) where the updated values are stored. For the simulation, the first thing to do is the creation of mesh that is run with command `blockMesh`. To do not get error there must be a coordination between the file of mesh and the ones in `0`: they must refer to the same type of patch. Finally in order to visualize results it is necessary a post-processing tool and a file that connects the data to this: this is done by Paraview 5.9.0 and by the `.foam` extension file.

Above it was illustrated how specifically works a simulation and the ones done for the EQS method follow the similar steps. The problems are solved by four different solvers, one for the type of discretization, to have more orderly organization of the data. First of all there is the geometry evaluation needed for the further computation of coefficients. Then there is the flux evaluation by the properties reading. It follows the transformation of the mesh and since reference equations are always the same, despite the insertion of delayed, there is the possibility to account for them. The final passages are the evaluation of

the coefficients and then of amplitude. Finally there is the post-processing of the results that occurs with Matlab® since ParaView is more suited for visualization of single time instant. All these passages are summarized in the figure 3.1. It is reported that to give a more freedom to user the properties are given with `setFieldsDict`. In this way it is easier to set them as function of the space even if OpenFOAM® classify them as variables to compute. In the next sections it will be described the lines of code.

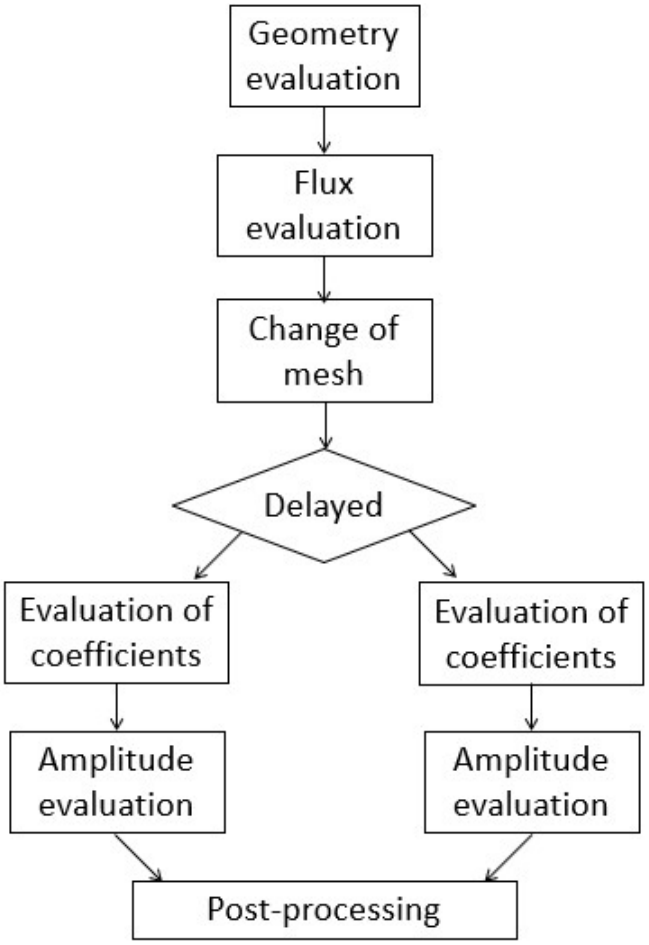


Figure 3.1: Scheme of algorithm for the simulation

3.2 Geometry computation

As it can be seen the method prescribes that the amplitude is dependent on time and another variables, leading to a 0-D (energy discretization) or 1-D (space and space-energy discretization) evaluation. On the other side, it is comprehensible that the reference flux evaluation is a dimensional evaluation and for interesting studies, quantities should be at least 2-D. It follows that the same mesh used for the reference flux can not be used, requiring a modification. Before doing this it is necessary to know the important quantities of the structure since they will be used for the computations of weights. The solver is able to evaluate:

- extremes of structure;

- sizes;

- number of cells on each direction;

- the average size of cell in each direction;

These quantities can be extracted by the informations stored by OpenFOAM®. Theoretically speaking these are inputs by user that already known all these quantities. However it is remembered that they are written on a file that OpenFOAM® reads and by them it creates a mesh, taking memory of this. So to retrieve the listed data it is necessary to pass by what is stored by OpenFOAM®, in particular the coordinates of single points. To do this it is created a code, called `geometry_study.H`, that is included by the main one and it allows the evaluation of the stated data and store them. However this piece of code is quite basic since it is built ad-hoc for the applications investigated. Only simple geometries like parallelepipeds with starting structured mesh can be studied.

The decision that this is the first procedure that is done is driven by a limitation of OpenFOAM®. Since it works with FVM, the best structure to be studied is three dimensional. So the tool can read only these geometries. A two or one dimensional study is allowed but they have to start from a three dimensional element. So this means that for a 2D application, this has to be a three dimensional system and the user have to specify that some faces are not considered. There could be the risk that, even if a dimension is not relevant, this will have equally effect on the evaluation of the quantities (for example in the evaluation of volume: even if it is a flat application it is considered the volume). This problem becomes quite evident for the evaluation of coefficients and a further treatment will be described later.

3.3 Flux computation

After the geometry study, it starts the flux evaluation for the reference solution. This is relevant for the evaluation of coefficients and also to get the criticality value. Since OpenFOAM® is a class of libraries written in C++, everything has to be defined and declared. For this reason the first thing to do is the creation of mesh and then the definition of the flux, the properties and dummy values, used for the correct work of the solver. It is remarkable that these quantities are given in $\frac{neutrons}{cm^2 \cdot s}$, cm and cm^{-1} but OpenFOAM® works only with the reference SI unit so, immediately after the declaration there is the conversion to meters. Then the solver requests the insertion of power to user by input: this will be used

for the normalization of flux at the end of the evaluation. In case of more than one group, it is asked also the value for normalization of the adjoint. At this point it starts the real evaluation of flux with a `while` loop, following the power method. In particular, what is done by Di Lecce [14] was used as support. The procedure consists in the separation of diffusion and fission part and to study one as the result of the other. So for the diffusion part (for simplicity it is showed the one for one group):

$$\nabla \cdot \left(D(\vec{r}) \nabla \varphi^n(\vec{r}) \right) - \Sigma_a(\vec{r}) \varphi^n(\vec{r}) = Q^n(\vec{r}) \tag{3.1}$$

Where in $Q^n(\vec{r})$ are hidden the values obtained by the fission part (in case of first iteration is an assumed value). Then it is evaluated the k effective as function of previous information and the actual one, as:

$$k^n = k^{n-1} \frac{\int \nu\Sigma_f(\vec{r}) \varphi^n(\vec{r}) dV}{\int \nu\Sigma_f(\vec{r}) \varphi^{n-1}(\vec{r}) dV} \tag{3.2}$$

At this point, having the criticality and the flux for the n iteration the errors are evaluated on both quantities, one as absolute error and the other as norm two, so:

$$err_k = \left| k^n - k^{n-1} \right| \tag{3.3}$$

$$err_{\varphi(\vec{r})} = \frac{\sqrt{\sum_{i=0}^{Ncells} \left(\varphi_i^n - \varphi_i^{n-1} \right)^2}}{\sqrt{\left(\varphi_i^{n-1} \right)^2}} \tag{3.4}$$

Where the superscript $n-1$ refer to the flux evaluated in the previous iteration. The tolerances are fixed at 10^{-15} for the first and 10^{-13} for the second. So until the errors are bigger than these two values the while loop continues to work.

Now that the flux of current iteration is computed it is calculated how neutrons are distributed trough fission mechanism constituting the new $Q(\vec{r})$ term for the next generation. So it is evaluated the equation:

$$\frac{1}{k^n} \int \nu\Sigma_f(\vec{r}) \varphi^n(\vec{r}) dV = Q^{n+1}(\vec{r}) \tag{3.5}$$

Where this time $Q(\vec{r})$ represents the unknown term.

With this there is the end of cycle and the errors decide if continue or get

out of this. In this case there is the normalization of the flux depending on the decided value of power. Having the flux and the necessary power, it is computed the constant conversion as:

$$C = \frac{P}{\int E\nu\Sigma_f(\vec{r})\varphi(\vec{r})dV} \tag{3.6}$$

where E assumes the value of energy produced by fission, so $200MeV$ that in Joule is $3.2 \cdot 10^{-11}J$. Having this the flux is ready to be used for the evaluation of weights.

In case of two groups the adjoint is different from the flux and another while loop is requested. The same procedure just described is followed, with the needed differences due to the different equations involved. What is written up to now can be summarized in the figure 3.2

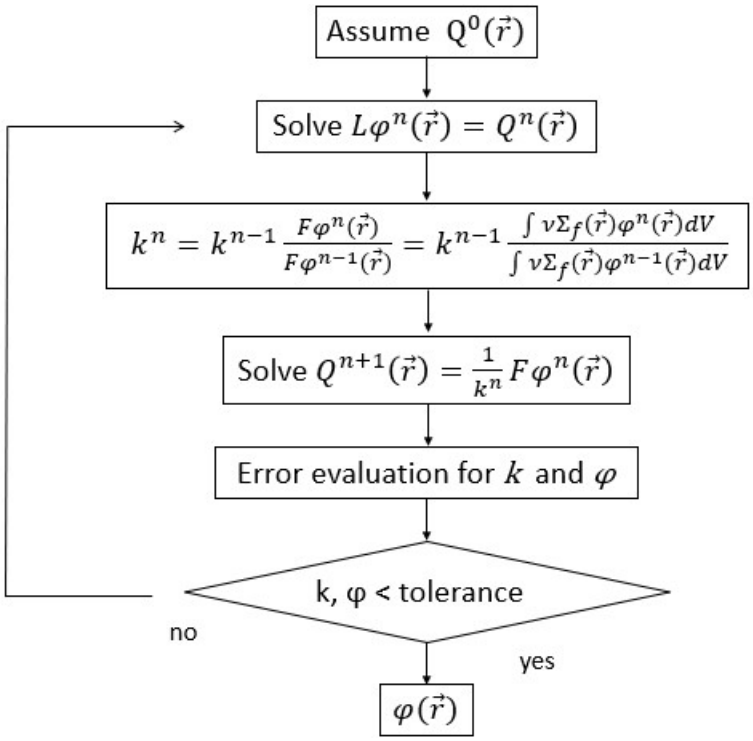


Figure 3.2: Steps followed for reference solution

It is important to remark something about creation of this algorithm. It is based a lot on C++ language and many for and while loop are present in order to compute the flux. This is quite obvious but it is reported to focus that even also other tools based on C or C++ language can be used. However it appears clear the usage of OpenFOAM® that allows the discretization of the equation without further lines of code. Indeed the main line of the code are:

```
fvScalarMatrix diff_term
(
-fvm::laplacian(DD,fi)
+fvm::Sp(sa,fi)
);
solve(diff_term == fvc::SuSp(one,qq));

...

 fvScalarMatrix qq_term
(
fvm::Sp(one,qq)
);

solve(qq_term == (1/kk_eff_new)*fvc::SuSp(nisf,fi));
```

Lines of code for the evaluation of flux in one group

As it can be seen the discretization procedure by OpenFOAM® is quite intuitive. It is also important to say that `fvm::` discretizes an implicit term while `fvc::` an explicit one. So in the first part the known term, is the assumed flux or the one computed in the previous iteration, while in the second part it is given by the flux and the fission cross section.

Finally it is remarked that all this occurs before the start of time and there is the possibility for the user to stop the simulation saving the data: so the solver could be used also to compute steady-state flux in one or two groups.

3.4 Change of mesh

The informations retrieved by the file `geometry_study.H` are used for coefficients but also to build a new mesh. It is already said that a new mesh is needed since OpenFOAM® has stored one for a two or three dimensions. To speed up the all process it is possible to ask to solver to create new mesh alone. This is done by the code "`Make_mesh.H`" that is able to transform the 2-D mesh (written in a proper way) in a 1-D mesh. However the user is free to decide to make on his own this process but advising at the end of the modification the solver. Rewritten the file, the code updates the new mesh launching alone the command by the C++ function `system` that writes on the terminal what is necessary to do. In this way, after the change in the two different ways, and sent the order, OpenFOAM® reads the file and creates the new mesh. This will be very similar to the previous one with the exception that now the discretization of cells is only on one direction. In case of pure energy study the mesh is prepared for a zero dimensional computation.

All this work was necessary for the mesh but not for quantities like flux since the multidimensional ones are already created for two or three dimensions computations and they will not be used with the new mesh. Equally, files that refer to amplitude are already written for a 0-D or a 1-D computation.

3.5 Weights computation

After the change of the mesh that is remarked to be an uncommon passage for standard simulation, the weights are evaluated. Before this a series of requests are posed to the user to decide the physics to study and the related aspects. Indeed velocities are required since they will be used for the coefficient in front of the partial derivative of time. Moreover as it can be red in the section 2.5 the reference equations, even considering the delay insertion, are the same. So another possibility is the decision to consider presence of delayed neutrons. This is an important thing because consequently weights and structure of equations will change. Finally in case of the insertion there is the possibility to decide the fraction of delayed and the decay constant. All these options are thought to give a more versatility to the solver giving the possibility to use it for different materials and/or different considerations.

Regarding the coefficients the integration is done in a numerical way implemented by the author. The mathematical formula are not reported since they are the ones that can be red in 2.3 and 2.5 so they are not reported here. The standard code, for example for $\alpha_1(y)$ coefficient that is equal for all computations is:

```
// ALFA1
jj=0; sum=0; kk=0; ii=0; scalar ww=0;

forAll(alfa1,cellI)
{
        alfa1[cellI]=0;
}

while(jj<(nx*ny*nz))
{
        sum=sum+psi1[jj]*fi1[jj]*(1/vv1)*dx*dz;

        if (kk==(nx-1))
        {
                alfa1[ii]=alfa1[ii]+sum;
                kk=0;
                ii=ii+1;
                sum=0;
        } else
        {
                kk=kk+1;
        }

        if(ww==(nx*ny-1))
        {
                ww=0;
                ii=0;
        }else
        {
                ww=ww+1;
        }

        jj=jj+1;
}

alfa1.write();
Info<<"alfa1 "<<endl;
```

Evaluation of coefficient $\alpha_1(y)$ for space energy discretization.

In the first line there is the initialization of quantities used for the successive evaluation: `jj` is the total counter of cells, `sum` can be considered the partial result of the integral, `kk` is the counter of cells in x direction and `ww` the one of plane, finally `ii` is the counter for the number of cells on y direction. It is important to remember that OpenFOAM® is a C++ library so the first value is `0`. This is the reason why the `while` loop works until the counter is equal to the number of cells since the first cell is the zero one. Then there is the integral that, as it can be seen, is the numerical formulation of what is written

in 2.3. The following `if` detects if the `kk` counter reached the number of cells in horizontal direction. If it is, the `ii` cell of the coefficients acquires the value of integral computed up to now and there is the reset of `kk`, since the horizontal limit is reached. Also `sum` is zeroed, since the integration along x is finished, and `ii` is increased by one in such a way to consider the next y cell. In `else` case there is just the update of `kk`. There is then the second `if` which counts the cell on plane $x-y$. The quantities like `fi` are stored as vector in a particular order: firstly there are the values of all cells located in the lowest y and z point; then there are the values of all the cells located in the successive y but same z point and so on (theoretically refer to points is wrong but it is for explicative purposes). So the data are layered firstly on y direction then along z direction. This is the reason why there is `ww` that counts the cells on a layer whose normal is the z direction. When it is equal to the number of cells on a layer, it is reset. This also explains why the coefficient is summed in the first `if`: in this way it is accounted the contribute of the previous cells located at a lower level in z direction. At the end there is the store of the coefficient and the message on terminal of ended computation.

It is remarkable to see the presence of the three differential. One of them should not be present in case of two dimensional study but in order to have a solver usable also for a 3D case it is inserted. In `geometric_study.H` the number of cells along the directions are evaluated and if it is found that on z is one `dz` is set equal to one.

For energy discretization the implementation is much more easier since the integration is in whole volume. So the standard formula for the numerical evaluation of integrals can be implemented like:

```
// ALFA1

jj=0; sum=0; kk=0; ii=0;

while(jj<(nx*ny*nz))
{
        sum=sum+psi1[jj]*fi1[jj]*(1/vv1)*dx*dy*dz;
        jj=jj+1;
}
forAll(A1,ii)
{
        alfa1[ii]=sum;
}

Info<<"alfa1 "<<endl;
alfa1.write();
```

Evaluation of coefficient $\alpha_1(y)$ for energy discretization.

As it can be seen it does not require any intermediate command in the `while` loop and it updates the value of the coefficients at the end of the computation by the `for` loop. OpenFOAM® to solve the equations request that the data are discretized in cells.

3.6 Amplitude computation

The amplitude computation is the final passage of the code and it is quite obvious since it requires just the translation of what is written in mathematical formula in section 2.3 with the due assumptions, in OpenFOAM®. For prompt space discretization (equivalent if it is in one or two groups) this is:

```
alfa*fvm::ddt(A)==
+ gamma*fvm::div(phi,A)
+ delta*fvm::laplacian(A)
+ fvm::Sp(eta,A)
```

Again it is noted that the syntax by OpenFOAM® is very intuitive, so no term is explained except the divergence. The `phi` inside the brackets is a numerical expedient to account the possibility to do the derivative. Since no way to do the derivative only on one dimension it is performed the divergence but this requires a vector that is not the case of amplitude. So to it, it is coupled a vector, `phi`, in such a way that it can be done the derivative on the vector and on the amplitude. The vector is modelled to be neutral for the final result. Mathematically, `phi` is U and it is $U = [0\ 1\ 0]$ so the result is:

$$\texttt{div(phi,A)} = \nabla\Big(\vec{U} \cdot A(y,t)\Big)$$

$$= \nabla\Big(\vec{U} \cdot A(y,t)\Big) = \frac{\partial U_x A(y,t)}{\partial x}\hat{e_x} + \frac{\partial U_y A(y,t)}{\partial y}\hat{e_y} + \frac{\partial U_z A(y,t)}{\partial z}\hat{e_z}$$

$$= \frac{\partial U_y A(y,t)}{\partial y} = U_y \frac{\partial A(y,t)}{\partial y} = \frac{\partial A(y,t)}{\partial y}$$

Where $\nabla\Big(\vec{U} \cdot A(y,t)\Big)$ is exactly `div(phi,A)`. The lines of code are inside a while loop that endures until it is reached the end prescribed in `controlDict` file. The only quantities that are saved are the amplitudes.

3.7 Evaluation of amplitude for more than one group

The discretization in energy groups give two different values of amplitude. However the important thing is the overall one, function of the power of both groups. This is evaluated as normalization of the power on time and as function of both amplitudes:

$$A(t) = \frac{\int p_1(y)A_1(y,t)dy + \int p_2(y)A_2(y,t)dy}{\iint \nu\Sigma_f(\vec{r},E)\varphi(\vec{r})dVdE} \tag{3.7}$$

where $p_1(y)$ and $p_2(y)$ are evaluated as integrals on x and z direction as (so the numerical passages are the same reported above):

$$p_1(y) = \iint \nu\Sigma_{f,1}(\vec{r})\varphi_1(\vec{r})dxdz \tag{3.8}$$

$$p_2(y) = \iint \nu\Sigma_{f,2}(\vec{r})\varphi_2(\vec{r})dxdz \tag{3.9}$$

In this way it can be obtained one value for the amplitude starting by the two.

3.8 Boundary conditions

A particular focus has to be done for the boundary conditions. OpenFOAM® allows the insertion of a lot of boundary conditions: the easiest and the most suitable for this purposes are the Neumann or the Dirichlet conditions. However the real boundary conditions want that a combination of the flux and the gradient at boundaries gives zero. This is not present in OpenFOAM®, maybe because it is too specific. So for the sake of simplicity the boundary conditions are set to be equal to zero. This assumption can be considered correct or give similar values if the diffusion length is much smaller than the system dimensions. For the evaluations that will be done, in chapter four it can be seen that this is so. It is remarkable that OpenFOAM® considers equal to zero not the first cells but the points at the edges. This means that first cell will be affected by a point magnitude that is equal to zero and other points that have much higher values due to the values of flux. The final value of flux is different from zero but it is still lower than the highest value observed.

Equivalent conditions are imposed also for the adjoint. However set of boundary conditions can be done directly by files contained in `0` folder and no code writing is requested.

Regarding the amplitude the zero-gradient condition can be easily adopted.

3.9 Schemes and algorithms

A great advantage of OpenFOAM® is the possibility to choose the algorithm for the evaluation of quantity used. This enlarges the options available for the user and the faculty to choose the most suitable numerical discretization for the simulation procedure. Moreover remembering that OpenFOAM® is completely free there is also the possibility, starting from the already available lines of code, to create new scheme solver. Finally it is remarkable that the decision of these quantities can be choosen inside the folder constant, in the `fvSchemes` file. The schemes selected for the evaluation of the quantities involved are:

- `ddtSchemes`: bacward or steadyState.

- `gradSchemes`: Gauss linear.

- `divSchemes`: Gauss linearUpwind.

- `laplacianSchemes`: Gauss linear corrected.

- `interpolationSchemes`: linear.

Chapter 4

Application on two groups case

In order to verify the validity of the method it is decided to study some applications: it is recalled the study case done by Dulla and Nervo [15] in their paper. To this, it is added the study of new transients and also new configurations; moreover also the properties of the system considered are different. In line with their work there is the simplicity of the case. Since it is a theoretical study, there is only the need to asses the validity of the method, and so even a simple case can give an idea of what occurs with the different discretizations. In this chapter will be showed many applications in order to present the work and observe the differences involved as well as the enhancements that are necessary for the method.

4.1 Assumptions and properties

It is immediately showed the study using two groups flux. It is believed to be the most interesting and quickest evaluation to do to asses the method. To study this configuration a list of assumptions has to be done, similarly to the ones in section 3:

- study is just in two groups;
- the system is a two dimensional system;
- there is not a source;
- all the neutrons born are fast;
- there is only one family of precursors;
- all the delayed neutrons are fast;

The perturbations studied are two, the same ones done in the paper by Dulla and Nervo [15]. The first one is wide and horizontally distributed along all the

core while the second is just a little part much shorter on horizontal direction but again 8 *cm* long in vertical direction. It is showed in figure 4.1.

