

COUPLED NEUTRONICS AND THERMAL-HYDRAULICS TRANSIENT CALCULATIONS BASED ON A FISSION MATRIX APPROACH: APPLICATION TO THE MOLTEN SALT FAST REACTOR

A. Laureau, M. Aufero, P.R. Rubiolo, E. Merle-Lucotte, and D. Heuer

LPSC, Université Grenoble-Alpes, CNRS/IN2P3
53 rue des Martyrs, F-38026 Grenoble Cedex, France
axel.laureau@lpsc.in2p3.fr

ABSTRACT

This work presents a time dependent version of the fission matrix method named Transient Fission Matrix (TFM) developed to perform kinetics calculations. Coupled neutronics and thermal-hydraulics transient calculations are studied using the TFM approach and a Computational Fluid Dynamics (CFD) code. The generation of the matrices is performed using the Monte Carlo neutronic code SERPENT beforehand the transient calculation. The neutronic module and the coupling are directly implemented in the CFD opensource code OpenFOAM. An application case is presented on the Molten Salt Fast Reactor (MSFR). This system is a circulating liquid fuel reactor characterized by a two meter core cavity and a fast spectrum. Thus the present approach is well suited since an accurate distribution of the velocity of the liquid fuel circulating in the cavity and of the delayed neutron precursor transport is required. A reactivity insertion incident of around 2.5\$ is presented, showing the good behavior of the MSFR in such a case. A load-following from 50% to the nominal power generation is also discussed.

Key Words: Transient, Fission Matrix, Coupling, Neutronics, Thermal-Hydraulics

1 INTRODUCTION

This paper presents a new approach designed to perform coupled transient calculations close to the Monte Carlo approach for the neutronics while sustaining a low computational cost. This method is based on a time dependent version of the fission matrix approach to solve the neutron kinetics response of the reactor. One of the advantages of this approach is that the fission matrices can be computed using the Monte Carlo neutronic code prior to the transient calculations to provide the neutronic reactor behavior during the transient. An accurate estimation of the neutronics kinetics response through time is obtained with a conversion of the Monte Carlo transport by matrix products. The transient reactor calculations are performed by coupling the neutronics model to a core thermal-hydraulic model developed in the Computational Fluid Dynamics (CFD) code OpenFOAM. The use of a CFD code allows us to calculate the 3D velocity and temperature distributions. The latter has a significant impact on the neutronic behavior through the induced variations in the neutron cross-sections. The neutronics model is inspired of the fission matrix approach, with the adjunction of a temporal aspect as detailed in section 2. The thermal-hydraulics modeling, together with the coupling strategy and the numerical implementation of this approach are also presented in section 2.

An application case of transient calculations on the Molten Salt Fast Reactor (MSFR) is discussed in section 3. This reactor design has been selected by the Generation IV International Forum because of its promising design and safety features. The present approach is well suited for such a reactor since the core is composed of a circulating liquid fuel, implying a motion of the delayed neutron precursors [1]. Two cases are presented: an incident transient (reactivity insertion) and a load following transient.

2 METHODOLOGY

2.1 Neutronics modeling

In this section the fission matrix utilization in current calculation codes is discussed, together with the improvements required to set up the proposed kinetics model.

2.1.1 Fission Matrix Introduction

The Fission matrix method is usually employed to accelerate the source convergence in neutronics Monte Carlo codes ([2] [3]). Indeed, the eigenvector of these matrices gives an estimation of the equilibrium neutron source shape (i.e. the propagation of any source distribution after an infinity of generations). Fission matrices can also be used to estimate the different modes of the neutron source shape (corresponding to the different eigenvectors) and the associated time constants (the eigenvalues) [4].

The fission matrix method is an application of a Green function approach. This Green function aims at characterizing the reactor response during one generation of neutrons due to a pulse of neutron injection. For each discrete position of space j , fission neutrons are created and their histories are tracked during one generation. The number of fission neutrons induced by these neutrons histories is scored for each position i . This quantity can be directly estimated using a neutronics Monte Carlo calculation and is the general term of line i and column j in the fission matrix. In this way, for a given neutron distribution in the core represented as a vector with the quantity of neutrons in the cell j , the Monte Carlo propagation of this distribution during one generation of neutrons is pre-calculated in the fission matrix and is given by a simple matrix-vector product.

The objective of the approach presented here is to realize transient calculations, adding a temporal aspect to the fission matrix approach.