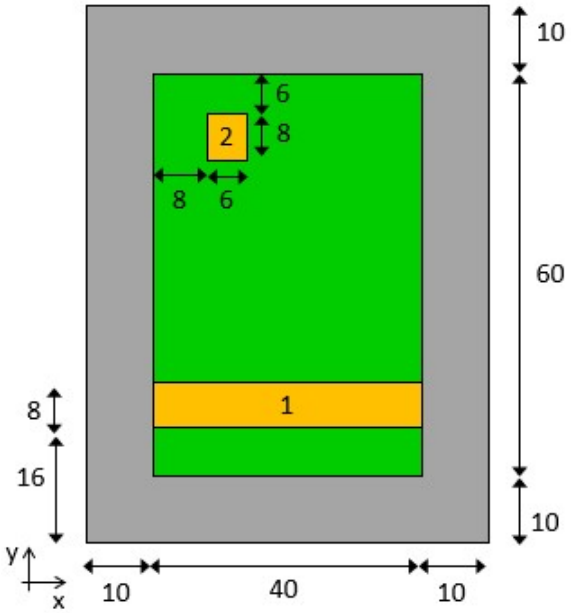


Figure 4.1: The system studied. The grey zone is the reflective material, the green is the core and the yellow parts are the regions where perturbation occurs in separately.

The properties are the one listed in table 4.1. They are not the ones of a particular material but they are selected in such a way that the reference flux at the initial instant produces a criticality value equal to one. Actually the k value is equal to 1.00004 that can be considered approximately equal to the reference value. With this consideration there is the implicit choice to consider that at initial condition the system is in steady state and it is providing always the same power.

Region	Group	D_g [cm]	$\Sigma_{a,g}$ [cm^{-1}]	$\Sigma_{g\to g+1}$ [cm^{-1}]	$\nu\Sigma_f$ [cm^{-1}]	χ [-]
Core	1	1.01	0.01	0.008	0.003583	1.0
	2	1.2	0.1	-	0.1498	0
Reflector	1	0.9	0.01	0.0095	0.0	1.0
	2	0.8	0.1	-	0.0	0

Table 4.1: two groups data.

The perturbations firstly inserted are the ones showed in the table 4.2. As it can be seen they are different in magnitude. The first one, that is only the increase of 10% of the absorption cross section, is in line with the extension of the zone modified. Since it is a wide zone, the perturbation value is quite small. Conversely for the other transient, that is done in a little part of the system

the magnitude of the change is quite high. Despite this it will be shown that the first modification is the more impactive.

	$\delta\Sigma_a/\Sigma_a$ [%]
HE-1	10
HE-2	40

Table 4.2: Transients HE

This is not the only transient studied, in particular they are:

- *HE transients* where there is the augmentation of absorption cross section;
- *RI transients* where the fission cross section is increased in such a way to reintroduce the reactivity lost;
- *SS transients* where there is the increase of scattering cross section following the data in the table 4.2;
- *FA transients* where there is the increase of fission cross section;
- *DD transients* where there is the increase of diffusion length following the data in the table 4.2;
- *_C transients* where there is the insertion of delayed.

These are the main transients done but they can change depending on the case. With two fluxes at least three different discretizations are possible (well four but then it is just the quasi-static method). The mathematical formulation beyond these discretizations is already described in section 2.3 in its general way. Now they are rewritten considering all the possible perturbations (except the one of source) and the assumptions done above. The equations become: for

space-energy:

$$
\begin{cases}
\left\langle w_1(\vec{r})\middle|\dfrac{1}{v_1}\varphi_{1,0}(\vec{r})\right\rangle_{\vec{r}-y}\dfrac{\partial A_1(y,t)}{\partial t} = \left\langle w_1(\vec{r})\middle|\Big(D_1(\vec{r})+\delta D_1(\vec{r})\Big)\varphi_{1,0}(\vec{r})\right\rangle_{\vec{r}-y}\dfrac{\partial^2 A_1(y,t)}{\partial y^2} \\
\qquad + \left\langle w_1(\vec{r})\middle|2\Big(D_1(\vec{r})+\delta D_1(\vec{r})\Big)\dfrac{\partial\varphi_{1,0}(\vec{r})}{\partial y}\right\rangle_{\vec{r}-y}\dfrac{\partial A_1(y,t)}{\partial y} \\
\qquad + \left\langle w_1(\vec{r})\middle|\delta D_1(\vec{r})\nabla^2\varphi_{1,0}(\vec{r}) + \Big(\delta\nu\Sigma_{f,1}(\vec{r})-\delta\Sigma_{r,2}(\vec{r})\Big)\varphi_{1,0}(\vec{r})\right. \\
\qquad \left. - \nu\Sigma_{f,2}(\vec{r})\varphi_{2,0}(\vec{r})\right\rangle_{\vec{r}-y} A_1(y,t) \qquad (4.2) \\
\qquad + \left\langle w_1(\vec{r})\middle|\Big(\nu\Sigma_{f,2}(\vec{r})+\delta\nu\Sigma_{f,2}(\vec{r})\Big)\varphi_{2,0}(\vec{r})\right\rangle_{\vec{r}-y} A_2(y,t) \\
\left\langle w_2(\vec{r})\middle|\dfrac{1}{v_2}\varphi_{2,0}(\vec{r})\right\rangle_{\vec{r}-y}\dfrac{\partial A_2(y,t)}{\partial t} = \left\langle w_2(\vec{r})\middle|\Big(D_2(\vec{r})+\delta D_2(\vec{r})\Big)\varphi_{2,0}(\vec{r})\right\rangle_{\vec{r}-y}\dfrac{\partial^2 A_2(y,t)}{\partial y^2} \\
\qquad + \left\langle w_2(\vec{r})\middle|2\Big(D_2(\vec{r})+\delta D_2(\vec{r})\Big)\dfrac{\partial\varphi_{2,0}(\vec{r})}{\partial y}\right\rangle_{\vec{r}-y}\dfrac{\partial A_2(y,t)}{\partial y} \\
\qquad + \left\langle w_2(\vec{r})\middle|\delta D_2(\vec{r})\nabla^2\varphi_{2,0}(\vec{r}) - \Sigma_{1\to 2}(\vec{r})\varphi(\vec{r})_{1.0}\right. \\
\qquad \left. - \delta\Sigma_{r,2}(\vec{r})\varphi_{2,0}(\vec{r})\right\rangle_{\vec{r}-y} A_2(y,t) \qquad (4.3) \\
\qquad + \left\langle w_2\middle|\Big(\Sigma_{1\to 2}(\vec{r})+\delta\Sigma_{1\to 2}(\vec{r})\Big)\varphi_{1,0}(\vec{r})\right\rangle_{\vec{r}-y} A_1(y,t)
\end{cases}
$$

for energy:

$$\begin{cases} \left\langle w_1(\vec{r})\middle|\frac{1}{v_1}\varphi_{1,0}(\vec{r})\right\rangle_{\vec{r}}\frac{dA_1(t)}{dt} = \left\langle w_1(\vec{r})\middle|\delta D_1(\vec{r})\nabla^2\varphi_{1,0}(\vec{r})\right. \\ \qquad\qquad + \Big(\delta\nu\Sigma_{f,1}(\vec{r}) - \delta\Sigma_{r,1}(\vec{r})\Big)\varphi_{1,0}(\vec{r}) - \nu\Sigma_{f,2}(\vec{r})\varphi_{2,0}(\vec{r})\Big\rangle_{\vec{r}}A_1(t) \\ \qquad\qquad + \Big\langle w_1(\vec{r})\Big|\Big(\nu\Sigma_{f,2}(\vec{r}) + \delta\nu\Sigma_{f,2}(\vec{r})\Big)\varphi_{2,0}(\vec{r})\Big\rangle_{\vec{r}}A_2(t) \qquad (4.3) \\ \left\langle w_2(\vec{r})\middle|\frac{1}{v_2}\varphi_{2,0}(\vec{r})\right\rangle_{\vec{r}}\frac{dA_2(t)}{dt} = \left\langle w_2(\vec{r})\middle|\delta D_2(\vec{r})\nabla^2\varphi_{2,0}(\vec{r}) - \Sigma_{1\to2}(\vec{r})\varphi_{1,0}(\vec{r})\right. \\ \qquad\qquad - \delta\Sigma_{r,2}(\vec{r})\varphi_{2,0}(\vec{r})\Big\rangle_{\vec{r}}A_2(t) \\ \qquad\qquad \Big\langle w_2(\vec{r})\Big|\Big(\Sigma_{1\to2}(\vec{r}) + \delta\Sigma_{1\to2}(\vec{r})\Big)\varphi_{1,0}(\vec{r})\Big\rangle_{\vec{r}}A_1(t) \qquad (4.4) \end{cases}$$

for space

$$\begin{aligned} &\left(\left\langle w_1(\vec{r})\middle|\frac{1}{v_1}\varphi_{1,0}(\vec{r})\right\rangle_{\vec{r}-y} + \left\langle w_2(\vec{r})\middle|\frac{1}{v_2}\varphi_{2,0}(\vec{r})\right\rangle_{\vec{r}-y}\right)\frac{\partial A(y,t)}{\partial t} \\ &= \Big(\Big\langle w_1(\vec{r})\Big|\Big(D_1(\vec{r}) + \delta D_1(\vec{r})\Big)\varphi_{1,0}(\vec{r})\Big\rangle_{\vec{r}-y} \\ &+ \Big\langle w_2(\vec{r})\Big|\Big(D_2(\vec{r}) + \delta D_2(\vec{r})\Big)\varphi_{2,0}(\vec{r})\Big\rangle_{\vec{r}-y}\Big)\frac{\partial^2 A(y,t)}{\partial y^2} \\ &+ \left(\left\langle w_1(\vec{r})\middle|2\Big(D_1(\vec{r}) + \delta D_1(\vec{r})\Big)\frac{\partial\varphi_{1,0}(\vec{r})}{\partial y}\right\rangle_{\vec{r}-y}\right. \\ &\left.+ \left\langle w_2(\vec{r})\middle|2\Big(D_2(\vec{r}) + \delta D_2(\vec{r})\Big)\frac{\partial\varphi_{2,0}(\vec{r})}{\partial y}\right\rangle_{\vec{r}-y}\right)\frac{\partial A(y,t)}{\partial y} \qquad (4.5) \\ &\Big(\Big\langle w_1(\vec{r})\Big|\delta D_1(\vec{r})\nabla^2\varphi_{1,0}(\vec{r}) + \Big(\delta\nu\Sigma_{f,1}(\vec{r}) - \delta\Sigma_{r,1}(\vec{r})\Big)\varphi_{1,0}(\vec{r}) + \delta\nu\Sigma_{f,2}(\vec{r})\varphi_{2,0}(\vec{r})\Big\rangle_{\vec{r}-y} \\ &+ \Big\langle w_2(\vec{r})\Big|\delta D_2(\vec{r})\nabla^2\varphi_{2,0}(\vec{r}) + \delta\Sigma_{1\to2}(\vec{r})\varphi_{1,0}(\vec{r}) - \delta\Sigma_{r,2}(\vec{r})\varphi_{2,0}(\vec{r})\Big\rangle_{\vec{r}-y}\Big)A(y,t) \end{aligned}$$

The first thing that is done is a grid independence.

4.2 Grid independence

A grid independence is done in order to understand the necessary number of cells to have a good study. This is done simultaneously decreasing the number of cells on horizontal and vertical direction, lowering overall the average cell size. The simple geometry involved allows an immediate study of mesh and it is easy to find the mean dimension of the element. Anyway a simple formula can be:

$$\delta x = \sqrt{\frac{Area}{Number\ of\ cells}} \tag{4.6}$$

In the table 4.3 the different configuration of mesh are shown, from the finest to the coarsest.

Mesh #	N. cells	Δy	Avg. size [cm]
1	60×80	1	1
2	30×40	2	2
3	15×20	4	4
4	12×15	5	5

Table 4.3: Mesh data

Starting from the maximum number of cells and halving it on the two directions lead always to the usage of a square (well it is not a square but a parallelepiped for what is said in section 3.2). It is considered only the space energy discretization.

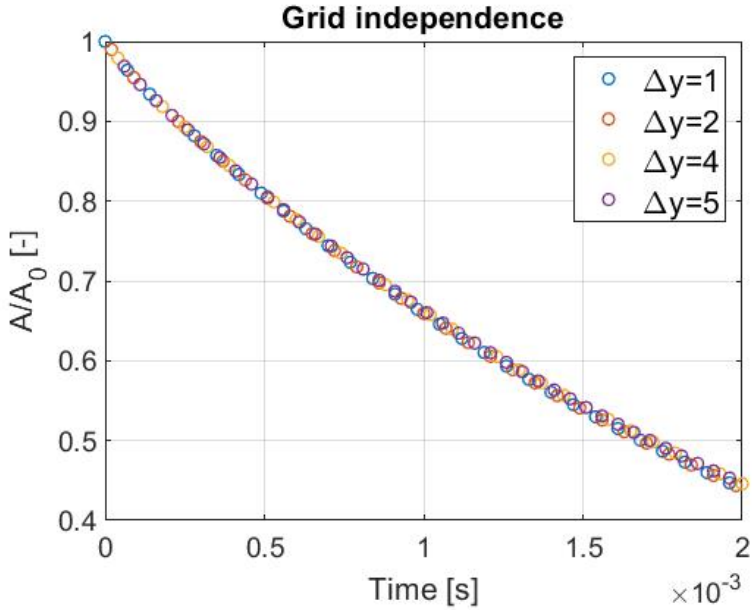


Figure 4.2: Grid independence

It seems that even a poor mesh allows a good study. Anyway the mesh selected is the first one. This decision was driven by the motivation of the simple proportionality between the number of cells on a direction and the dimension of this. No doubling is requested that can be a source of error. This choice was permitted by the still low computational time requested. For sake of simplicity it is assumed that the mesh is good to study all the problems involved and suitable for all the discretizations.

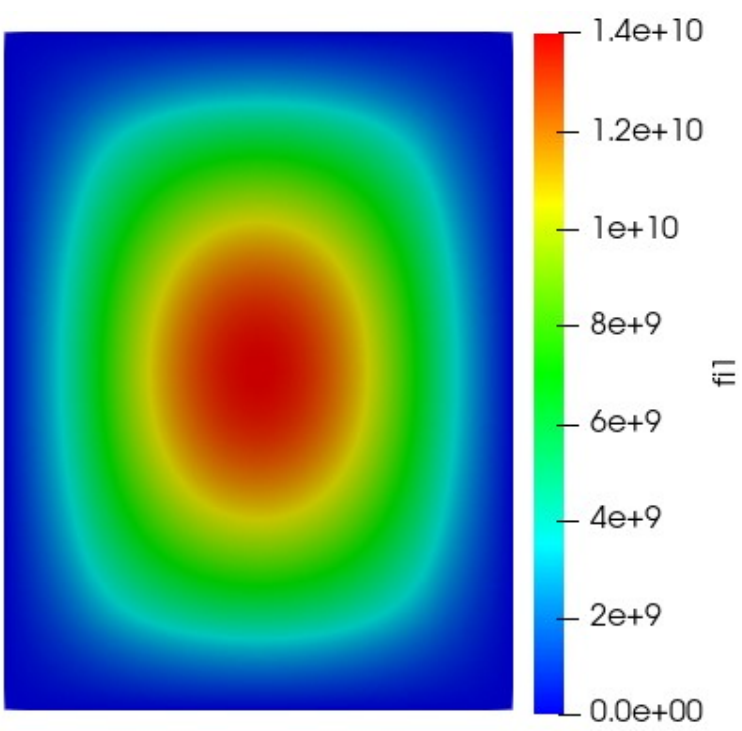
4.3 Map of flux

First of all it is reported the map of the flux (values are in $\frac{neutrons}{(cm^2 \cdot s)}$). As it is written in section 3.3 the flux si not ready to be used at the end of the power method but it has to be normalized respect to the power that user inserts. For all the evaluations (otherwise results are not comparable) the data chosen are:

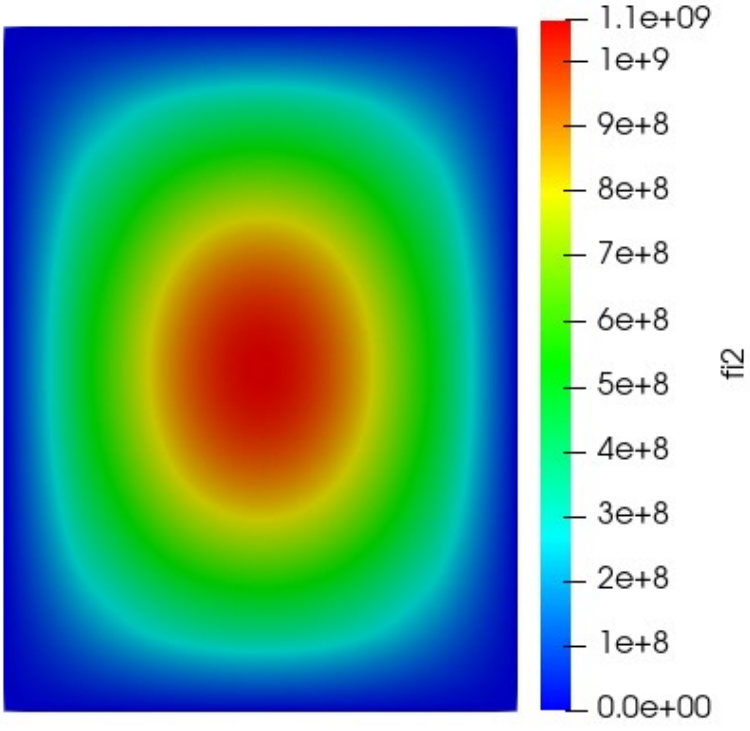
$$P = 1000\ W$$

$$\left\langle w(\vec{r}) \middle| \varphi(\vec{r}) \right\rangle = 10^{20}$$

The value of the power wants to be coherent with the system that is $60 \times 80\ cm$ and the value for importance is so high in such a way to avoid low values and then numerical cancellation. The fluxes and adjoints are showed in figures 4.3 and 4.4.

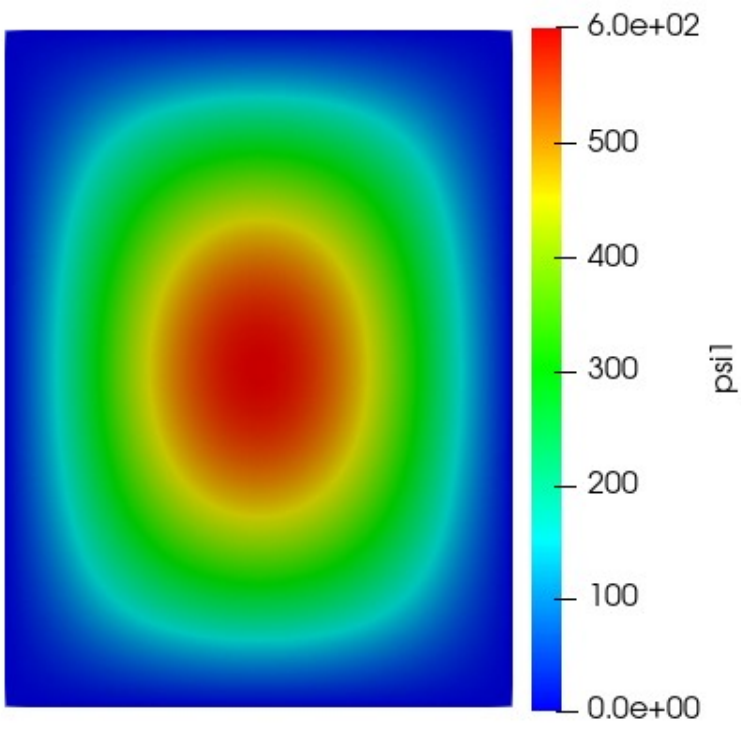


(a) Map of flux for group one

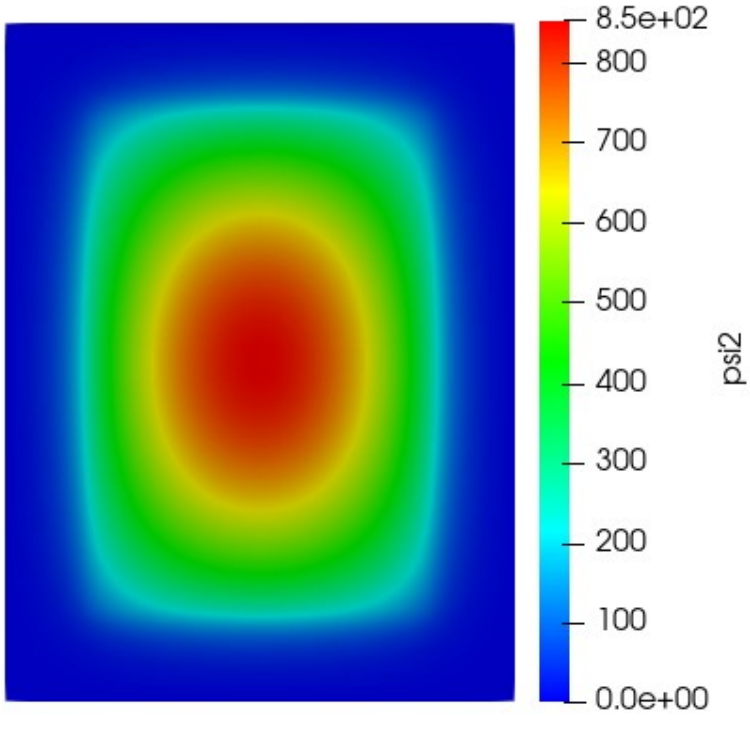


(b) Map of flux for group two

Figure 4.3: Maps of fluxes



(a) Map of adjoint for group one



(b) Map of adjoint for group two

Figure 4.4: Maps of adjoints

As it can be expected all quantities have its maximum in the centre of the system while the minimum at sides. Indeed the system is symmetric respect to

the horizontal and vertical axis. This means that the reference solution could be solved just on a quarter of the system. Despite this, this is not true for the amplitude computations. The map illustrated in figures 4.3 and 4.4 are not the ones obtained by the FVM but they are the ones by the points of the cells. This is only an aesthetic reason since the maps by the cells are pixelated. These also show the flux equal to zero at boundaries. This is not for the cells. Indeed they have to consider both the points at the edges, that are zero, but also the next ones which values are much higher, as it already said in section 3.8.