2.1.2 The Transient Fission Matrix (TFM) Approach

We call $\tilde{G}_{\chi_p \nu_p}(\mathbf{r}', \mathbf{r})$ the continuous operator associated to the fission matrix representing the probability that a prompt neutron created in \mathbf{r}' (i.e. column j) induces a new prompt neutron in \mathbf{r} (i.e. line i), \mathbf{r}' and \mathbf{r} being defined in the reactor space \mathcal{R} . This operator represents a neutron production density per source neutron injected and is expressed in $n_{prod}/m^3/n_{src}$, the produced

neutrons n_{prod} being the neutron source n_{src} of the next generation. The terms χ_p and ν_p refer to the prompt neutron emission spectrum and to the prompt neutron production per fission. The delayed neutron influence will be discussed afterwards. For a given distribution of prompt neutrons $N_g(\mathbf{r})$ at generation g expressed in n_{src}/m^3 , the image (Equation 1) of this source by the operator $\tilde{G}_{\chi_p\nu_p}$ is the associated source of the next generation $g + 1$: $N_{g+1}(\mathbf{r})$. This product corresponds to a matrix-vector product with the discretized version of $\tilde{G}_{\chi_p\nu_p}$ and N .

$$N_{g+1}(\mathbf{r}) = \left| \tilde{G}_{\chi_p\nu_p}(\mathbf{r}', \mathbf{r}) \middle| N_g(\mathbf{r}') \right\rangle = \int_{\mathbf{r}' \in \mathcal{R}} \tilde{G}_{\chi_p\nu_p}(\mathbf{r}', \mathbf{r}) \cdot N_g(\mathbf{r}') d\mathbf{r}' \quad (1)$$

It is also possible to create a time dependent version of this operator associating the probability between a creation in \mathbf{r}' at time t' and the production of a new neutron in \mathbf{r} at t . This quantity only depends on the time elapsed between the source neutron creation and the fission neutron production resulting of the transport of the original input neutron. This operator is written $G_{\chi_p\nu_p}(t', t, \mathbf{r}', \mathbf{r})$, expressed in $n_{prod}/m^3/s/n_{src}$. For a given source distribution of prompt neutrons $S_g(t, \mathbf{r})$ in $n_{src}/m^3/s$, the convolution product image (Equation 2) of this source by the operator $G_{\chi_p\nu_p}$ is the associated source at the next generation: $S_{g+1}(\mathbf{r}, t)$. If the operator $G_{\chi_p\nu_p}$ has a time discretization, it can be represented as a series of fission matrices, one for each time step. Then this convolution corresponds to a series of matrix-vector product.

$$S_{g+1}(\mathbf{r}, t) = \left| G_{\chi_p\nu_p}(t', t, \mathbf{r}', \mathbf{r}) \middle| S_g(\mathbf{r}', t') \right\rangle = \iint_{t' < t, \mathbf{r}' \in \mathcal{R}} G_{\chi_p\nu_p}(t', t, \mathbf{r}', \mathbf{r}) \cdot S_g(\mathbf{r}', t') d\mathbf{r}' dt' \quad (2)$$

2.1.3 Fission to fission time l_{eff} calculation

In the next sub section, two versions of the kinetics equation for a neutron population in a reactor with different computational costs are described. The simplified one requires an upstream estimation of the effective fission to fission time, with corresponds to the effective prompt life time l_{eff} as discussed below. This subsection is devoted to the calculation of this quantity.

As shown in equation 3, it is possible to create an operator representing the first time moment of the neutron production through time: $T_{\chi_p\nu_p}$. It can be constructed using the operator $G_{\chi_p\nu_p}$ if the accuracy of its time discretization is correct. Another solution is to estimate it directly in the Monte Carlo calculation code as a matrix (spacial discretization) by integrating, during the neutron history, a score $(t'' \cdot \nu_p \sigma_f^{i \leftarrow j} \phi)$ where $t'' = t - t'$ is the elapsed time between the neutron creation and the interaction.

$$T_{\chi_p\nu_p}(\mathbf{r}', \mathbf{r}) = \frac{\int_{t'' > 0} G_{\chi_p\nu_p}(t'', \mathbf{r}', \mathbf{r}) \cdot t'' dt''}{\int_{t'' > 0} G_{\chi_p\nu_p}(t'', \mathbf{r}', \mathbf{r}) dt''} \quad (3)$$