Physically it has to be underlined that the flux of first group is higher than the one of group two. This was expected since the region of the core has an high fission cross section for group two and all the neutrons pass to group one. The reflector region is too small to represent that the flux of group two is higher. For adjoint it is remarked a similar concept. The adjoint related to second group is higher than the other one as it was expected.

Finally it is necessary to report the values of velocities used. They are not important for the flux computation since this is a steady state evaluation. The values chosen are:

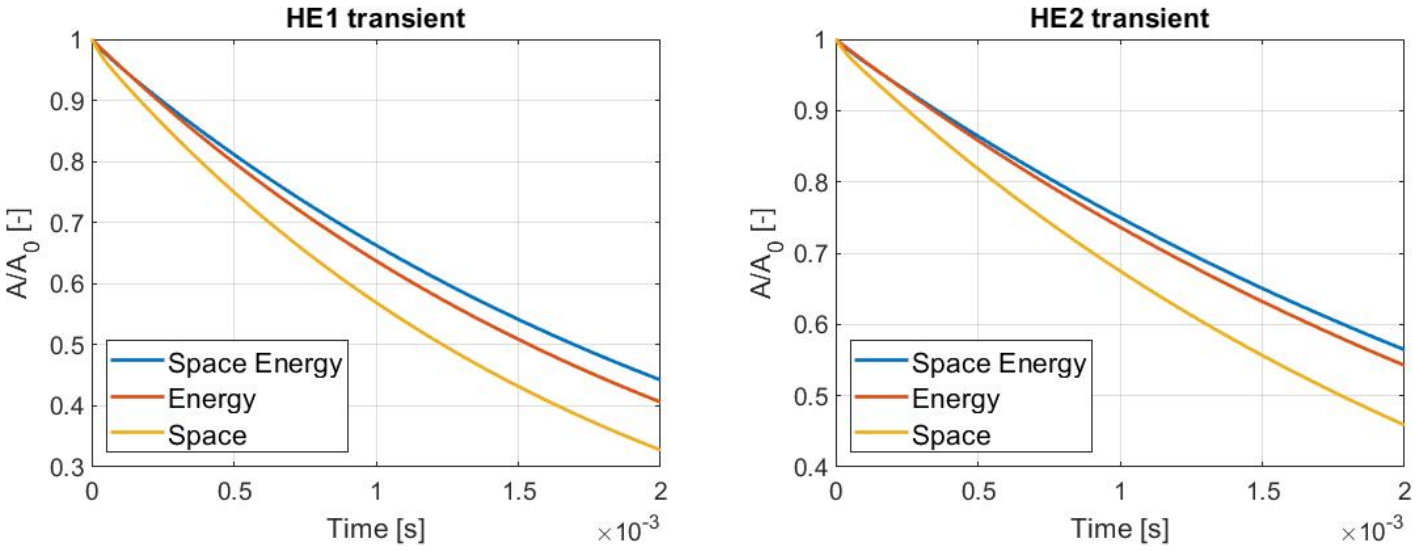
$$v_1 = 16757700 \ \frac{cm}{s}$$

$$v_2 = 404717 \ \frac{cm}{s}$$

They will be important for the evaluation of the coefficients.

4.4 HE transients

This kind of transients involve the increase of absorption cross sections so the drop of the power is expected.



(a) Amplitude for HE1

(b) Amplitude for HE2

Figure 4.5: Amplitudes for perturbations HE

As it is expected overall the amplitude drops. It can be said that wider perturbation comports a more serious decrease. Indeed the powers for transient *HE1* have lower values at time $t = 0.002\ s$. In both graphs it can be observed that the space discretization has smaller values. This can be explained by the structure of equation and so by the physics of the problem. The fact that from two amplitudes there is the collapse to only one group makes disappear the sources for the single group of neutrons. In equation 4.1 there are not sources and the only thing that neutrons can do is to disappear. In case of two amplitudes, it is still accounted the presence of neutrons of the other groups. In this configuration when a neutron disappear it can be for two reasons: it is absorbed or it passes to the other group. In the second case it is not completely lost but it simply appears in the other computation. So if it was a fast neutron and it makes scatter this will appear again as slow; if it was a thermal neutron and it makes fission it will appear as fast. The overall effect is that the amplitude will decrease slowly. For this consideration, it is believed that in case of more groups the amplitude will drop even slower. On the other side considering just one amplitude, a neutron if it is absorbed, disappears from the computation and there is no way to consider it again. To better understand the difference between the discretizations, two tables are reported that show the relative error between the space and space-energy and the energy and space energy discretization at different time instants (values are in %).

HE1	$t = 50\mu s$	$t = 100\mu s$	$t = 150\mu s$	$t = 200\mu s$
E/SE	1.63	3.86	6.03	8.16
S/SE	7.57	14.1	20.2	25.9

Table 4.4: Errors for HE1

HE2	$t = 50\mu s$	$t = 100\mu s$	$t = 150\mu s$	$t = 200\mu s$
E/SE	0.703	1.78	2.84	3.89
S/SE	5.29	9.99	14.2	18.7

Table 4.5: Errors for HE2

As it could be expected the errors tend to be higher for the most hit case so for transients *HE1* and in time it is even worse the discrepancy. Increasing the perturbation of the removal cross section enhances the behaviour of disappearing, that is more remarkable in case of one group study. The consequence of this is that the discrepancies rise passing the time. Conversely they are small at the beginning since all amplitudes start with one.

4.5 RI transients

The *RI transients* provide that for the reactivity removed, this is re-inserted by the augmentation of the number of neutrons born for fission (for the properties, the fission cross section). For this reason it is expected that amplitude should keep constant on value of one. The procedure to perform these transients is:

- evaluation of the k effective for the reference case;
- change the absorption cross section depending on the study case;
- evaluation of the k effective for perturbed case;
- evaluation of the reactivity between the two configurations;
- change of the properties related to the results of previous step;
- evaluation of amplitude.

Due to the system configuration it is quite easy to evaluate the modification needed for ν. The starting k is known and the perturbation on absorption cross section to, so it can be computed the new criticality value and then the reactivity

removed. Making explicit the general formula of reactivity and exploiting the system configuration:

$$\rho = \frac{\frac{1}{k}\langle w(\vec{r})|\delta\hat{F}\phi\rangle - \langle w(\vec{r})|\delta\hat{L}\phi\rangle}{\frac{1}{k}(w\langle\vec{r})|\delta\hat{F}\phi\rangle}$$

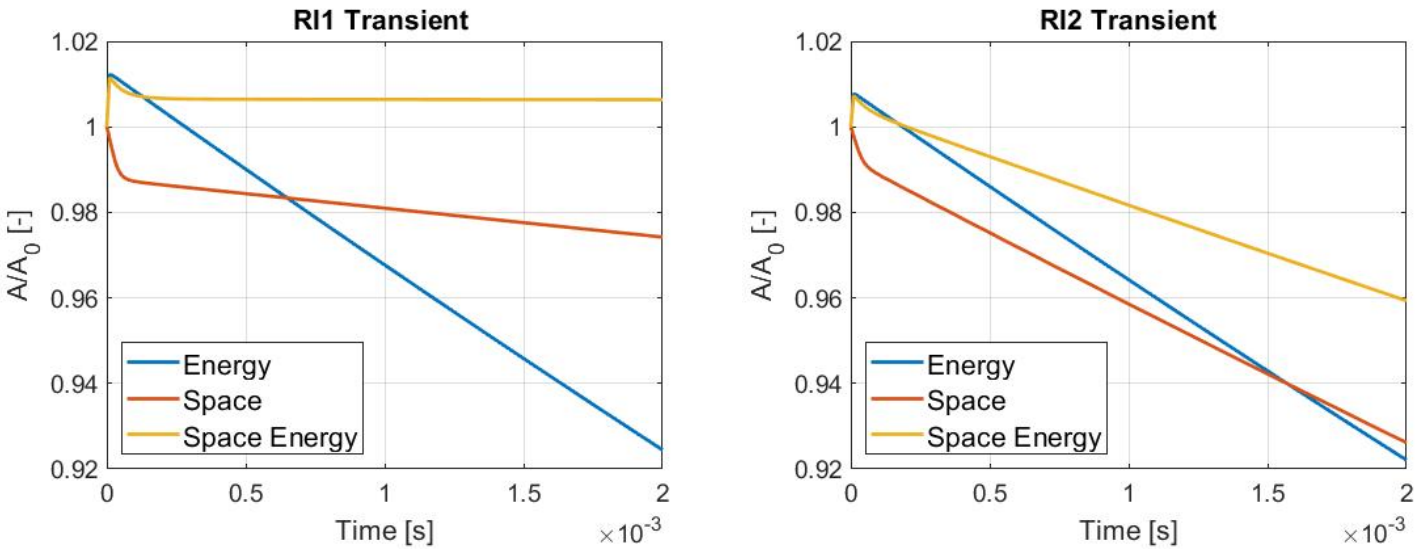
$$\rho = \frac{\langle w|\delta\hat{F}\phi\rangle}{\langle w|\hat{F}\phi\rangle} = \frac{\delta\nu}{\nu} \tag{4.7}$$

It is important to observe that ν is constant and equal for the two groups. This is the reason why it can be extracted from the integrals that are then simplified.

The analysis of this transient is done in a series of systems. The first two, *RI1* and *RI2*, are the compensations of the transients *HE1* and *HE2*. The other ones *RI3*, *RI4*, *RI5* and *RI6* are further studies to see in detail what is occurring. Data are listed in table 4.8.

4.5.1 RI1 and RI2 transients

The results for these configurations are showed in figures below.



(a) RI1 transient **(b)** RI2 transient

Figure 4.6: Transients for RI cases

They show that is not so easy reach the compensation. For transient *RI1* only the space-energy discretization is able to represent the reality of the event arranging the amplitude close to value of one. Even the space discretization is not

so bad realizing an error that is more or less of 2.5% even if it is destined to decrease. The energy discretization does not represent well what is going on in the system. After controlling if there was an error in the solver and observing that even Matlab gives similar results, there has to be something in the equation. The structure of the equations is surely correct since it is a system that couples the two amplitudes, so the answer has to be inside the coefficients. These are just scalar, so they make easier the treatment. If γ_1 and β_1 as well as γ_2 and β_2 are equal between themself, it is easy to see that the solution is constant and equal to the initial value (the derivative is equal to zero). More generally the system 4.4 can be seen as product of matrix and vectors and the eigenvalues and eigenvectors of the first decide the trend of the amplitudes. The data of coefficients, so of matrix, are:

	RI1	RI2
α_1 $[-]$	$5.40687 \cdot 10^{14}$	$5.40687 \cdot 10^{14}$
α_2 $[-]$	$2.32095 \cdot 10^{15}$	$2.32095 \cdot 10^{15}$
γ_1 $[1/s]$	$-9.8657 \cdot 10^{19}$	$-9.85175 \cdot 10^{19}$
γ_2 $[1/s]$	$-9.9075 \cdot 10^{19}$	$-9.87899 \cdot 10^{19}$
β_1 $[1/s]$	$9.9376 \cdot 10^{19}$	$9.895334 \cdot 10^{19}$
β_2 $[1/s]$	$9.81833 \cdot 10^{19}$	$9.81833 \cdot 10^{19}$

Table 4.6: Values of coefficients for energy discretization

Since they are just scalar it can considered the ratio between γ and β and α that is:

	RI1	RI2
γ_1/α_1 $[1/s]$	$-1.82466 \cdot 10^5$	$-1.82208 \cdot 10^5$
γ_2/α_2 $[1/s]$	$-4.26872 \cdot 10^4$	$-4.25643 \cdot 10^4$
β_1/α_1 $[1/s]$	$1.83796 \cdot 10^5$	$1.83014 \cdot 10^5$
β_2/α_2 $[1/s]$	$4.2302 \cdot 10^4$	$4.2303 \cdot 10^4$

Table 4.7: Ratios of the coefficients

It is avoided the analytical solution behind the system of equations but it is known that in case of determinant equal to zero the amplitudes keep their initial value. For *RI1*, determinant is equal to $1.4024 \cdot 10^7$ while for *RI2* it is equal to $1.35147 \cdot 10^7$. The important thing is the relative value respect to addends: these are in the order of $8 \cdot 10^9$ that are much bigger respect to the value of

determinant. It is remarkable that with modification of fission cross section only γ_1 and β_1 are affected, since no change occur in the other terms. The other coefficients are equal to the case *HE1* and *HE2*. From their value it can be understood that they were similar already for a case in which a dramatic drop of amplitude was expected so this means that the method is particularly sensitive to the values of coefficients. Indeed values of γ_1/α_1 and β_1/α_1 for the previous transients were -1.83153×10^5 $[1/s]$ and 1.81598×10^5 $[1/s]$ for the first and -1.82651×10^5 $[1/s]$ and 1.81598×10^5 for the second. The little changes are due to the small perturbation, just 1.21% and 0.78%, lower respect to the ones of absorption cross section. The discretization by energy, integrating overall the space, smooths the contributes of the properties, finding a sort of averaged value. Considering all the space and also that the differentials are equal, coefficients are influenced by the biggest modifications: perturbation of cross section is higher than the other one. On the other side, in space-energy case, the integration only on x direction allows to feel better the change along vertical direction, giving better and more precise results. Integration on x direction limits perturbation contribution just on a part of y; conversely considering both the direction the high values of change affects a lot the final results, so the coefficients.

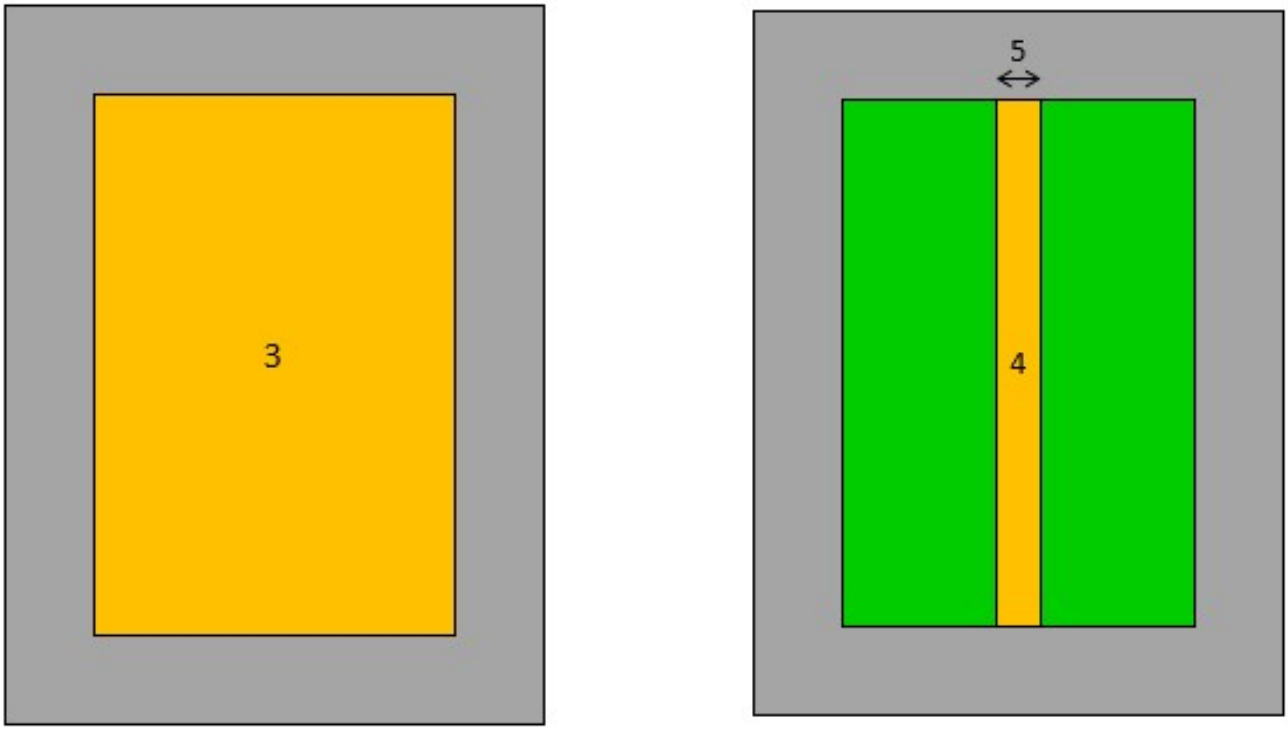
Concerning only on the second reactivity insertion it appears that neither the space energy discretization that before offer very good results, is able to describe what occurs. In some way the considerations done for the energy can be extended to this case. The modified zone is very limited in horizontal space (is just 10% of the system while the other is 66%), in this way it is not well considered what is happening in the x direction, since just a small part of the system is perturbed on this direction. Again the coefficients are much more influenced by that small space where absorption cross section increases, than the rest of core. As in the previous case, in the energy discretization that considers all the space it was found a value influenced by the absorption cross section, here there is something similar for the space energy discretization that fails since the limited perturbation on x.

4.5.2 RI3, RI4, RI5 and RI6 transients

To verify what is just stated above, four others compensated transients are performed: *RI3*, *RI4*, *RI5* and *RI6*. The data that characterize them are listed in the table 4.8 and the perturbed zones are the ones showed in figures 4.7 and 4.8 (the reference system is the same of figure 4.1).

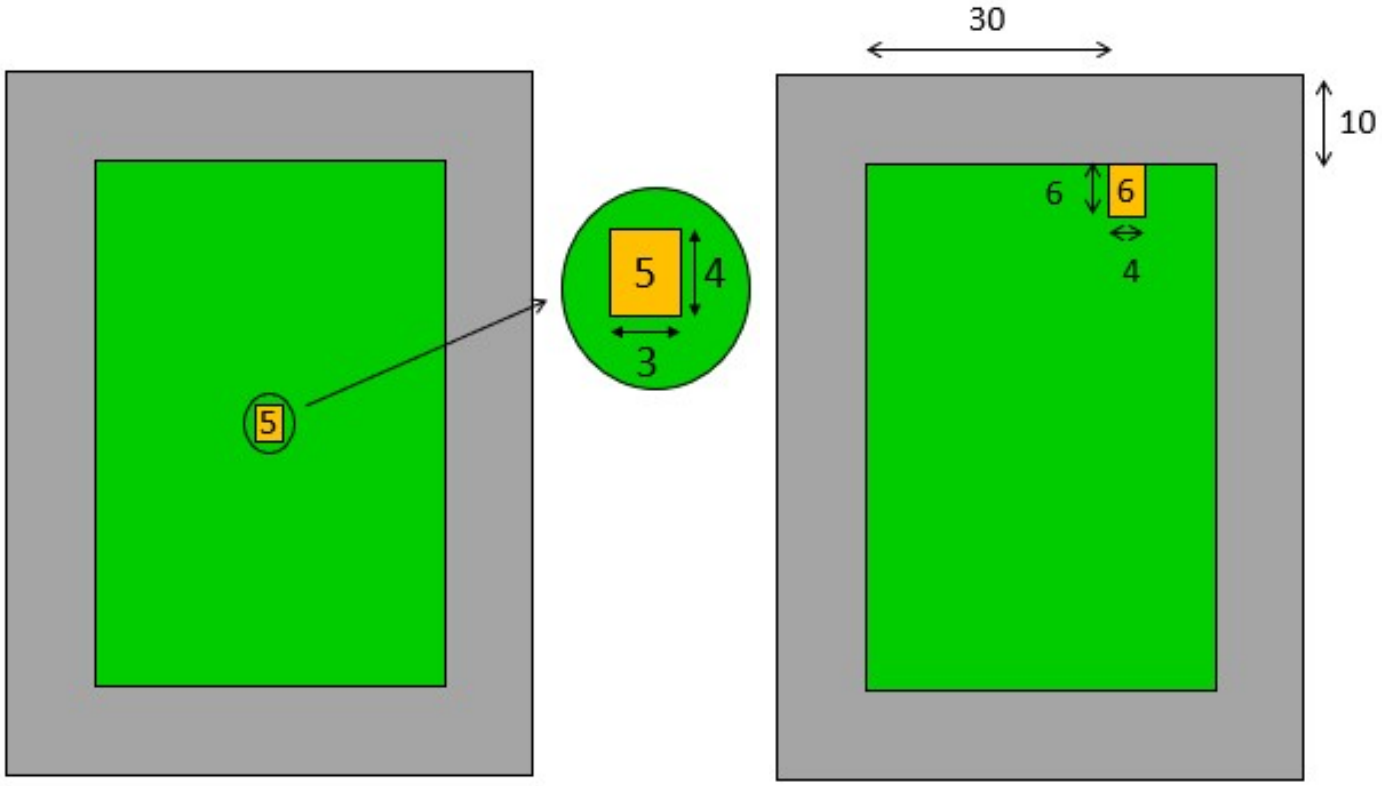
	$\delta\Sigma_r(\vec{r})/\Sigma_r(\vec{r})$ [%]	$\Delta\rho$ [pcm]
RI1	10	1210
RI2	40	780
RI3	1	1372
RI4	4	1168
RI5	45	803.2
RI6	400	816.1

Table 4.8: Reactivity removed/inserted



(a) RI3 system **(b)** RI4 system

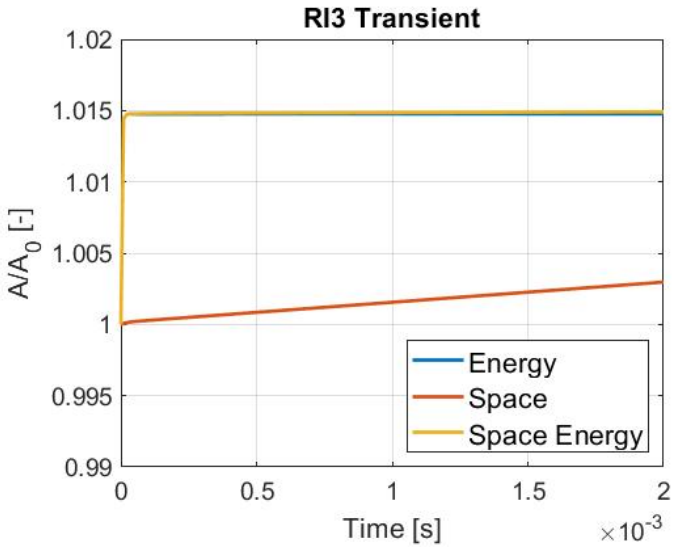
Figure 4.7: Systems for the transients *RI3* and *RI4*



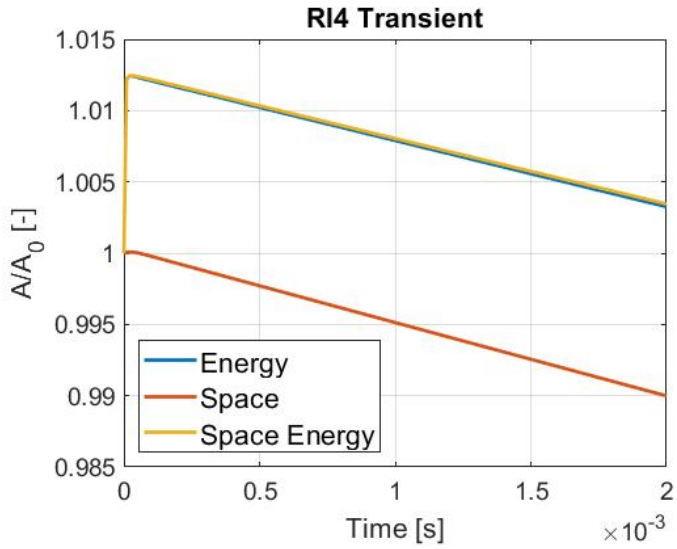
(a) RI5 system **(b)** RI6 system

Figure 4.8: Systems for the transients *RI5* and *RI6*

These transients are very different between themself and they want to asses the ideas that come from the main two transients, *RI1* and *RI2*, that can be considered the reference one since they are similar to ones studied by Dulla and Nervo. They are designed in order to see better the effects that are just marked in the two previous transients. For this reason the reactivity removed and inserted are comparable to the ones caused in *RI1* and *RI2*. The *RI3* transient consists in the perturbation of the absorption cross section in a wide zone, in particular in the whole core region. In this way, in this region every point suffer both the perturbations. The *RI4* time problem is designed in order to see if the space-energy discretization offer good resolution even if the modified part is extended in vertical direction. In this way, it can be understood if there is the sensibility to catch the modifications even if they are very narrow. The transients *RI5* and *RI6* finally want to asses the discrepancies found by the main time problem *RI2*. The results are:

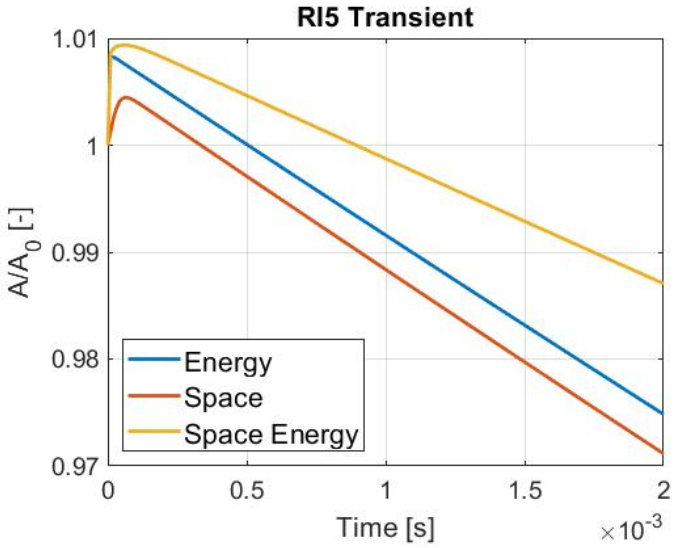


(a) RI3 transient

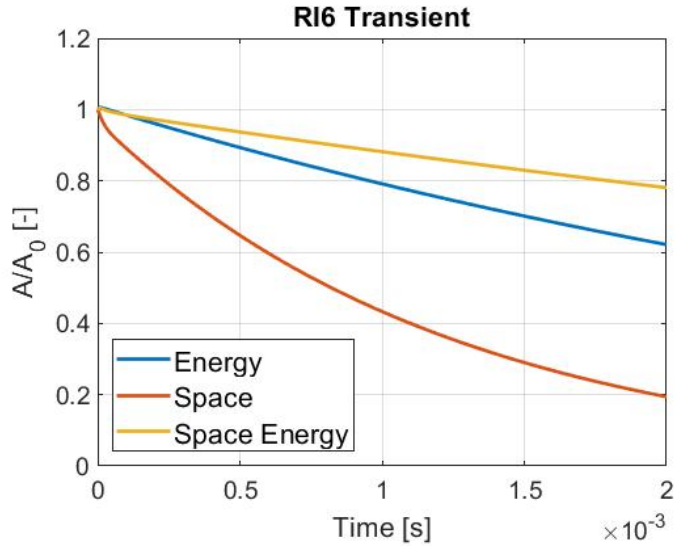


(b) RI4 transient

Figure 4.9: RI3 and RI4 transients



(a) RI5 transient



(b) RI6 transient

Figure 4.10: RI5 and RI6 transients

The results for *RI3* in figure 4.9a show that in case of an overall change of the properties the methods are able to feel this change. Conversely to the first two cases the modifications are small and distributed for all the core. Both space-energy and energy discretization are able to describe what is occurring. The very small difference between them can be explained by the different discretization procedure. Distinguish another variable in space to see what is happening on a particular direction is useless. The perturbations are extended to most part of the core and they are small: integrating on whole space or just on a direction is basically equal. The integration procedure does not lose particularity of perturbation since they are diffused. The space discretization does not seem to follow

the trend given by the other methods. This means that a discretization on energy groups is more important. Despite this, it is remarkable that the errors committed between these discretizations are much lower than the ones realized in the previous analysis. This means that even a discretization on energy is not so necessary, if it is not required an high accuracy. The reason for the good trend of space discretization is again the wide distribution of the perturbations. This makes the system closer to a 0-D system. With such discretization, is treated the total amplitude and not the single ones of the groups. Another difference that appears is that the first two reach immediately the equilibrium, that is not the case for the space focus. This is an expected behaviour since with the first discretizations it is possible to focus on the single group observing better the coupling between the neutrons. The passage from a group to another is studied continuously while for space discretization this is not. However it is remarkable that even time period studied has a certain effect: a long one can result in bad prediction (see slope of amplitude for space).

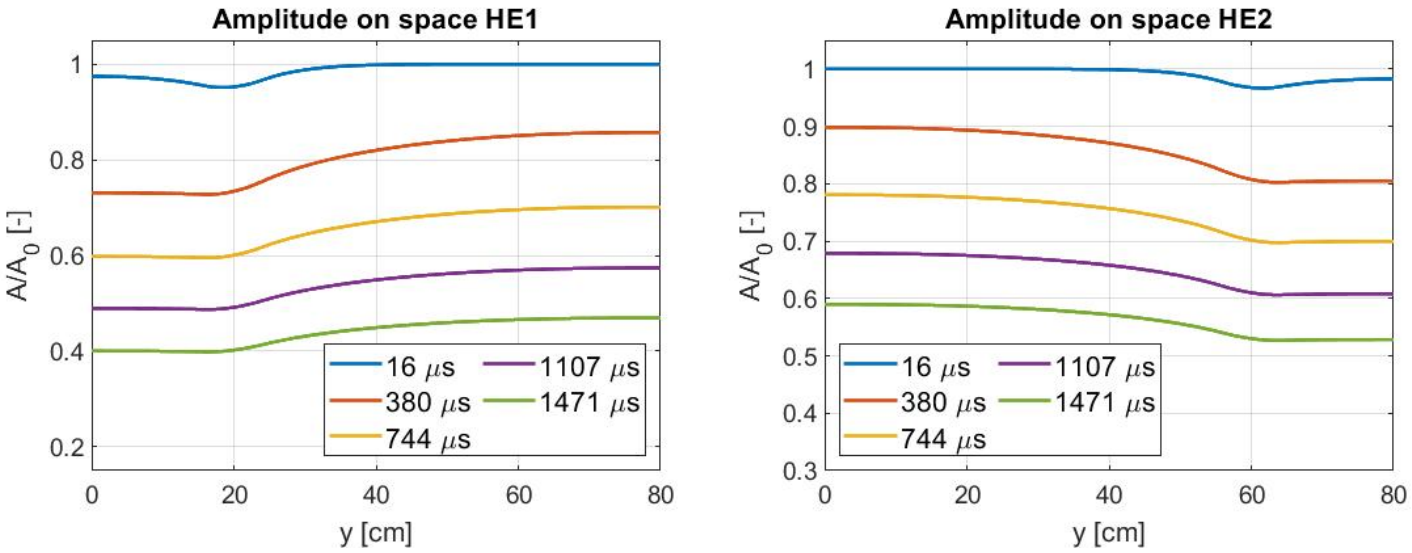
It is showed by the transient *RI1* the good capability of the space-energy discretization to describe a perturbation that is distributed on space. Even the just showed configuration has demonstrated something similar. For this reason it is decided to see if the methods are able to study what is happening just on a narrow line across the core with transient *RI4*. The results do not show a good prediction of the model. The values of amplitudes are generally close to the reference value of one but the slope seem to lead to errors in a longer time. However the discretization of the energy seems to offer results very similar to space-energy discretization. This can be explained by the fact that the perturbation is very short on horizontal direction and integration on x is not able to clearly observe what is occurring, since just a limited part is changed. So the consideration done before for energy, can be extended to this case. The energy discretization offers similar results since the integration still do not observe the limited perturbation but this is is wide on y direction. What leads to failure this study is the integration along x direction, that makes lose how the properties set on space especially on this direction that is integrated using both discretization. It was observed that neither turning the system changes so much the results, so maybe some problems related to nature of perturbation are connected.

The last two transients are created in order to better study the effect of small and limited perturbations in space. The effect of these seems to not be well determined by the three methods. First of all, for *RI5*, it is remarkable that the magnitude is slightly bigger than the other case *RI2* even if it is its half. Indeed this perturbation occurs in the region where the flux is at its maximum: a small perturbation would mean a big insertion or removal of

reactivity due to the big flux involved. The results show again what the previous analysis showed. A limited perturbation on space is not well described by the methods that would fail in a long time study. The reactivity insertion number six again confirm what is just described but it is reported to show that even the location of the modification is very important. As it can be seen by the figure 4.8b the perturbation occurs at the right side of the system, where the flux changes a lot respect to its normal value. As it can be noted the errors tend to be higher. Considering also these limited transients and the previous two, it appears that the energy and space-energy discretization differ when the perturbation is extended on y direction. The second factorization is still function of this direction and it is able to feel the changes related to properties. This is impossible for the other discretization.

4.6 Trend in space for amplitude

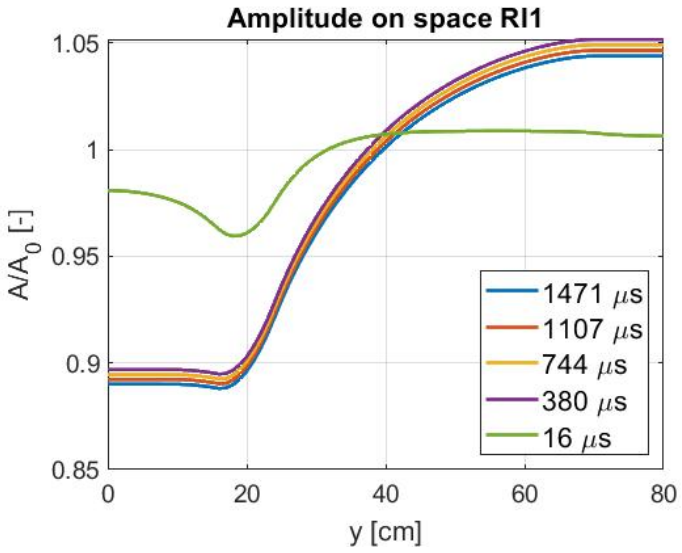
The great power of this method is the capability to see how the amplitude distributes in the space. This can be very important because it could allow the comprehension on the production of power. So also how the temperatures distribute along the reactor and finally, in case of emergency, where is necessary to act firstly to decrease the power generation. The four figures report the shape of amplitude on space, giving a good idea of what is occurring.



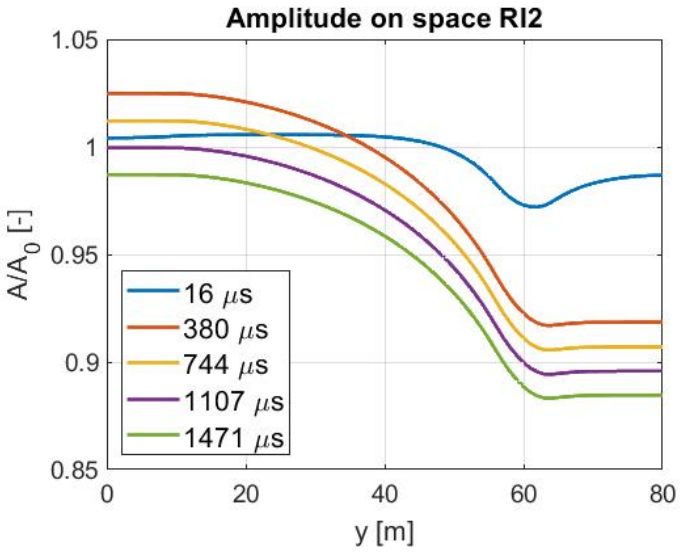
(a) Amplitude on space for HE1

(b) Amplitude on space for HE2

Figure 4.11: Amplitudes on space for HE1 and HE2



(a) Amplitude on space for RI1



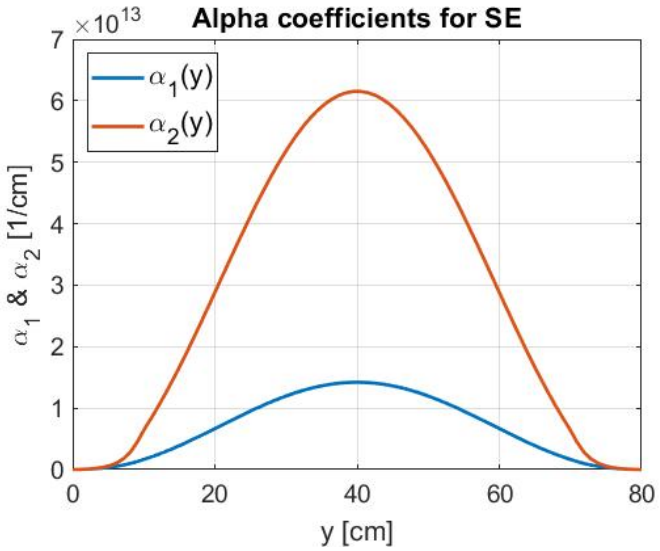
(b) Amplitude on space for RI2

Figure 4.12: Amplitudes on space for RI1 and RI2

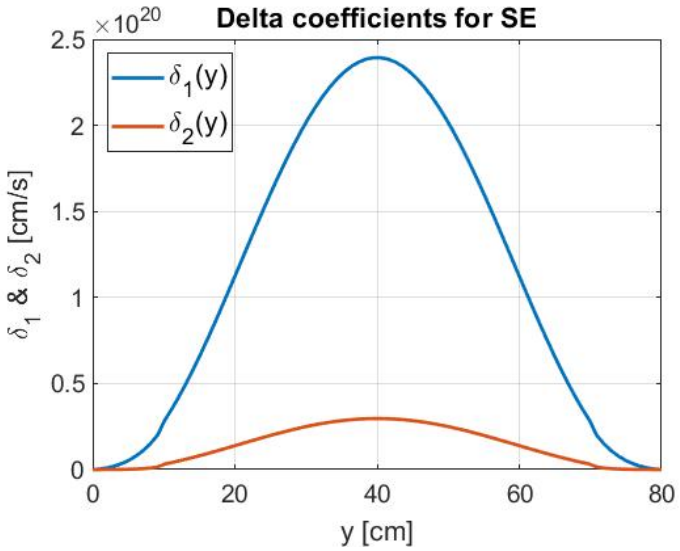
The trends reproduce the amplitude correctly: it can be seen that the minimum amplitude corresponds to the regions where the perturbation is located both for the transient *HE1* and *HE2*. Overall the amplitude tends to decrease. Moreover it appears that the biggest changes occur in the first microseconds while passing time the amplitudes arrange on a certain value. Regarding the transients *RI1* and *RI2*, the amplitude shape in space describes well the system. It can be noted that it decreases in the perturbed regions while it increases in the core one, exactly where there is the increase of fission cross sections. Finally it is remarkable that when there is the compensation of the reactivity removed the amplitude tends to assess to the value of one and to slowly drop. This can be again seen with the overlapping of shapes, in particular in the region of the core for transient *RI1*.

4.7 Coefficients

It is reported the trend of the coefficients. They are function of flux, adjoints and properties. The coefficients change most of the time so the ones for transient *RI1*, for space energy discretization, are shown.

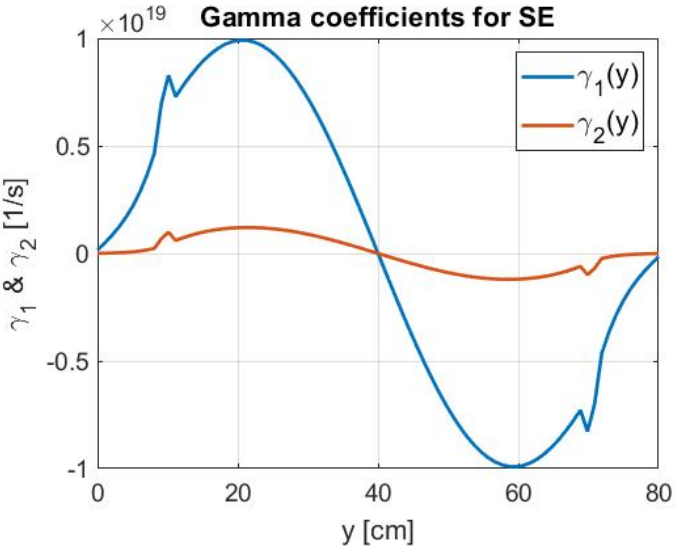


(a) $\alpha_1(y)$ and $\alpha_2(y)$ coefficients for RI1

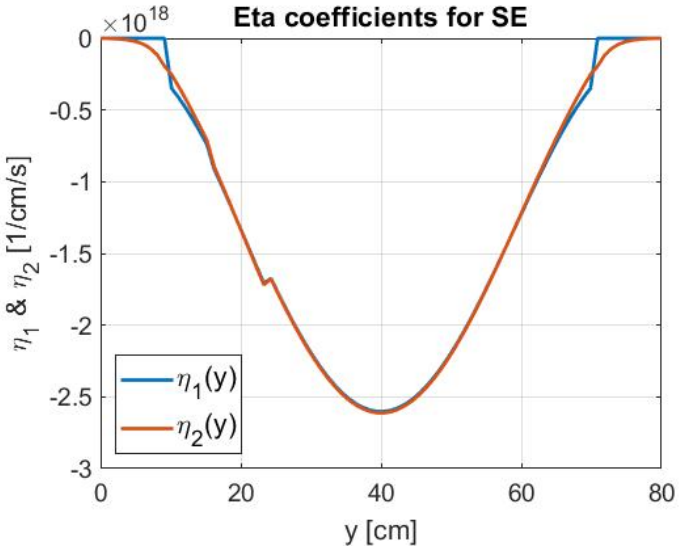


(b) $\delta_1(y)$ and $\delta_2(y)$ coefficients for RI1

Figure 4.13: α and δ coefficients



(a) $\gamma_1(y)$ and $\gamma_2(y)$ coefficients for RI1



(b) $\eta_1(y)$ and $\eta_2(y)$ coefficients for RI1

Figure 4.14: γ and η coefficients

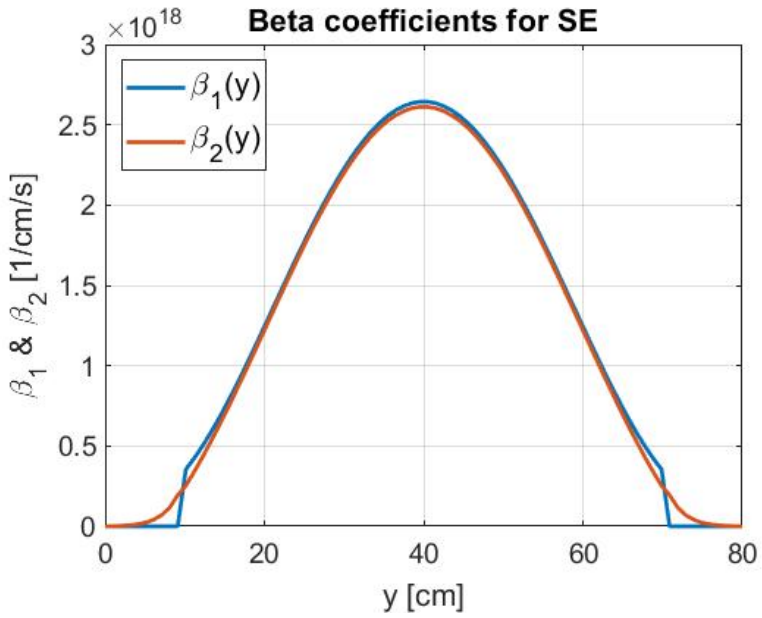
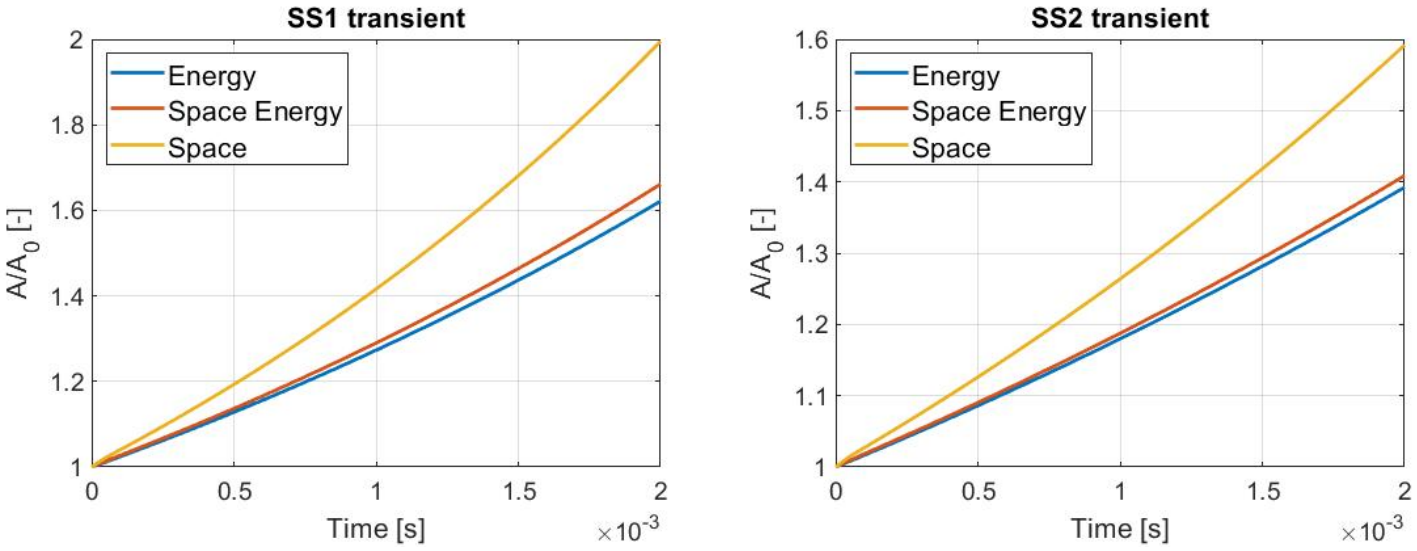


Figure 4.15: $\beta_1(y)$ and $\beta_2(y)$ coefficients for RI1

It is clear how the flux and the adjoints shape the coefficients: they are very similar to the flux profile. The alpha coefficients that are given by the product and integration of the flux and adjoint, times the inverse of velocity, assumes a bell shape that has symmetric profile. Even delta coefficients have this shape, that is interrupted by the breaks of lines for the diffusion length that changes between the reflector and the core region. The gamma are given by the derivative of the flux and the typical shape of sinus is traceable. The strange profile of the coefficient is given by the diffusion length that can be seen as a piecewise function. Indeed those peaks are located where there is the change of diffusion length. The eta coefficients assume negative values. In η_1 only the last term is subtracted and it is non zero much more times than the other addends; in η_2 the values are always subtracted. The fact that on the left side the line loses its sinuosity is given by the perturbation term. Finally the beta terms have almost a bell shape: β_1 accounts for the fission cross section and its perturbation, indeed it is zero after the core region; β_2 has a bell shape interrupted by the change of scattering cross section. So coefficients are very dependent on the shape of flux.