Then, the product $T_{\chi_p\nu_p}(\mathbf{r}', \mathbf{r}) \cdot \tilde{G}_{\chi_p\nu_p}(\mathbf{r}', \mathbf{r})$, which is a term to term multiplication for the elements of the associated matrices, is the neutron production associated to the average time of the response from \mathbf{r}' to \mathbf{r} . Finally, with N_p the eigenvector of $\tilde{G}_{\chi_p\nu_p}$ which corresponds to the equilibrium prompt

neutron source in the reactor, the average fission to fission time l can be calculated using equation 4. The prompt life time is different from l since the non-fission reactions are not taken into account.

$$l = \frac{\iint_{\mathbf{r}' \in \mathcal{R}, \mathbf{r} \in \mathcal{R}} \left[T_{\chi_p \nu_p}(\mathbf{r}', \mathbf{r}) \cdot \tilde{G}_{\chi_p \nu_p}(\mathbf{r}', \mathbf{r}) \right] N_p(\mathbf{r}') \, d\mathbf{r}' \, d\mathbf{r}}{\iint_{\mathbf{r}' \in \mathcal{R}, \mathbf{r} \in \mathcal{R}} \tilde{G}_{\chi_p \nu_p}(\mathbf{r}', \mathbf{r}) N_p(\mathbf{r}') \, d\mathbf{r}' \, d\mathbf{r}} \quad (4)$$

The fission to fission time thereby calculated does not take into account importance effects. The importance of the position \mathbf{r} in the core is the amount of neutrons coming from this position. The fission matrix associated to the operator $\tilde{G}_{\chi_p \nu_p}$ represents the probability that a neutron created in j (i.e. \mathbf{r}') induces a new fission neutron in i (i.e. \mathbf{r}). Thus the transpose matrix represents the probability that a neutron created in j comes from a previous history stated in i , and its eigenvector N_p^* is by definition the importance of the source distribution. If we call $N_p^*(\mathbf{r})$ the importance function associated to this vector, the effective fission to fission time l_{eff} can be calculated using equation 5. It corresponds to the effective prompt life time since non-fission reactions have a zero-importance

$$l_{eff} = \frac{\iint_{\mathbf{r}' \in \mathcal{R}, \mathbf{r} \in \mathcal{R}} N_p^*(\mathbf{r}) \left[T_{\chi_p \nu_p}(\mathbf{r}', \mathbf{r}) \cdot \tilde{G}_{\chi_p \nu_p}(\mathbf{r}', \mathbf{r}) \right] N_p(\mathbf{r}') \, d\mathbf{r}' \, d\mathbf{r}}{\iint_{\mathbf{r}' \in \mathcal{R}, \mathbf{r} \in \mathcal{R}} N_p^*(\mathbf{r}) \tilde{G}_{\chi_p \nu_p}(\mathbf{r}', \mathbf{r}) N_p(\mathbf{r}') \, d\mathbf{r}' \, d\mathbf{r}} \quad (5)$$

2.1.4 Equation of the prompt neutron kinetics

As previously mentioned (Equation 2), the convolution of $G_{\chi_p \nu_p}(t', t, \mathbf{r}', \mathbf{r})$ by a prompt source distribution $S_g(\mathbf{r}, t)$ gives the source distribution of the next generation $S_{g+1}(\mathbf{r}, t)$. Then if we consider $S(\mathbf{r}, t)$, the total prompt neutron source, due to non-variation of the operator $G_{\chi_p \nu_p}$ for the different generations and the linearity of equation 2, the equation of the prompt neutron source can be written as the convolution on all the past neutron sources with $G_{\chi_p \nu_p}$ (Equation 6). We can perform a kinetics evolution limited to the prompt neutrons thanks to this equation. The numerical integration of this quantity over one time step requires a series of matrix-vector multiplication (one by time discretization of $G_{\chi_p \nu_p}$).

Considering perturbations with a time constant much longer than l_{eff} , it is possible to simplify this equation by considering the neutron source as a neutron population $N(\mathbf{r}, t)$ creating fissions with a time constant of l_{eff} . Thus, during dt , $N(\mathbf{r}, t) \frac{dt}{l_{eff}}$ prompt neutrons disappear, creating $\left| \tilde{G}_{\chi_p \nu_p}(\mathbf{r}', \mathbf{r}) \right| N(\mathbf{r}, t) \frac{dt}{l_{eff}}$ new prompt neutrons (Equation 7). The advantage of this formulation is to require a much lower computational cost.

$$S(\mathbf{r}, t) = \left| G_{\chi_p \nu_p}(t', t, \mathbf{r}', \mathbf{r}) \right| S(\mathbf{r}', t') \quad (6)$$

$$\frac{dN(\mathbf{r}, t)}{dt} = \left| \tilde{G}_{\chi_p \nu_p}(\mathbf{r}', \mathbf{r}) \right| N(\mathbf{r}, t) \frac{1}{l_{eff}} - N(\mathbf{r}, t) \frac{1}{l_{eff}} \quad (7)$$

2.1.5 Equation of the neutron kinetics

To characterize the evolution of the prompt neutrons and of the delayed neutrons in the reactor, separate operators have to be defined and computed. A neutron can produce prompt and delayed neutrons by fission, thus the amount of prompt neutrons produced by fission ν_p and the amount of delayed neutrons produced by fission ν_d must be used. On the other hand, these two quantities must be estimated twice: with an incident prompt emission spectrum χ_p if the initial neutron is a prompt one, and otherwise with an initial delayed emission spectrum χ_d . Thus, four operators are required: $G_{\chi_p\nu_p}$, $G_{\chi_p\nu_d}$, $G_{\chi_d\nu_p}$ and $G_{\chi_d\nu_d}$. Note that the number of operators is limited to four by approximating the delayed emission spectrum of the different families by the averaged one.

All the quantities required to realize a kinetics model can be calculated as follows (Equation 8), where $P_i(\mathbf{r}, t)$ is the precursor population of the family i , λ_i its associated decay constant, and $\frac{\beta_i}{\beta_0}$ its normalized delayed fraction ($\sum_i \frac{\beta_i}{\beta_0} = 1$).

$$\begin{aligned} \frac{dP_i}{dt}(t, \mathbf{r}) &= \frac{\beta_i}{\beta_0} \left[\left| G_{\chi_p\nu_d}(t-t', \mathbf{r}', \mathbf{r}) \left| S(t', \mathbf{r}') \right\rangle + \left| G_{\chi_d\nu_d}(t-t', \mathbf{r}', \mathbf{r}) \left| \sum_i \lambda_i P_i(t', \mathbf{r}') \right\rangle \right] - \lambda_i P_i \right. \\ S(t, \mathbf{r}) &= \left. \left| G_{\chi_p\nu_p}(t-t', \mathbf{r}', \mathbf{r}) \left| S(t', \mathbf{r}') \right\rangle + \left| G_{\chi_d\nu_p}(t-t', \mathbf{r}', \mathbf{r}) \left| \sum_i \lambda_i P_i(t', \mathbf{r}') \right\rangle \right. \right. \end{aligned} \quad (8)$$

Those equations can also be simplified by considering the prompt neutron population $N(\mathbf{r}, t)$. The simplified equations of the kinetics are balanced equations (Equation 9). During dt , $N(t, \mathbf{r}) \frac{dt}{l_{eff}}$ prompt neutrons disappear, creating $\left| \tilde{G}_{\chi_p\nu_p}(\mathbf{r}', \mathbf{r}) \left| N(t, \mathbf{r}') \frac{dt}{l_{eff}} \right\rangle \right.$ new prompt neutrons and also $\left| \tilde{G}_{\chi_p\nu_d}(\mathbf{r}', \mathbf{r}) \left| N(t, \mathbf{r}') \frac{dt}{l_{eff}} \right\rangle \right.$ new precursors. In the same time, $\sum_i \lambda_i P_i(t, \mathbf{r}) dt$ precursors disappear, generating $\left| \tilde{G}_{\chi_d\nu_p}(\mathbf{r}', \mathbf{r}) \left| \sum_i \lambda_i P_i(t, \mathbf{r}') dt \right\rangle \right.$ new prompt neutrons and $\left| \tilde{G}_{\chi_d\nu_d}(\mathbf{r}', \mathbf{r}) \left| \sum_i \lambda_i P_i(t, \mathbf{r}') dt \right\rangle \right.$ new precursors.