4.8 SS transients

The previous sections were limited just to the change of absorption cross section and fission cross section, neglecting what is happening on the other properties. With this transient it is decided to study the change of scattering cross section as it is done for *HE1* and *HE2*. It is remarkable that this does not imply any modification for the coefficients of equation of group one.



(a) Trasient SS1

(b) Transient SS2

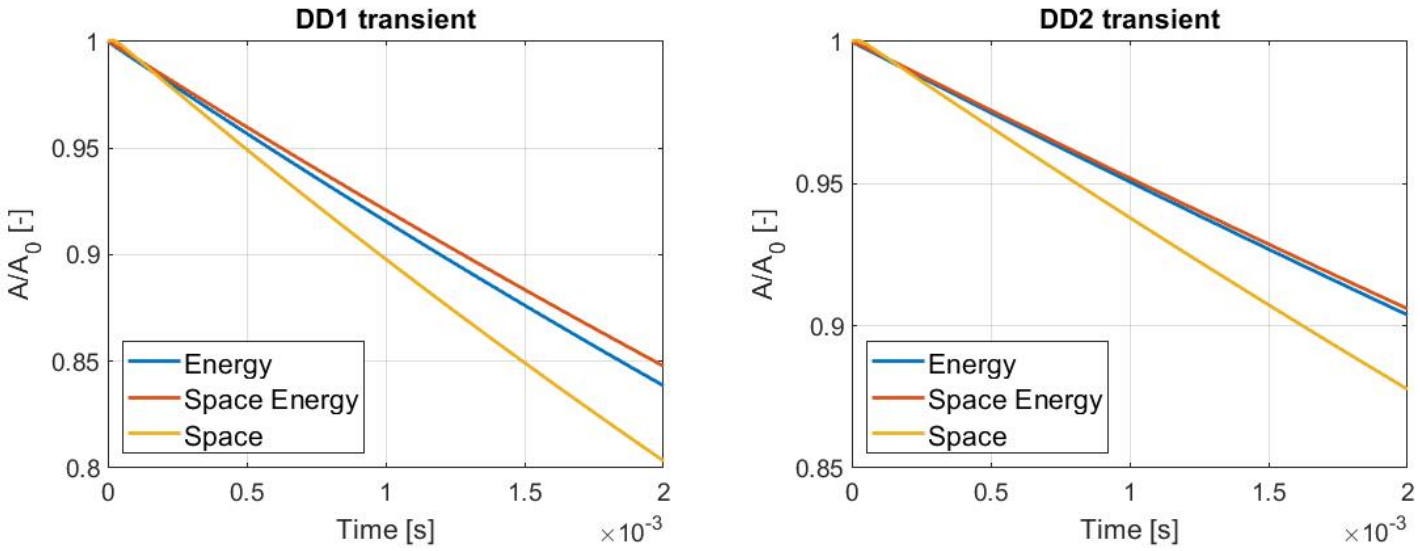
Figure 4.16: Transients for SS cases

The results show the increase of amplitude as it is expected. This is due to the increase of the thermal neutrons, the ones that are more incline to produce power. It can be seen that the space-energy and energy discretization are very similar while the space study leads to an over prediction of the power produced. The first two methods indeed study the coupling of the groups for all the transient: there is a direct updating of the behaviour of amplitudes. The distinction of two groups allows the possibility in each time step to study that a fast neutron makes scattering, it becomes thermal and it has more possibility to make fission leading to new fast neutrons. It is important to remember that a fast neutron has also the possibility to be absorbed. Conversely the simple study space records just the positive effect of the augmentation of scattering cross section. In some way it is neglected the fact that some neutrons are lost. So for this reason there is an over prediction of the results and it is believed, even if it is not tested, that a decrease of this cross section would lead to an underestimation. However the fact that the energy and space-energy study do not differ so much between themself leads to think that the discretization in space is not so important as the one on energy.

4.9 DD transients

This kind of transients is studied as an adjoint contribute to the work already present. The study done by Dulla and Nervo is focused only on the analysis of the perturbations of the cross sections and not on the length of diffusion. Moreover this will give another point of interpretation on the difference about

the methods. Differently from the last two sections, 4.5 and 4.8, here the perturbation occurs on a property that does not couple directly the fast and thermal neutrons.



(a) Trasient DD1 **(b)** Transient DD2

Figure 4.17: Transients for DD cases

The increase of diffusion length cause the increase of the path of neutrons in the system: this means that they have lower possibility to be absorbed as well as to make fission. The final result is the decrease of the power. This is clearly showed by the three different methods. As in previous analysis there is again the difference between the methods that appears. The energy and the space-energy discretization seems to overlap between themself, giving the idea that the remarkable thing is not the discretization on space but the one on energy. It is remarkable that the trend of space does not differ so much from the one of the other two meaning that even just space discretization can give an idea of what is occurring inside the reactor.

4.10 FA and DD transients

The previous analysis show the errors that can be committed using the same magnitude perturbations already used for the transients *HE*. This may lead to consider that the differences showed in sections 4.8 and 4.9 can be given by the magnitude of perturbation. For this reason they are repeated introducing and removing reactivity values that are comparable to the ones of transients *HE*. These are showed in the first table while in the second the relative change of properties:

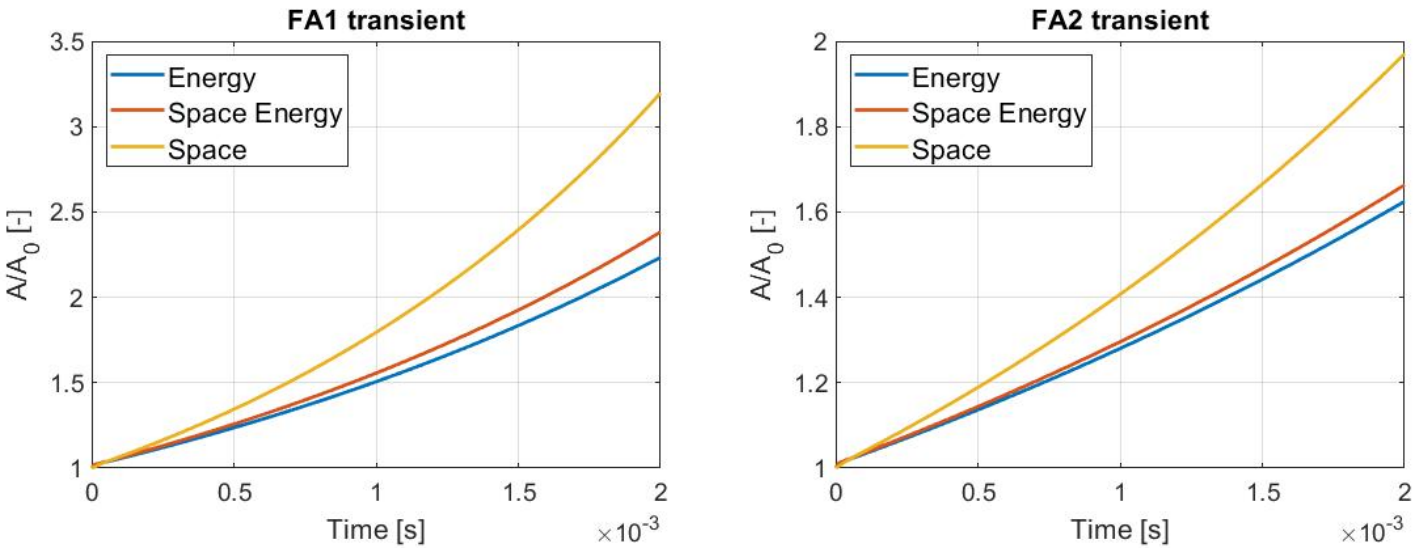
	First transient	Second transient
FA	1273	776
DD	1174	749

Table 4.9: Reactivities for FA1, FA2, DD3 and DD4 transients (data are in *pcm*)

	First transient	Second transient
FA	10.7	60
DD	45.9	800

Table 4.10: Relative change for FA1, FA2, DD3 and DD4 transients (data are in %)

The first one consider a pure augmentation of fission cross section. In this way it can be understood for an introduction of reactivity like the one on table how much the amplitude should increase.

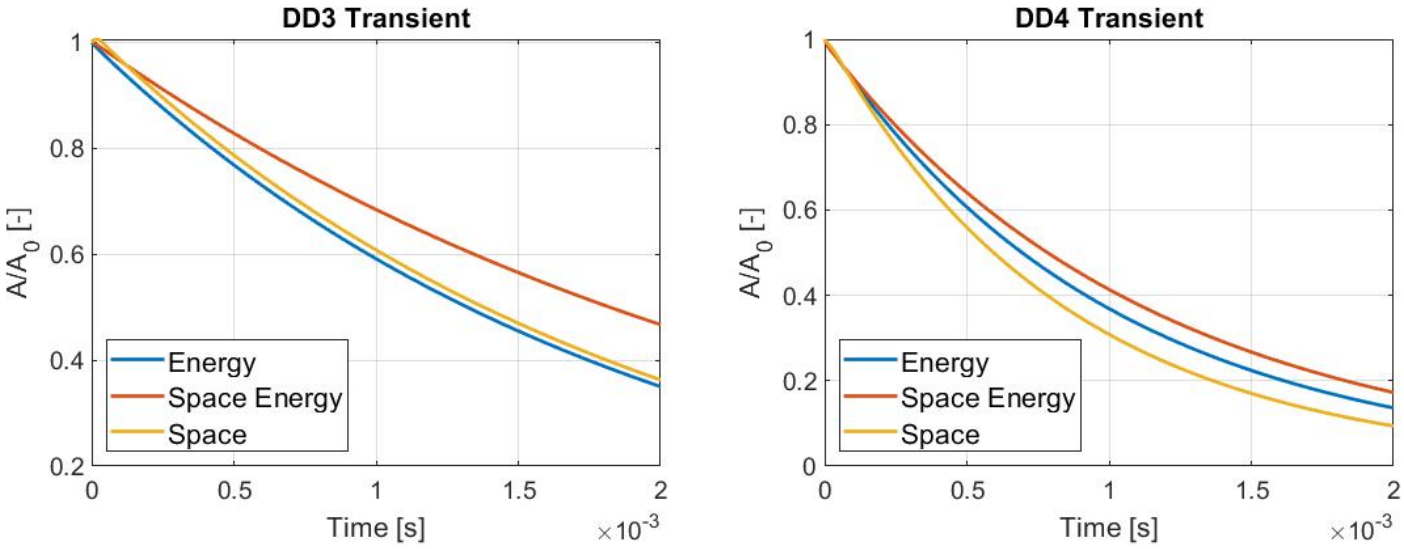


(a) Trasient FA1 **(b)** Transient FA2

Figure 4.18: Transients for FA cases

By the figures 4.18a and 4.18b it comes that the amplitudes increase. This was an expected behaviour since the higher fission cross section. Differently from the *RI* cases the modification is not done for a compensation of the negative reactivity and this is why such big values. The effects can be explained by considering what is already observable by the transients *SS*. The increase of fission cross section is seen as event that raises the production of power of the entire reactor. It is neglected that a neutron can be also removed, without considering the energy discretization.

Finally other transients that involve the increase of diffusion length are involved, in such a way to comport the same loss of reactivity.



(a) Transient DD3 **(b)** Transient DD4

Figure 4.19: Transients for DD3 and DD4 cases

Differently from the previous cases the trend do not differ so much. This behaviour can be explained by the fact that the increase of diffusion does not comport that neutrons are lost or produced but they are simply more diffused in space. For this reason even a space discretization is important but also enough to describe what is happening inside the reactor. Indeed it is remarked that in 4.19a the space and energy discretizations results are quite close.

The analysis of these transients allow also to make another consideration more related to physical behaviour. It is reported the errors committed between the discretization:

		$t = 50\mu s$	$t = 100\mu s$	$t = 150\mu s$	$t = 200\mu s$
HE1	E/SE	1.63	3.86	6.03	8.16
	S/SE	7.57	14.1	20.2	25.9
SS1	E/SE	0.745	1.3	1.85	2.4
	S/SE	4.96	9.78	14.8	20.1
FA1	E/SE	1.69	3.22	4.74	6.23
	S/SE	6.92	15.3	24.4	34.1
DD3	E/SE	7.21	13.6	19.5	25
	S/SE	4.98	11.1	16.9	22.3

Table 4.11: Errors for transients (data are in %)

Looking at the errors it appears that the errors for *FA1* are bigger than the one noted for *HE1* and *DD3* (at leat for space discretization). It is believed that a similar consideration can be extended also to transient *SS1* that has an insertion of reactivity that is 758 *pcm* so lower than the about 1200 *pcm* involved for the first kind of transient. So it is noted that the errors committed between the space discretization and the space-energy discretization are bigger when the properties perturbed involve the coupling between the groups. Regarding the energy and space energy it seems the contrary. Probably this is due to the different structure of equations. Space energy and energy discretization share the fact that they create a system of equations and changing fission or scattering cross section have effect directly on this coupling. Conversely the change of other properties do not have a direct impact on this characteristic and it is felt differently depending on the discretization. A particular focus should be done for the diffusion length. δD_g that is present in different ways between the two discretization and this explains why errors are bigger.

4.11 Insertion of delayed

In this section the presence of delayed is introduced. They were considered in the work made by Ravetto et al. but neglected by Dulla and Nervo. All the main previous transients are repeated and the results are showed. In the graphs even the amplitude for prompt cases are reported: they are the dashed lines.

4.11.1 Assumptions and model set

The transients are characterized by the presence of two new terms: the fraction of delayed and the decay constant. Even if there are a lot of families of delayed neutrons, that can be reduced just to six families, it is considered the presence of only one. For this reason the data chosen for the just listed presented properties are:

$$\lambda = 0.1 \ \frac{1}{s}$$

$$\beta = 0.0065$$

In particular the fraction of delayed is the one of uranium-235. Delayed neutrons are in relation with the prompt ones participating to the overall count when they are emitted. For the sake of simplicity it is assumed that all the delayed neutrons emitted are fast and belong to group one. Due to the model set of

OpenFOAM®, for each variable used (reference flux, amplitudes, etc.) an initial condition and boundary conditions are requested, so also for the concentration of neutrons. Even if it is not correct from a mathematical point of view it is done the hypothesis that the concentration of neutrons at boundaries is zero (see section 3.1). So the assumptions done are:

- there is only one family of delayed;
- all the neutrons generated by decay of fragments are fast;
- concentrations of neutrons at boundaries is null;

The system studied are the same of before, so the properties are not reported again. Despite this the equations are rewritten:

$$
\begin{cases}
\Big\langle w_1(\vec{r})\Big|\dfrac{1}{v_1}\varphi_{1,0}(\vec{r})\Big\rangle_{\vec{r}-y}\dfrac{\partial A_1(y,t)}{\partial t} = \Big\langle w_1(\vec{r})\Big|\Big(D_1(\vec{r})+\delta D_1(\vec{r})\Big)\varphi_{1,0}(\vec{r})\Big\rangle_{\vec{r}-y}\dfrac{\partial^2 A_1(y,t)}{\partial y^2} \\
\qquad\qquad + \Big\langle w_1(\vec{r})\Big|2\Big(D_1(\vec{r})+\delta D_1(\vec{r})\Big)\dfrac{\partial\varphi_{1,0}(\vec{r})}{\partial y}\Big\rangle_{\vec{r}-y}\dfrac{\partial A_1(y,t)}{\partial y} \\
\qquad\qquad + \Big\langle w_1(\vec{r})\Big|\Big((1-\beta)\delta\nu\Sigma_{f,1}(\vec{r})-\delta\Sigma_{r,1}(\vec{r})\Big)\varphi_{1,0}(\vec{r}) \\
\qquad\qquad - \beta\nu\Sigma_{f,1}(\vec{r})\varphi_{1,0}(\vec{r})-\nu\Sigma_{f,2}(\vec{r})\varphi_{2,0}(\vec{r})\Big\rangle_{\vec{r}-y}A_1(y,t) \\
\qquad\qquad + \Big\langle w_1(\vec{r})\Big|(1-\beta)\Big(\nu\Sigma_{f,2}(\vec{r})+\delta\nu\Sigma_{f,2}(\vec{r})\Big)\varphi_{2,0}(\vec{r})\Big\rangle_{\vec{r}-y}A_2(y,t) \\
\qquad\qquad + \lambda\Big\langle w_1(\vec{r})\Big|C(\vec{r},t)\Big\rangle_{\vec{r}-y} \\
\Big\langle w_2(\vec{r})\Big|\dfrac{1}{v_2}\varphi_{2,0}(\vec{r})\Big\rangle_{\vec{r}-y}\dfrac{\partial A_2(y,t)}{\partial t} = \Big\langle w_2(\vec{r})\Big|\Big(D_2(\vec{r})+\delta D_2(\vec{r})\Big)\varphi_{2,0}(\vec{r})\Big\rangle_{\vec{r}-y}\dfrac{\partial^2 A_2(y,t)}{\partial y^2} \\
\qquad\qquad + \Big\langle w_2(\vec{r})\Big|2\Big(D_2(\vec{r})+\delta D_2(\vec{r})\Big)\dfrac{\partial\varphi_2(\vec{r})}{\partial y}\Big\rangle_{\vec{r}-y}\dfrac{\partial A_2(y,t)}{\partial y} \\
\qquad\qquad + \Big\langle w_2(\vec{r})\Big|-\Sigma_{1\to2}(\vec{r})\varphi_{1,0}(\vec{r})-\delta\Sigma_{r,2}(\vec{r})\varphi_{2,0}(\vec{r})\Big\rangle_{\vec{r}-y}A_2(y,t) \\
\qquad\qquad + \Big\langle w_2(\vec{r})\Big|\Big(\Sigma_{1\to2}(\vec{r})+\delta\Sigma_{1\to2}(\vec{r})\Big)\varphi_{1,0}(\vec{r})\Big\rangle_{\vec{r}-y}A_1(y,t) \\
\Big\langle w_1(\vec{r})\Big|\dfrac{\partial C(\vec{r},t)}{\partial t}\Big\rangle_{\vec{r}-y} = -\lambda\Big\langle w_1(\vec{r})\Big|C(\vec{r},t)\Big\rangle_{\vec{r}-y} \\
\qquad\qquad + \Big\langle w_1(\vec{r})\Big|\beta\Big(\nu\Sigma_{f,1}(\vec{r})+\delta\nu\Sigma_{f,1}(\vec{r})\Big)\varphi_{1,0}(\vec{r})\Big\rangle_{\vec{r}-y}A_1(y,t) \\
\qquad\qquad + \Big\langle w_1(\vec{r})\Big|\beta\Big(\nu\Sigma_{f,2}(\vec{r})+\delta\nu\Sigma_{f,2}(\vec{r})\Big)\varphi_{2,0}(\vec{r})\Big\rangle_{\vec{r}-y}A_2(y,t)
\end{cases}
\tag{4.8}\tag{4.9}\tag{4.10}
$$

For energy:

$$
\begin{cases}
\Big\langle w_1(\vec{r})\Big|\dfrac{1}{v_1}\varphi_{0,1}(\vec{r})\Big\rangle_{\vec{r}}\dfrac{dA_1(t)}{dt} = \Big\langle w_1(\vec{r})\Big|\Big((1-\beta)\delta\nu\Sigma_{f,1}(\vec{r}) - \delta\Sigma_{r,1}(\vec{r})\Big)\varphi_{1,0}(\vec{r}) \\
\qquad\qquad - \beta\nu\Sigma_{f,1}(\vec{r})\varphi_{1,0}(\vec{r}) - \nu\Sigma_{f,2}(\vec{r})\varphi_{2,0}(\vec{r})\Big\rangle A_1(t) \\
\qquad\qquad + \Big\langle w_1(\vec{r})\Big|(1-\beta)\Big(\nu\Sigma_{f,2}(\vec{r}) + \delta\nu\Sigma_{f,2}(\vec{r})\Big)\varphi_{2,0}(\vec{r})\Big\rangle A_2(t) \\
\qquad\qquad + \lambda\Big\langle w_1(\vec{r})\Big|C(\vec{r})\Big\rangle
\end{cases} \tag{4.11}
$$

$$
\begin{cases}
\Big\langle w_2(\vec{r})\Big|\dfrac{1}{v_2}\varphi_{2,0}(\vec{r})\Big\rangle\dfrac{dA_2(t)}{dt} = \Big\langle w_2(\vec{r})\Big| - \Sigma_{1\to 2}(\vec{r})\varphi_{1,0}(\vec{r}) - \delta\Sigma_{r,2}(\vec{r})\varphi_{2,0}(\vec{r})\Big\rangle A_2(t) \\
\qquad\qquad + \Big\langle w_2(\vec{r})\Big|\Big(\Sigma_{1\to 2}(\vec{r}) + \delta\Sigma_{1\to 2}(\vec{r})\Big)\varphi_{1,0}(\vec{r})\Big\rangle_{\vec{r}} A_1(t)
\end{cases} \tag{4.12}
$$

$$
\begin{cases}
\Big\langle w_1(\vec{r})\Big|\dfrac{dC(\vec{r},t)}{dt}\Big\rangle_{\vec{r}} = -\lambda\Big\langle w_1(\vec{r})\Big|C(\vec{r},t)\Big\rangle_{\vec{r}} \\
\qquad\qquad + \Big\langle w_1(\vec{r})\Big|\beta\Big(\nu\Sigma_{f,1}(\vec{r}) + \delta\nu\Sigma_{f,1}(\vec{r})\Big)\varphi_{1,0}(\vec{r})\Big\rangle A_1(y,t) \\
\qquad\qquad + \Big\langle w_1(\vec{r})\Big|\beta\Big(\nu\Sigma_{f,2}(\vec{r}) + \delta\nu\Sigma_{f,2}(\vec{r})\Big)\varphi_{2,0}(\vec{r})\Big\rangle A_2(y,t)
\end{cases} \tag{4.13}
$$

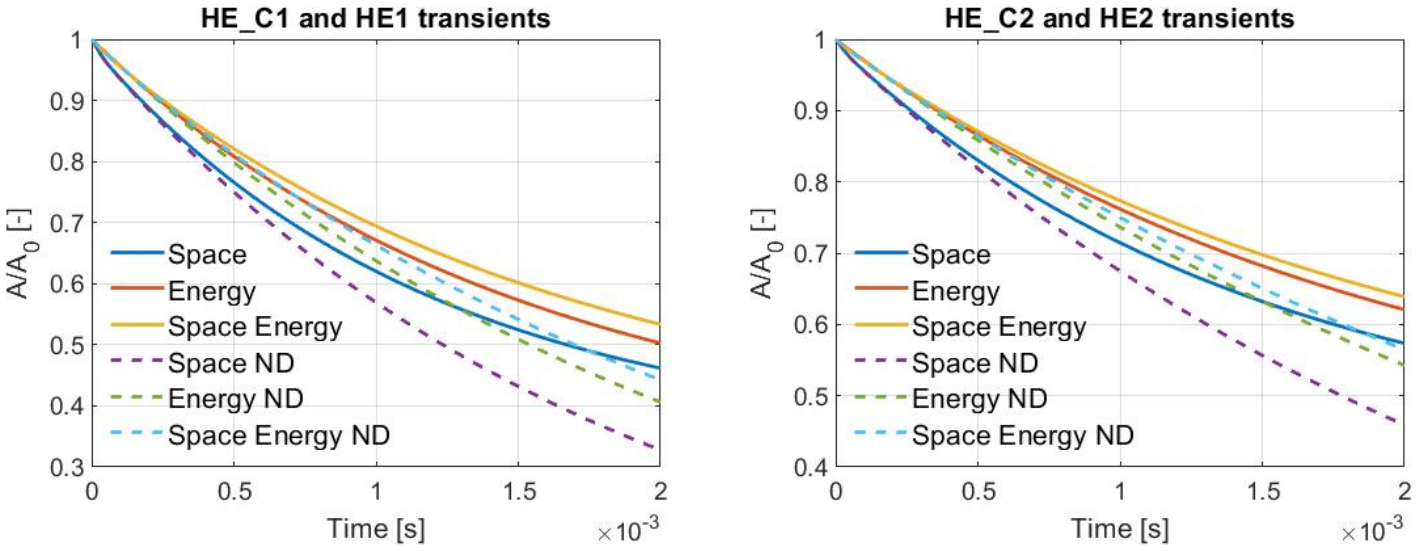
For space:

$$
\begin{cases}
\left(\left\langle w_1(\vec{r})\middle|\frac{1}{v_1}\varphi_{1,0}(\vec{r})\right\rangle_{\vec{r}-y}+\left\langle w_2(\vec{r})\middle|\frac{1}{v_2}\varphi_{2,0}(\vec{r})\right\rangle_{\vec{r}-y}\right)\frac{\partial A(y,t)}{\partial t} \\
=\left(\left\langle w_1(\vec{r})\middle|\Big(D_1(\vec{r})+\delta D_1(\vec{r})\Big)\varphi_{1,0}(\vec{r})\right\rangle_{\vec{r}-y}\right. \\
\left.+\left\langle w_2(\vec{r})\middle|\Big(D_2(\vec{r})+\delta D_2(\vec{r})\Big)\varphi_{2,0}(\vec{r})\right\rangle_{\vec{r}-y}\right)\frac{\partial^2 A(y,t)}{\partial y^2} \\
+\left(\left\langle w_1(\vec{r})\middle|2\Big(D_1(\vec{r})+\delta D_1(\vec{r})\Big)\frac{\partial\varphi_{1,0}(\vec{r})}{\partial y}\right\rangle_{\vec{r}-y}\right. \\
\left.+\left\langle w_2(\vec{r})\middle|2\Big(D_2(\vec{r})+\delta D_2(\vec{r})\Big)\frac{\partial\varphi_{2,0}(\vec{r})}{\partial y}\right\rangle_{\vec{r}-y}\right)\frac{\partial A(y,t)}{\partial y}
\end{cases}
\tag{4.14}
$$

$$
\begin{cases}
\left(\left\langle w_1(\vec{r})\middle|\delta D_1(\vec{r})\nabla^2\varphi_{1,0}(\vec{r})+\Big(\delta\nu\Sigma_{f,1}(\vec{r})-\delta\Sigma_{r,1}(\vec{r})\Big)\varphi_{1,0}(\vec{r})+\delta\nu\Sigma_{f,2}(\vec{r})\varphi_{2,0}(\vec{r})\right\rangle_{\vec{r}-y}\right. \\
\left.+\left\langle w_2(\vec{r})\middle|\delta D_2(\vec{r})\nabla^2\varphi_{2,0}(\vec{r})+\delta\Sigma_{1\to2}(\vec{r})\varphi_{1,0}(\vec{r})-\delta\Sigma_{r,2}(\vec{r})\varphi_{2,0}(\vec{r})\right\rangle_{\vec{r}-y}\right)A(y,t) \\
\left\langle w_1(\vec{r})\middle|\frac{\partial C(\vec{r},t)}{\partial t}\right\rangle_{\vec{r}-y}=-\lambda\left\langle w_1(\vec{r})\middle|C(\vec{r},t)\right\rangle_{\vec{r}-y} \\
\qquad\qquad+\left(\left\langle w_1(\vec{r})\middle|\beta\Big(\nu\Sigma_{f,1}(\vec{r})+\delta\nu\Sigma_{f,1}(\vec{r},t)\Big)\varphi_{1,0}(\vec{r})\right\rangle_{\vec{r}-y}\right. \\
\qquad\qquad\left.+\left\langle w_1(\vec{r})\middle|\beta\Big(\nu\Sigma_{f,2}(\vec{r})+\delta\nu\Sigma_{f,2}(\vec{r},t)\Big)\varphi_{2,0}(\vec{r})\right\rangle_{\vec{r}-y}\right)A(y,t)
\end{cases}
\tag{4.15}
$$

4.11.2 HE

This kind of transients wants the increase of absorption cross section and this leads to a decrease of amplitude.



(a) Trasient HE_C1 **(b)** Transient HE_C2

Figure 4.20: Transients for HE_C cases

The insertion of delayed create a clear different trend of the amplitudes: actually the power does not drop as before. This was an expected thing. Such behaviour can be forecasted due to insertion of delayed. They act as sort of source, indeed the above equations are basically equal to the ones at section 4.1 but there is the adjoint term of the delayed. So to normal contribution of the prompt neutrons (even if a small fraction is held by fraction of delayed) there is a small one of decay of the fragments. So this explains why the amplitude tends to be higher. The insertion of delayed moreover is a little step towards the real representation of the physics: delayed neutrons are always present and they can not be neglected, otherwise big errors can be committed. Maybe this explains also why the errors committed are lower as it is shown in the tables 4.12 and 4.13. Moreover the insertion of delayed creates a sort of equilibrium: as the concentration of fragments generate fast neutrons ready for fission, a part of this goes to the fraction of delayed.

HE_C1	$t = 50\mu s$	$t = 100\mu s$	$t = 150\mu s$	$t = 200\mu s$
E/SE	1.63	3.86	6.03	8.16
S/SE	7.57	14.1	20.2	25.9
E_C/SE_C	1.5	3.3	4.72	5.77
S_C/SE_C	6.67	10.8	12.8	13.5

Table 4.12: Errors for HE_C1

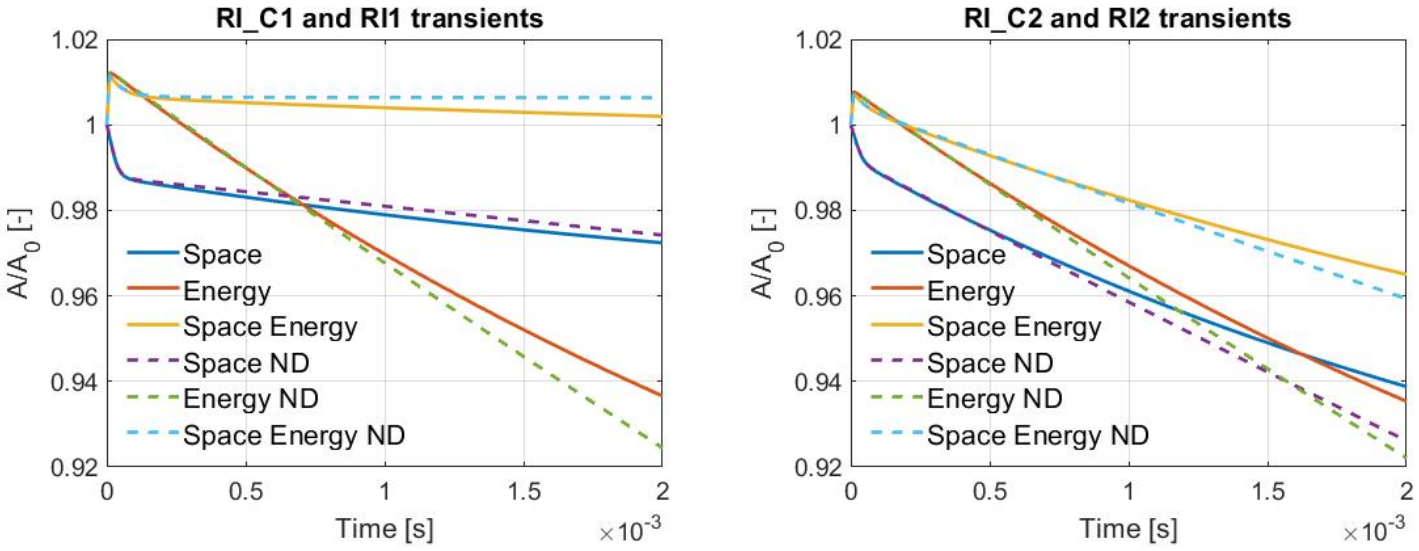
HE_C2	$t = 50\mu s$	$t = 100\mu s$	$t = 150\mu s$	$t = 200\mu s$
E/SE	0.703	1.78	2.84	3.89
S/SE	5.29	9.99	14.2	18.7
E_C/SE_C	0.646	1.53	2.26	2.84
S_C/SE_C	4.68	7.63	9.41	10.2

Table 4.13: Errors for HE_C2

As it can be seen the errors tend to be lower for cases where the delayed are considered. In particular the difference between the two types of error is evident when time passes.

4.11.3 RI_C transients

The two transients *RI1* and *RI2* are repeated with the inclusion of delayed neutrons.



(a) Trasient RI_C1 **(b)** Transient RI_C2

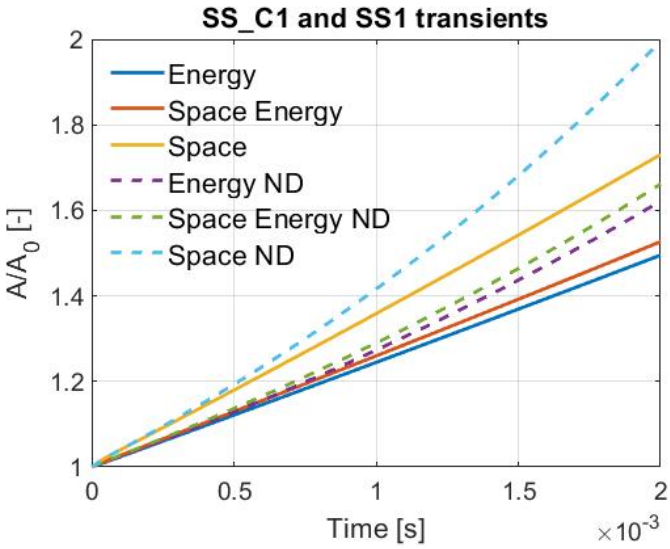
Figure 4.21: Transients for RI_C cases

The results are similar to the prompt case. This was expected due to the data decided for the description of these transients with delayed. The presence of delayed gives higher or lower results of amplitude depending on what occurred with prompt generation. In case of energy discretization, there is not a good representation of compensation: the power tends to decrease. The presence of delayed slows this drop acting as a sort of source. This behaviour was already observed in the previous transients. Conversely for the space-energy discretization, the power arranges on a value close to one but the remarkable thing is that

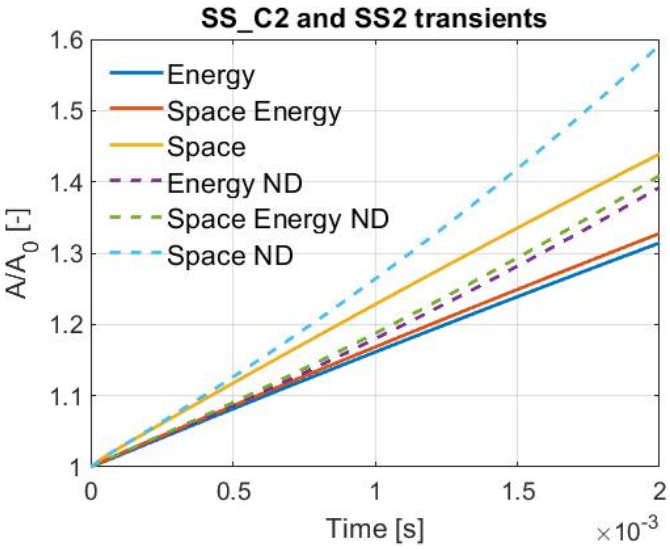
it decreases. Observing the trends, it appears that the delayed have a counter effect on power, slowing the increase and moreover changing its slope. Delayed insertion implies that a small part of the neutrons are not temporarily available for the fission. The consequence is the lower production of power. In this case, the removal of these neutrons does not only cause a lower power production but addresses it to decline. This may be explained by the fact that neutrons are emitted and not immediately available: in this fraction of time the prompt ones are not enough to balance the removal for absorption cross section increase and the delayed ones are emitted with small rate time. The overall effect is the decrease of power. Regarding the space analysis of transient *RI1* it can be seen that the delayed insertion cause an effect that was not expected. Since amplitude tends to decrease, the delayed insertion should be a balance, being a sort of source. This is not the case, since the power generated considering delayed neutrons is lower than the one only by prompt. This means that even the space discretization generation showed in the present graph and the one in figure 4.6a, reach the equilibrium between the created neutrons and the ones lost. The fact that the amplitude values are close to 0.98 it can be explained by the less information that this discretization method has. Regarding the transients *RI2* that show that the method is not good to describe this kind of compensation, it can be seen that the amplitude considering delayed is always higher, that is an expected behaviour.

4.11.4 SS_C transient

Even the transient where there is a perturbation of scattering cross section is repeated.



(a) Transient SS_C1



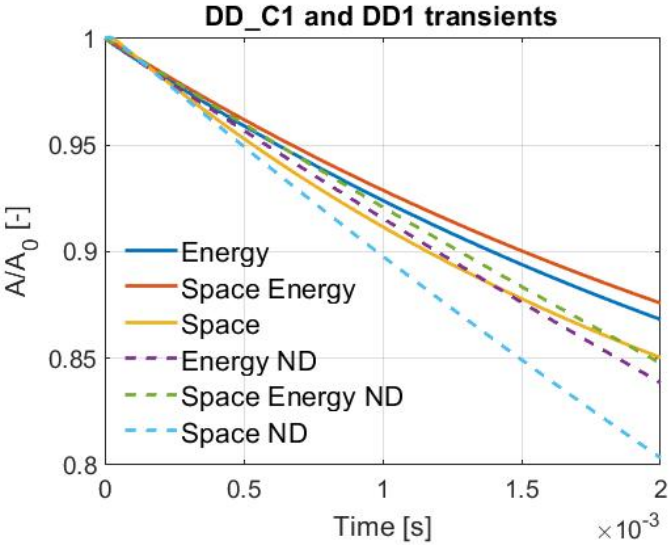
(b) Transient SS_C2

Figure 4.22: Transients for SS_C cases

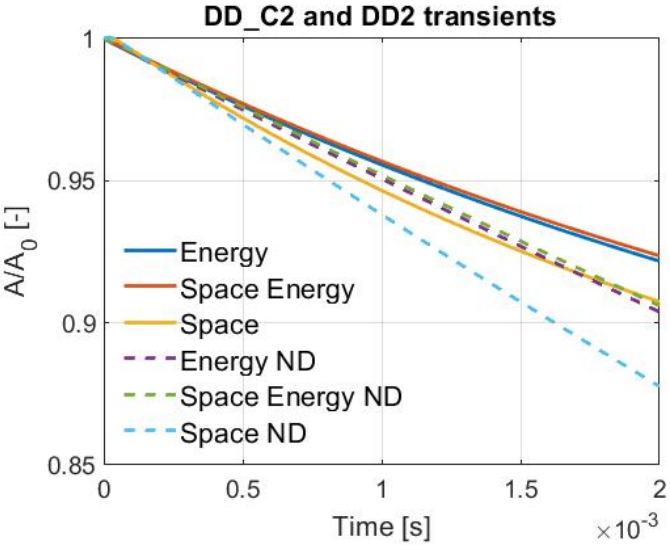
The figures show what is already said for the transients *RI*. The consideration of delayed neutrons makes slower power increase. It is remarkable that even with the insertion of delayed the discrepancies between the discretization do not differ so much. Again the space discretization overestimates the real power generation.

4.11.5 DD_C transients

The trend given by the increase of diffusion length is reported again.



(a) Trasient DD_C1



(b) Transient DD_C2

Figure 4.23: Transients for DD_C cases

The lines respect the expectations. The power caused considering the delayed neutrons is higher since they act as sources. In these two cases, it appears even clearer that in the case of delayed neutrons the errors between the methods are lower. Indeed the increase of diffusion length create a wider diffusion of neutrons that do not make fission neither causes removal of them.

4.11.6 HE_C_04 transients

The presence of delayed neutrons implies also that they should reach an equilibrium. If the neutrons are destined to be removed, a part of them do not disappear immediately due to presence of delayed. They can slow a lot this drop, holding the neutrons.