$$\begin{aligned} \frac{dP_i}{dt}(t, \mathbf{r}) &= \frac{\beta_i}{\beta_0} \left[\left| \tilde{G}_{\chi_p\nu_d}(\mathbf{r}', \mathbf{r}) \left| N(t, \mathbf{r}') \frac{1}{l_{eff}} \right\rangle + \left| \tilde{G}_{\chi_d\nu_d}(\mathbf{r}', \mathbf{r}) \left| \sum_i \lambda_i P_i(t, \mathbf{r}') \right\rangle \right] - \lambda_i P_i(t, \mathbf{r}) \right. \\ \frac{dN}{dt}(t, \mathbf{r}) &= \left. \left| \tilde{G}_{\chi_p\nu_p}(\mathbf{r}', \mathbf{r}) \left| N(t, \mathbf{r}') \frac{1}{l_{eff}} \right\rangle + \left| \tilde{G}_{\chi_d\nu_p}(\mathbf{r}', \mathbf{r}) \left| \sum_i \lambda_i P_i(t, \mathbf{r}') \right\rangle \right. - \frac{1}{l_{eff}} N(t, \mathbf{r}) \right. \end{aligned} \quad (9)$$

Once the operators $G_{\chi_x\nu_x}$ and $T_{\chi_p\nu_p}$ computed, a kinetics evolution calculation can be performed. Before that, another aspect has to be taken into account to carry out transient calculations with a fuel temperature evolution.

2.1.6 Fission Matrix interpolation

During transient calculations, the neutron behavior in the core will change due to reactivity feedback effects. Because of the huge difference between the fission to fission time and the characteristic time of the feedback effects (linked to the thermal-hydraulics), we can consider the operator G constant during one generation of neutrons. However the drift of G along neutron generations must be taken into account.

Calculating all the TFM operators during the transient would be very time consuming since it would require new Monte Carlo calculations. Another solution is to calculate the evolution of this operator with the density (thermal expansion) and the Doppler effects. A linear evolution for the density and a logarithmic evolution for the Doppler by analogy with the group constants in multi-group diffusion codes is considered.

If we consider the reference operator G_{ref} calculated at the temperature T_{ref} , G^ρ calculated by modifying the density to reproduce the thermal expansion of the temperature $T_{ref} + \Delta T$, and $G^{Doppler}$ calculated by changing the cross sections to correspond to the temperature $T_{ref} + \Delta T$, it is possible to create the following operators that will be used for the interpolation:

$$\Delta_\rho G(\mathbf{r}', \mathbf{r}) = \frac{G^\rho(\mathbf{r}', \mathbf{r}) - G_{ref}(\mathbf{r}', \mathbf{r})}{\Delta T} \text{ and } \Delta_{Doppler} G(\mathbf{r}', \mathbf{r}) = \frac{G^{Doppler}(\mathbf{r}', \mathbf{r}) - G_{ref}(\mathbf{r}', \mathbf{r})}{\log \frac{T_{ref} + \Delta T}{T_{ref}}} \quad (10)$$

The model proposed (Equation 11) can be used to interpolate the value of $G(\mathbf{r}', \mathbf{r})$ for a given temperature distribution $T(\mathbf{r})$. This model only takes into account the neutron's cell position temperature.

$$G(\mathbf{r}', \mathbf{r}) = G_{ref}(\mathbf{r}', \mathbf{r}) + (T(\mathbf{r}') - T_{ref}(\mathbf{r}')) \Delta_\rho G(\mathbf{r}', \mathbf{r}) + \log \frac{T(\mathbf{r}')}{T_{ref}(\mathbf{r}')} \Delta_{Doppler} G(\mathbf{r}', \mathbf{r}) \quad (11)$$

2.2 Thermal Hydraulics modeling

2.2.1 Introduction

Different strategies can be employed to solve the Navier-Stokes equations modeling the hydraulics phenomena. The first one named Direct Numerical Simulation (DNS) consists in solving them directly, including all the dimensions of vortexes in a turbulent flow. This approach is not applicable for an industrial application with a complex system like a nuclear reactor. The second one named Large Eddy Simulation (LES) consists in cutting the lower level of vortexes using mathematical models. This approach can be employed on complex geometries but requires a huge computational cost. The third one is the Reynolds-Averaged Navier-Stokes (RANS) approach, consisting in the resolution of the time-average value of the different parameters, modeling the fluctuations with different models. The high frequencies are cut off in the RANS approach. A good estimation of the hydraulics pattern is obtained with a reasonable cost, even if, for a fuel assembly with a lot of solid-liquid interfaces and boundary layers, the cost remains very high. Those three approaches are grouped as the Computational Fluid Dynamics (CFD).

Other approaches exist. For a reactor with pin fuel, a thermal-hydraulics calculation with a low cost is carried out using a sub-channel model which describes the core as an arrangement of fuel channels. This approach is common for PWR reactors.