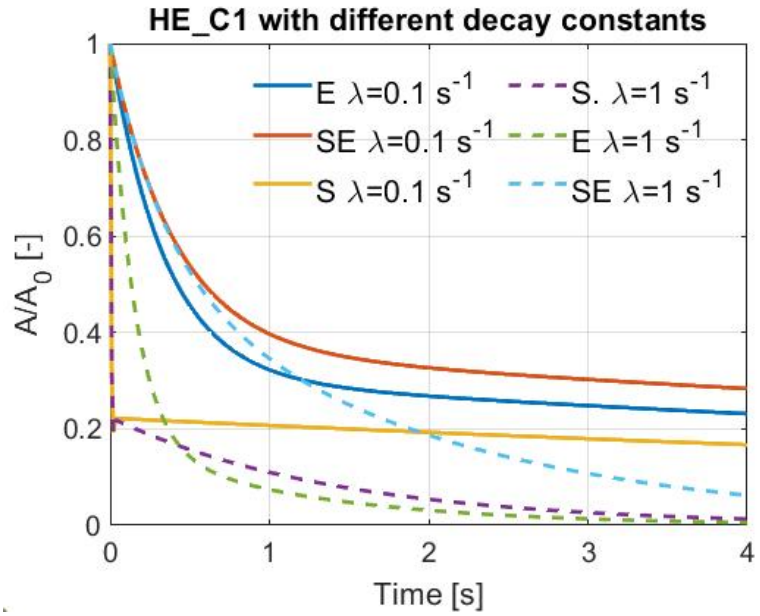
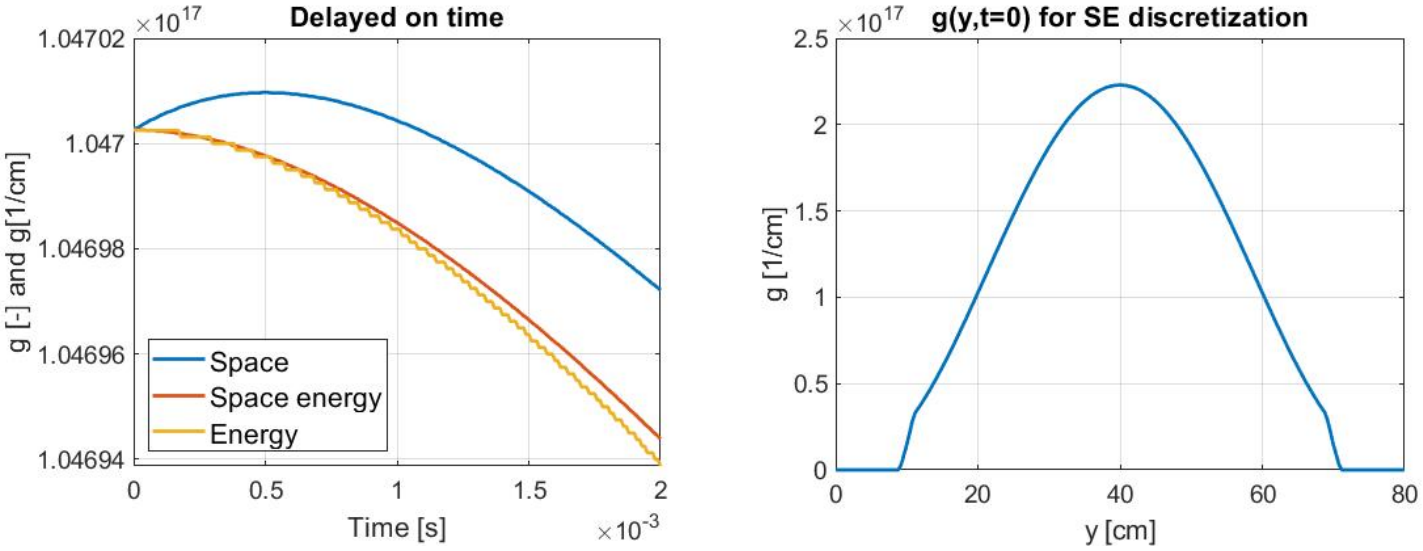


Figure 4.24: Trasient HE_C1_04

As it can be seen by the graphs a lower decay constant leads easily to an equilibrium. After a fast drop of the amplitude, caused by the more neutrons absorption there is a sort of stabilization. This can be considered a balance between the prompt neutrons that suffer the increase of removal cross section and the delayed ones that are preserved by the removal. On the other side in case of an higher decay constant (the value chosen does not refer to any particular physical case) the drop continues. The neutrons have a smaller delay so they are available earlier for fission as well as for removal and since there is an increase of absorption cross section, it is more probable that neutrons disappear. The fact that some amplitudes drop dramatically is believed to be a numerical effect. However the interesting thing is the asymptotic behaviour

4.11.7 Delay distribution

It is reported the delay distribution and trend on time, for case *HE_C1*.



(a) Delayed on time

(b) Delayed on space

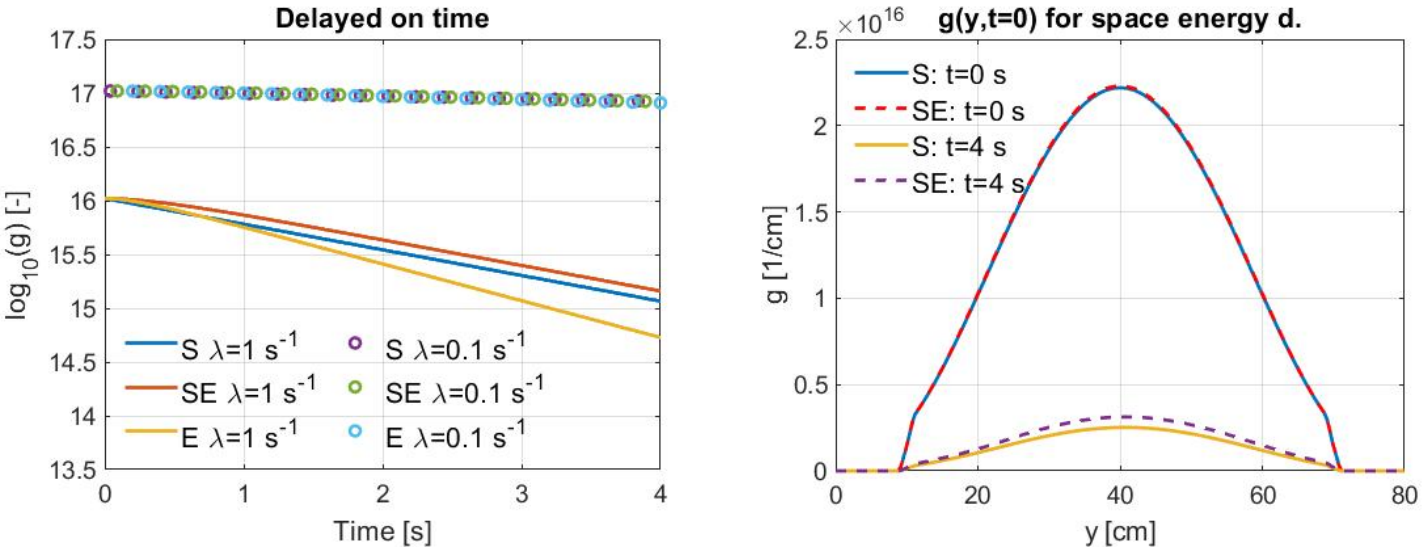
Figure 4.25: Delayed on time and space

The choice to report only one trend can be even explained by the plots. Due to the magnitudes, much more bigger than the one of the amplitudes is very difficult to see the differences between the lines. These differences can be achieved looking at trends on time and restricting a lot the the range of visibility. Even if the lines are distinguishable the errors committed are very low, on the order of 10^{-5}. This means that the discretization method chosen is not very important in case of delayed neutrons study. However it is noted that for space discetization there is an increase and then a decrease of concentration. Maybe this is due to the different coefficients involved, since $\zeta(y) > (\zeta_1(y) + \zeta_2(y))$. Moreover it can be argued that the time of study and also the decay constant that for case *HE_C1* is $\lambda = 0.1\ s^{-1}$ are very small. This can explains also the strange trend of the energy amplitude that seems to be constant for certain period of time. Despite this the time range studied is always between 0 and 0.002 s and the value of 0.1 s^{-1} is the closest to physical ones.

Regarding the shape on space of the delayed this was expected. Looking at formulas for $g(y,t)$ this is function of flux and fission cross section. The first one, due to symmetry of the system has shape of a bell; the fission cross section is zero out of the core. This is the reason why $g(y,t)$ is zero towards the edges. The change of slope at the extremes of the bell can be explained by graphic reason since there are close points with values with orders of magnitude of difference.

To overcome problems related to short time study it is observed also the

trend of delay concentration in a larger time period and with different decay constants.



(a) Delayed on time

(b) Delayed on space

Figure 4.26: Delayed on time and space

Studying the transients with different decay constants leads obviously to different results. Firstly it has to be noted that between the initial values there is the difference of one unit, so of one order of magnitude (the graph is in logarithmic scale). The lower value means that the neutrons are emitted less frequently, so it decreases slowly. This is also what can be understood by the plots: the trends with $\lambda = 0.1\ s^{-1}$ remains always on a value around nineteen. On the other side the concentration for $\lambda = 1\ s^{-1}$ decrease quite rapidly. Looking at the different discretizations, it appears that differently from all other cases, the space is closer to the line of space-energy discretization. This can be explained by the difference types of equations involved. The energy discretization returns a system of ordinary differential equations while in the other cases there is a system of partial differential equations. These last two are much more similar between themself and even the structure that describe the delay concentration is more similar in these two last cases. These effects of different decay constants have impacts also on the distribution of concentration, as it can be seen by the figure 4.26b.

Chapter 5

One group case and 3D study

5.1 Mono-energetic case

In order to have a full description it is done a mono-energetic case, considering the same system of the previous chapter. To make a good comparison, the data are changed in order to find a criticality constant equal to one. The properties are the one listed in table 5.1 and they are not the one of a particular material. They give $k = 1.00002$ that can be considered approximately equal to the reference value.

One group data	D [cm]	Σ_a [cm^{-1}]	$\nu\Sigma_f$ [cm^{-1}]
Core	1.0	0.03	0.03562
Reflector	0.8	0.03	-

Table 5.1: One group data.

Observing the system and knowing the boundary conditions from the 3.8 it appears clear that the results has to be symmetric. Such behaviour is obtained as it seems in figure 5.1. In the image it is showed the distribution evaluated by the FVM so the surface is not smooth but pixelated. They are the cells that discretize the surface. This is showed contrary to the the images 4.3 in order to exhibit the real representation. As it can be seen the minimum value is not zero but higher due to the method used for computation. It is remarked that the flux is also the adjoint and values are in $\frac{neutrons}{cm \cdots s}$.

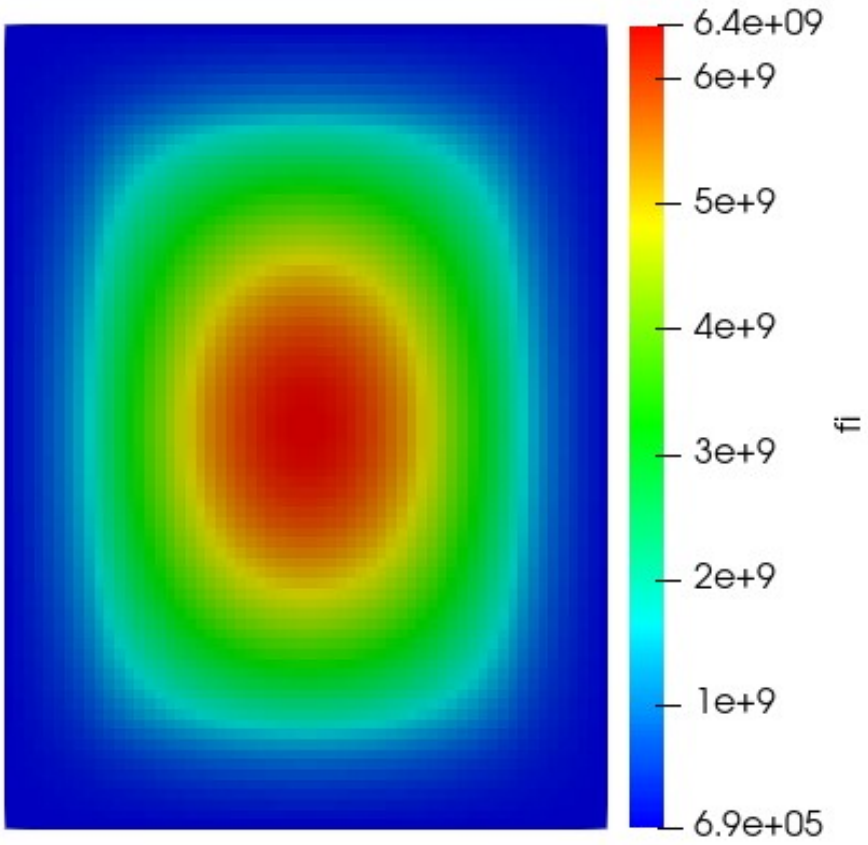


Figure 5.1: Flux for one group case

The number of cells are 4800 in line for what is written above and the assumptions are:

- study is in one group;
- the system is a two dimensional system;
- there is not a source;
- there is only one family of precursors;

With these the equations form can be simplified and they become (all the possible perturbations are considered):

$$\left\langle w(\vec{r})\middle|\frac{1}{v}\varphi_0(\vec{r})\right\rangle_{\vec{r}-y}\frac{\partial A(y,t)}{\partial t}=\left\langle w(\vec{r})\middle|\Big(D(\vec{r})+\delta D(\vec{r})\Big)\varphi_0(\vec{r})\right\rangle_{\vec{r}-y}\frac{\partial^2 A(y,t)}{\partial y^2}$$
$$+\left\langle w(\vec{r})\middle|2\Big(D(\vec{r})+\delta D(\vec{r})\Big)\frac{\partial\varphi_0(\vec{r})}{\partial y}\right\rangle_{\vec{r}-y}\frac{\partial A(y,t)}{\partial y}$$
$$+\left\langle w(\vec{r})\middle|\delta D(\vec{r})\nabla^2\varphi_0(\vec{r})+\Big(\delta\nu\Sigma_f(\vec{r})-\delta\Sigma_a(\vec{r})\Big)\varphi_0(\vec{r})\right\rangle_{\vec{r}-y}A(y,t) \tag{5.1}$$

Then it is showed the equation with presence of delayed.

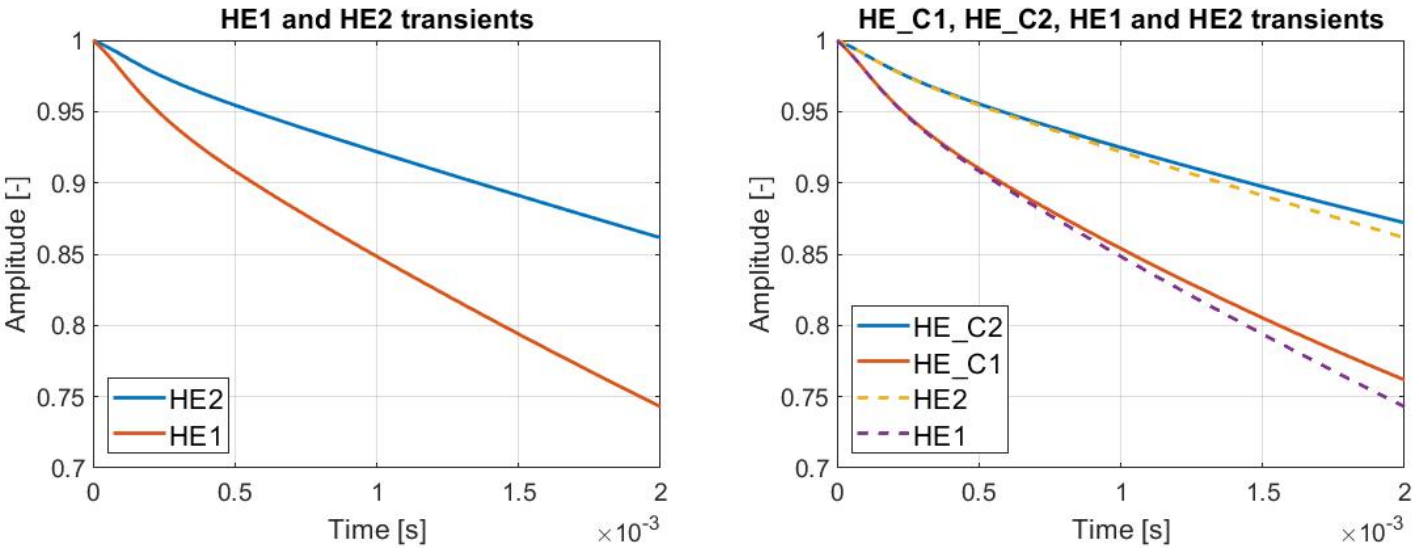
$$\begin{cases} \left\langle w(\vec{r})\middle|\frac{1}{v}\varphi_0(\vec{r})\right\rangle_{\vec{r}-y}\frac{\partial A(y,t)}{\partial t} = \left\langle w(\vec{r})\middle|\Big(D(\vec{r})+\delta D(\vec{r})\Big)\varphi_0(\vec{r})\right\rangle_{\vec{r}-y}\frac{\partial^2 A(y,t)}{\partial y^2} \\ \qquad + \left\langle w(\vec{r})\middle|2\Big(D(\vec{r})+\delta D(\vec{r})\Big)\frac{\partial\varphi_0(\vec{r})}{\partial y}\right\rangle_{\vec{r}-y}\frac{\partial A(y,t)}{\partial y} \\ \qquad + \left\langle w(\vec{r})\middle|\delta D(\vec{r})\nabla^2\varphi_0(\vec{r}) + \Big(\delta\nu\Sigma_f(\vec{r}) - \delta\Sigma_a(\vec{r})\Big)\varphi_0(\vec{r})\right\rangle_{\vec{r}-y} A(y,t) \end{cases} \tag{5.3}$$

$$\begin{cases} \left\langle w(\vec{r})\middle|\frac{\partial C(\vec{r},t)}{\partial t}\right\rangle_{\vec{r}-y} = -\lambda\left\langle w(\vec{r})\middle|C(\vec{r},t)\right\rangle_{\vec{r}-y} \\ \qquad + \left\langle w(\vec{r})\middle|\beta(\nu\Sigma_f(\vec{r}) + \delta\nu\Sigma_f(\vec{r}))\varphi(\vec{r})\right\rangle_{\vec{r}-y} A(y,t) \end{cases} \tag{5.4}$$

As it can be seen some terms disappear: there is not the source and properties can be seen as piece-wise functions, extracting them from the derivatives, studying just one of them.

5.1.1 HE transients

The first transients studied concern the augmentation of the absorption cross section (data are in table 4.2). Clearly it is expected that amplitude decreases.



(a) HE transients

(b) HE transients with delayed

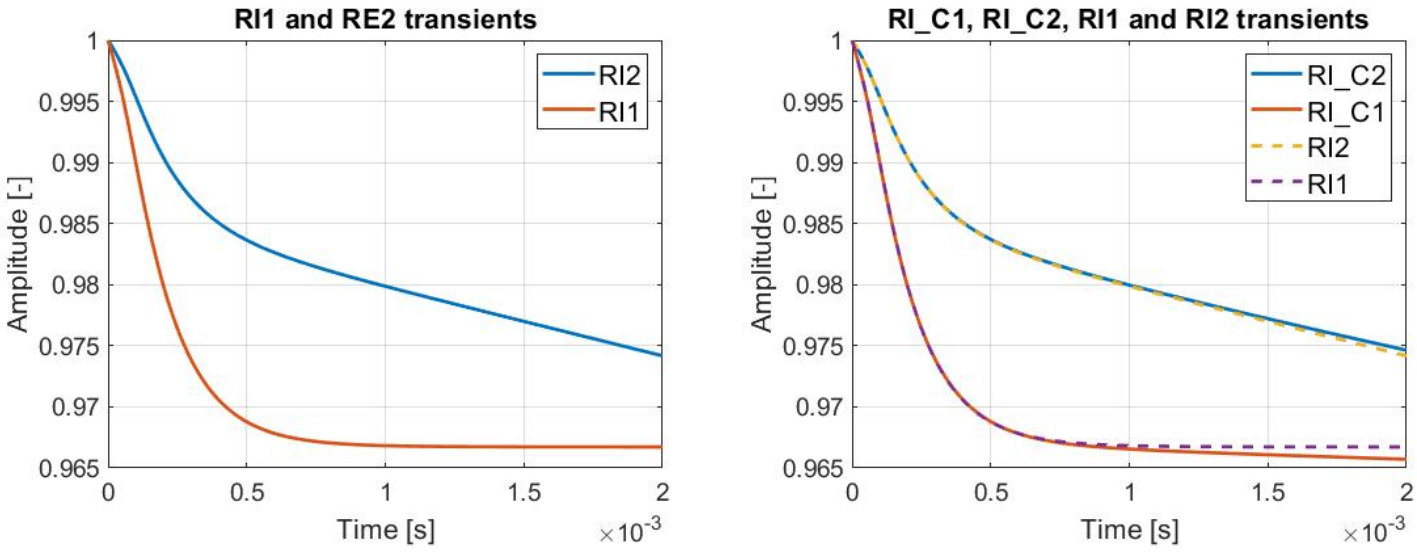
Figure 5.2: Transients for HE

It appears that the increase of absorption cross section leads to decrease of

amplitude. The other conclusion that is extracted is the importance of the extension of the perturbation, that was already observed previously. The most impactive change is the one by the transient *HE1* that is due to an extended change of the absorption cross section. The presence of delayed, satisfy the expectation because they slow the decline of the amplitude. Indeed the delay presence act as a way of source for neutrons.

5.1.2 RI transients

The RI transients provide the reinsertion of reactivity that is removed by the absorption cross section rise. Theoretically speaking this should compensate the loses, setting up the power to the value of one.



(a) RI transients **(b)** RI_C transients

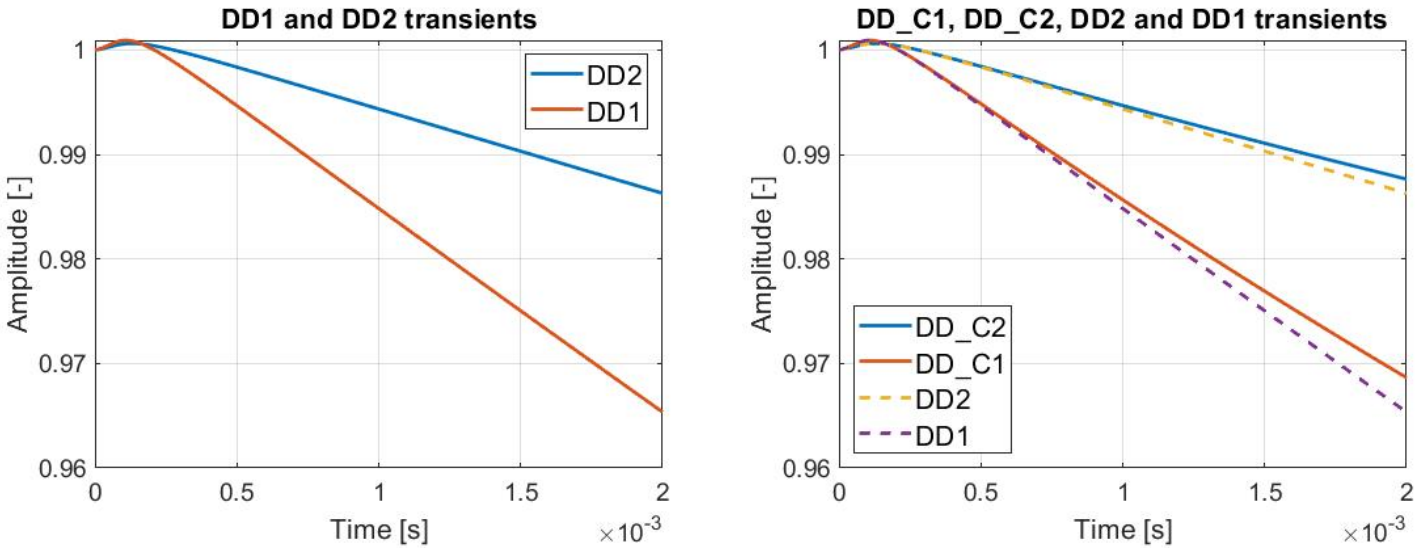
Figure 5.3: RI transients for space

As it appears in the figures 5.3 there is the compensation of the amplitude. The increase of fission cross section lead to a bigger production of neutrons that make fission so more power is generated. The effects are clearly visible by the enhance of the amplitude. However as it is written the theoretical value reached should be 1. An error between three and four per cent is associated. This may be due to the boundary conditions assumption that lead to a low values of flux distorting the physics: assuming the zero value should give lower values of flux respect to real one. The zero flux should be out of the system. A similar behavior was noted also for the space discretization in two groups. So it appears that the amplitude sets up on value that is not one but close to it. This is less obvious in case of a limited perturbation on space, as it is shown by the transient *RI2* The insertion of delayed does not make things so different: the prompt behaviors

are retrieved. A little thing that goes against the normal comprehension is the amplitude for transient RI_C1. The presence of delayed should be a sort of source for the normal amplitude so it is expected to slow the drop of it. However considering the delayed is slightly less than the other one. This means that even the small part removed by the fraction of delayed was necessary to cause amplitude leveling. Moreover it can be considered also that already with prompt the amplitude reached an equilibrium and with the insertion of delayed the neutrons are removed, causing amplitude decrease. Similarly to what is obtained in sections 4.8 and 4.10. The values of the reactivity removed/inserted are 946 *pcm* for *RI1* and 442 *pcm* for *RI2*.

5.1.3 DD transients

Similarly to what is done for the cross section there is the perturbation of the diffusion lengths.



(a) DD transients **(b)** DD_C transients

Figure 5.4: DD and DD_C transients

The increase of diffusion length means that neutrons can continue for its path without making any collision: the consequence is the lower probability to make fission or absorption. This is the reason why the effect on amplitude is not so big because the removal or the production properties are not so modified.

5.2 3D cases

To test the capability of the solver it is performed the study of a 3D case. It is very simple as well as the transients. Despite this the geometry wants to be

similar to a real case. This is the reason of the sizes chosen and also the kind of transients performed. They are two: one where there is the augmentation of absorption cross section, the other where there is the rise of fission cross section. The figures 5.7 and 5.8, at the end of the chapter, and the table 5.2 show the data important for the evaluation of the amplitude.

Two groups data		D_g [cm]	$\Sigma_{a,g}$ [cm^{-1}]	$\Sigma_{g\to g+1}$ [cm^{-1}]	$\nu\Sigma_f$ [cm^{-1}]	χ [-]
Core	1	1.01	0.01	0.008	0.003946	1.0
	2	1.2	0.1	-	0.165	0
Reflector	1	0.9	0.01	0.0095	0.0	1.0
	2	0.8	0.1	-	0.0	0

Table 5.2: two groups data.

The first transient studied wants reproduce the first one of two dimensional cases, so it is extended in x direction and at the edge of this. In this one it is studied just the increase of absorption cross section. The other study, wants to observe the effect on fission cross section. This tries to simulate the removal of a control rod from the center of the reactor and the amplitude response is seen. The distribution of flux is not reported for space region but it is found again the symmetry of the quantities due to the configuration.

5.2.1 3D1

Here it is reported the study on an increase of the removal cross section.

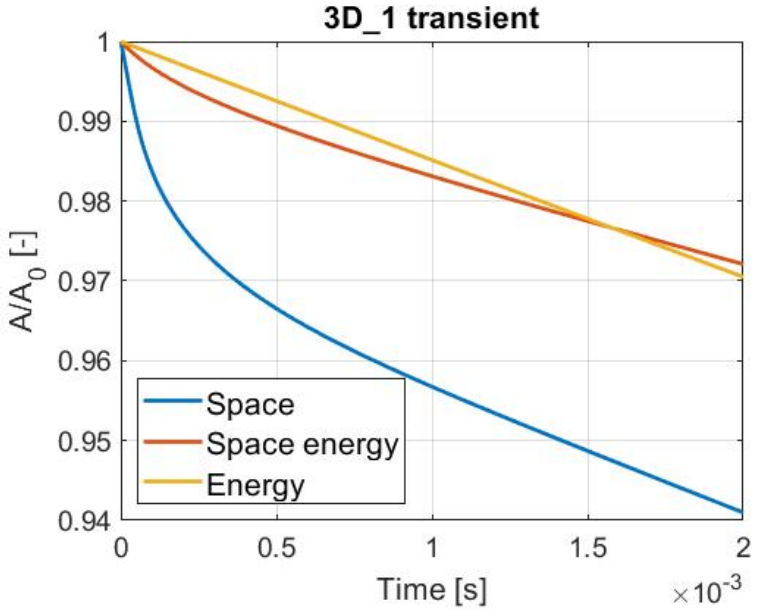


Figure 5.5: 3D_1 transient

The region increase is small and it is similar to the one of *HE1*. The results are the one expected: the amplitude tends to decrease. Again the space discretization does not correspond to the other types, giving lower results. The reason of this is due to the bad factorization. Using only the space to study the amplitude trend considers that in the system there is only the increase of the removal of neutrons. On the other side with the energy discretization, the decrease is slower, since a neutron does not only disappear but it can just pass to the other group.

5.2.2 3D2

In this section it is presented the increase of fission cross section. It is considered like the expulsion of a rod.

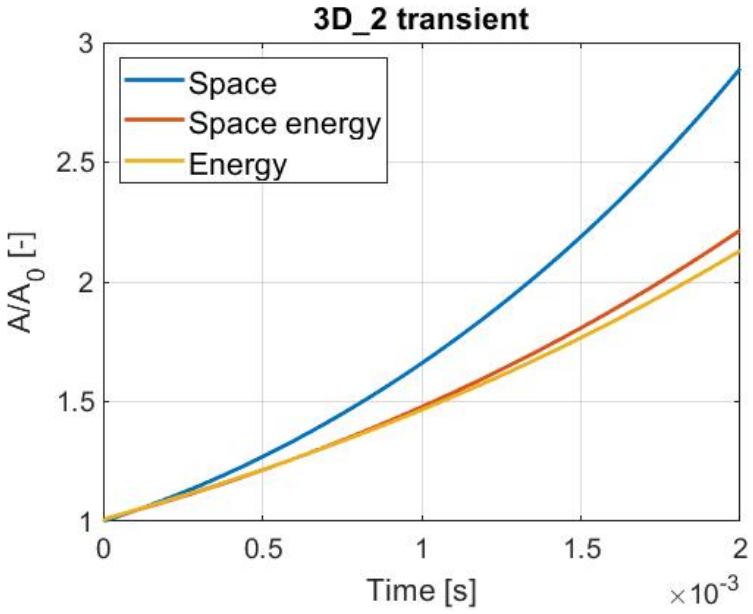
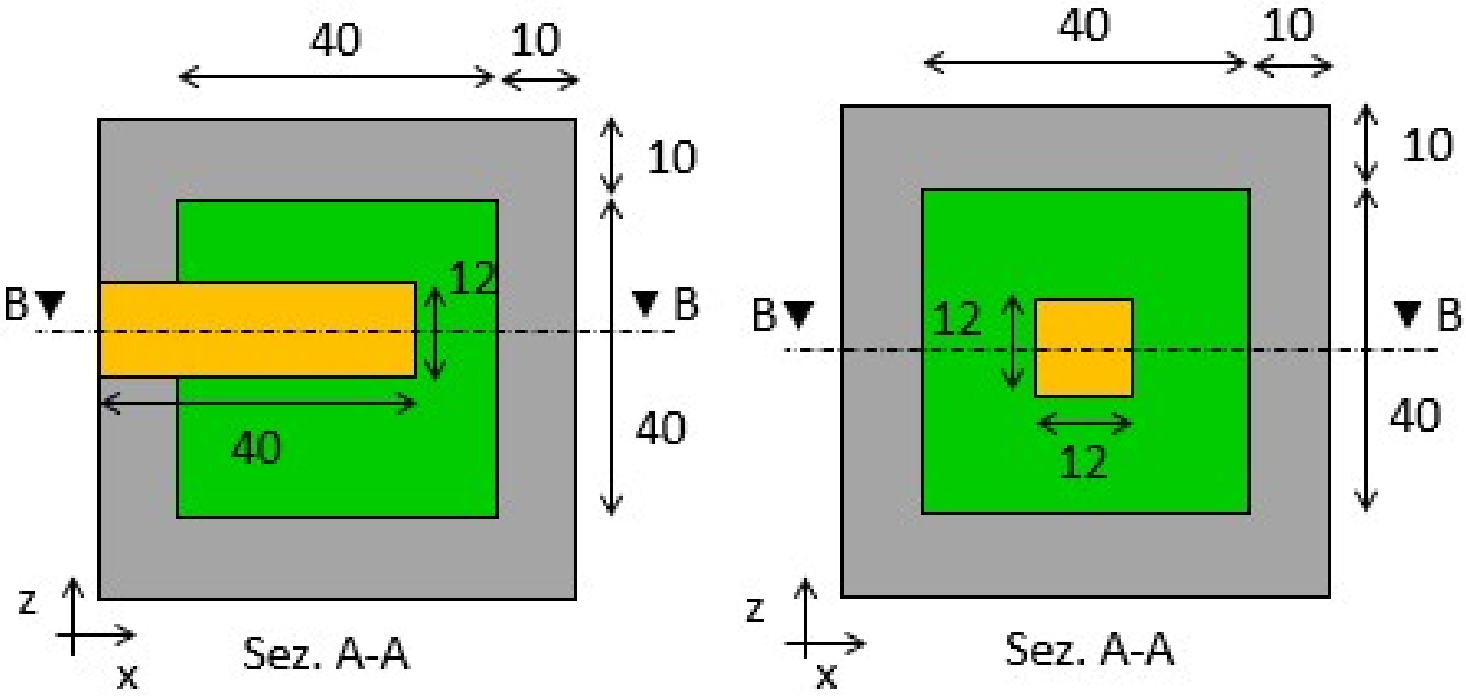


Figure 5.6: 3D_1 transient

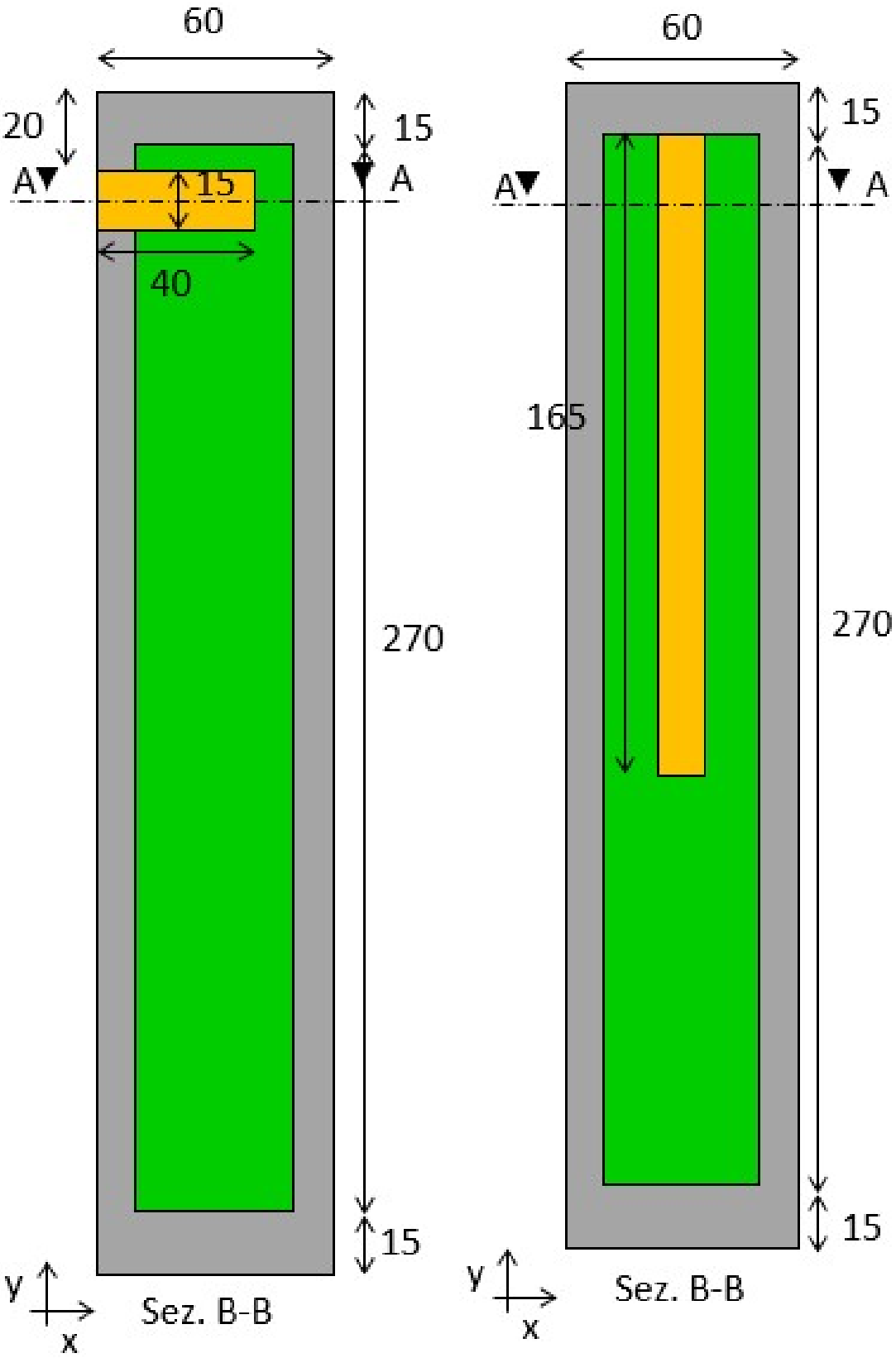
The graph show what is obtained before. The space discretization overestimates the results given by the energy and space energy factorization. However it is noted that the errors committed are sligtly lower than the ones realized in the transients *FA1* and *FA2*. At instant $t = 0.002\ s$ they are 3.87 % and 30.5 %. Maybe is due to the perturbation size, that is extended along y direction so consideration done in 4.5 can be extended to this case.



(a) Section B-B of the system

(b) Section B-B of the system

Figure 5.7: 3D systems



(a) Section A-A of the system **(b)** Section A-A of the system

Figure 5.8: Sections A-A of the 3D systems

Chapter 6

Conclusion

In this chapter some final conclusions are done, considering what is simulated and the results extracted in the previous two chapters. Finally also a list of the needed updates is done.

Most of the studies are the ones that consider two groups in a 2D system. Overall it is noted that the EQS method can offer a good prediction of the results. Not strange effect are noted like decrease of power when it is expected to rise. The space discretization seems to offer overestimation or underestimation of the results respect to the two other types of factorization. However the errors are not so big, if it is considered that only a perturbation is activated and the consequent trend of amplitude is expected. These tend to increase with time and with the magnitude of the perturbation. In case of reactivity compensation it is noted that the method is not so good to predict the expected trend that wants that the amplitude sets up to one. This is achieved well by the space and space-energy discretization but not always. It is noted that in case of an horizontal wide and not centred perturbation it works well, even in case of mono-energetic approximation despite some errors. The energy discretization seems to fail. However if the perturbation is small and distributed it seems to work. Conversely if the modification is localized while the fission cross section is increased in a large space the compensation fails. Even space energy and energy discretization differ since this one is not able to report the trend of quantities on y direction. Indeed it was noted that bigger differences appear when the perturbation is limited on vertical direction. The failure is bigger depending also on the location: if it is more displaced, towards the edge of the system, the errors are bigger. In case of fission and scattering cross section rise the space discretization seems to differ a lot respect to the one of energy and space-energy. These offer always lower results respect to the first one: maybe this is due to the capability to distinguish for all the period of study the division in groups. A reversed behaviour is expected in case of decrease of scattering and fission cross section. Finally it seems that the diffusion length can be studied equally by

the three types of discretization. Moreover it is noted that space discretization give more different results when the properties changed are the ones that have a direct effect on coupling. The insertion of delayed generally seem to agree with the prompt evaluations. Also the one group case seem to offer results that do not differ from the expectations. The three dimensional problem is studied well and the behaviours similar to previously cited are again found. The errors between the space and the others two discretizations are lower respect to the 2D case in case of fission cross section augmentation. Maybe this is due to the further averaging that integration on another direction causes.

Concerning the needed updates to the code, many things should be inserted. Starting from the begin of the solver, the possibility to consider more complex system should be investigated. A big lack of the code is the impossibility to consider a source of neutron. Even if this is rare inside a reactor, it could be important for didactic purposes. This should be considered from the reference flux evaluation, modifying the code. Another interesting developing could be the increase of number of groups studied. Actually they are just two but more of them could offer simulations that are closer to the real physics. To this, also the possibility to the cross sections to change on time enhances a lot prediction of a true reactor system. Finally as it can be seen by the hypothesis 2.9 and the successive of this type, the flux is factorized by the amplitude and the reference flux that is the one of the solution at instant equal to zero. This approximation is good for limited time period studied, that in case of EQS method could be also larger respect to the QS study, but of course for longer ones there is the need of updates. This is quite difficult in OpenFOAM® because it requests a continuous change of mesh, internal to the code.

Bibliography

[1] Greenshields C. J., 2015, *OpenFOAM - The Open Source CFD Toolbox - Programmer's guide*, OpenFOAM

[2] Wikipedia,2021, *GNU General Public License*, fromhttps://en.wikipedia.org/wiki/GNU_General_Public_License.

[3] Wikipedia, 2021, *OpenFOAM*, from https://en.wikipedia.org/wiki/OpenFOAM.

[4] Wikipedia, 2021, *Fluidodinamica computazionale*, fromhttps://it.wikipedia.org/wiki/Fluidodinamica_computazionale.

[5] OpenFOAM Wiki, *Nuclear Technical Committee*, fromhttps://wiki.openfoam.com/Nuclear_Technical_Committee

[6] Fiorina C., Clifford I., Aufiero M., Mikityuk K., 2015, *GeN-Foam: a novel OpenFOAM® based multi-physics solver for 2D/3D transient analysis of nuclear reactors*, Nuclear Engineering and Desgin, 294, pp. 24-37.

[7] Aufiero M., Brovchenko M., Cammi A. Clifford I., Geoffroy O., Heuer D., Laureau A., Losa M., Luzzi L., Merle-Lucotte E., Ricotti M. E., Rouch H., 2013, *Calculating the effective delayed neutron fraction in the Molten Salt Fast Reactor: Analytical , deterministic and Monte Carlo approaches*, Politecnico di Milano, from https://re.public.polimi.it/retrieve/handle/11311/761453/386932/Calculating%20the%20effective%20delayed%20neutron%20fraction%20in%20the%20Molten%20Salt%20Fast%20Reactor_11311-761453_Luzzi.pdf

[8] Ma Y., Wang Y., Yang J., 2021, *ntkFoam: An OpenFOAM based neutron transport kinetics*, Computers and Mathematics with Applications, 81, pp. 512-531

[9] Cosgrove P., Shwageraus E.,2017, *Development of MoCha-Foam: a new Method of Characteristics Neutron Transport Solver for OpenFOAM*, International Conference on Mathematics & Computational Methods Applied to Nuclear Science & Engineering,Jeju, Korea.

[10] Fiorina C. M., Kerkar N., Mikityuk K., Rubiolo P., Pautz A., 2016, *Development and verification of the neutron diffusion solver for the GeN-Foam multi-physics platform*, Annals of nuclear energy, 96, pp. 212-222.

[11] Stewart J., 2020, *Gen-Foam Multiphysics model development for molten salt Reactor*, UNLV, Theses, Dissertations, Professional Papers and Capstones. 3962.

[12] Cervi E., Lorenzi S., Cammi A., Luzzi L., 2019, *Development of an SP3 neutron transport solver for the analysis of the Molten Salt Fast Reactor*, Nuclear Engineering and Design, 346, pp. 209-219.

[13] Clifford I. D., 2007, *Object-oriented multi-physics applied to spatial reactor dynamics*, Mini-dissertation submitted for the degree of Master of Engineering (nuclear), Potchefstroom Campus of the North-West University.

[14] Di Lecce,2018, *Neutronic and thermal-hydraulic simulations for Molten Salt Fast Reactor safety assessment*,MsC thesis, Corso di Laurea in Ingegneria Energetica e Nucleare, Politecnico di Torino.

[15] Dulla S., Nervo M., 2013, *Performance of non-conventional factorization approaches for neutron kinetics*, M & C 2013 : International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering, Sun Valley, Idaho, US.

[16] Ravetto P., Coppa G. G. M., Cassina G. A., Larosa A. M., Mattioda F., Roberti M., 1996,*Multidimensional Reactor Dynamics with the Enhanced Quasi-Static Method*, International Conference on the Physics of Reactor, PHYSOR '96,pp. 80-89, Mito.

[17] CFD-online, 02/2006, *Generic scalar transport equation*, from https://www.cfd-online.com/Wiki/Generic_scalar_transport_equation.