The coupling with the neutronics can be performed with different thermal-hydraulics models. The application case developed in the last section is the MSFR, characterized by a large cavity. Thus a good description of the flow pattern with potential recirculation is required. For this reason, the RANS approach has been employed in this study.

2.2.2 RANS model

The fluid is considered as incompressible $\rho = \rho_0$, the mass equation can be simplified as:

$$\nabla \cdot (\bar{\mathbf{u}}) = 0 \quad (12)$$

Where $\bar{\mathbf{u}}$ is the average of the velocity vector \mathbf{u} and $\nabla \cdot ()$ the divergence operator.

The momentum equation is:

$$\begin{aligned} \frac{\partial(\bar{\mathbf{u}})}{\partial t} + \nabla \cdot (\bar{\mathbf{u}} \otimes \bar{\mathbf{u}}) = & -\frac{1}{\rho_0} \nabla \left(\bar{p} + \frac{2}{3}k \right) + \nabla \cdot \left(\nu_{eff} \left(\frac{1}{2} (\underline{\nabla}(\bar{\mathbf{u}}) + \underline{\nabla}(\bar{\mathbf{u}})^t) - \frac{2}{3} \nabla \cdot (\bar{\mathbf{u}}) \underline{Id} \right) \right) \\ & + \mathbf{g} (1 + \beta_{boyancy} (\bar{T} - T_0)) \end{aligned} \quad (13)$$

Where \bar{p} is the average pressure. The kinetic turbulent energy k and the turbulent viscosity ν_{eff} are both calculated by the turbulence model. In this study, the k-epsilon realizable turbulence model is used. The last term corresponds to the Boussinesq approximation, representing the buoyancy force modeled as an external force: $\mathbf{g} (1 + \beta_{boyancy} (\bar{T} - T_0))$, where $\beta_{boyancy}$ is the fluid thermal expansion coefficient, \bar{T} the average temperature and \mathbf{g} the gravity acceleration.

Finally, the energy balance equation is:

$$\frac{\partial \bar{T}}{\partial t} + \nabla \cdot (\bar{T} \bar{\mathbf{u}}) = \kappa_{eff} \Delta (\bar{T}) + S_{external} \quad (14)$$

Where κ_{eff} is the effective diffusivity coefficient, which accounts for the turbulent diffusivity. We can notice that the external source term $S_{external}$ corresponds to the power released by the fissions and calculated by the neutronics module.

2.3 Numerical implementation

The two main codes employed in this study are SERPENT for the neutronics, and OpenFOAM for the Thermal-Hydraulics. SERPENT is a Monte Carlo neutronic calculation code [5]. OpenFOAM is an open source CFD calculation code [6]. In both cases, modifications have been performed directly in the source code to implement the TFM approach.

2.3.1 Neutronics module

The neutronics module is decomposed in two parts, the calculation of discretized operators $G_{\chi_x \nu_x}$, and the numerical integration of the kinetics equations.

The SERPENT code is employed to compute the transient fission matrices prior to the transient calculation. SERPENT can perform a simulation using a tetrahedral mesh exported from OpenFOAM. For each fission neutron source created in the core during a critical calculation, this neutron has an attribute corresponding to the cell number of its birth, and another attribute indicating if this neutron is a delayed neutron. Then, at each interaction of the neutron transport the probability to create a fission neutron prompt and delayed is scored. The production of fission neutron multiplied by the life time of the neutron is also scored. Finally, the spatially discretized operators $G_{\chi_p \nu_p}$, $G_{\chi_p \nu_d}$, $G_{\chi_d \nu_p}$, $G_{\chi_d \nu_d}$ and $T_{\chi_p \nu_p}$ are estimated using one calculation. The variation of the operator with the density and the Doppler are performed using two other distinct calculations.

Once the matrices generated, the integration of equation 9 has directly been implemented in the Thermal-Hydraulics code since no more Monte Carlo calculations are required. The temperature field calculated by the thermal-hydraulics is used to interpolate the matrices during the transient.

2.3.2 Thermal-Hydraulics module and coupling approach

The thermal-hydraulics solution is obtained using the CFD calculation code OpenFOAM.

A more refined mesh is required to perform the CFD calculation compared to the TFM's one. The mapping of the different fields between the meshes is realized by OpenFOAM using internal libraries based on standard finite volume mapping technics.

As mentioned in section 2.2.2, the power density field in the reactor is a source term in the energy equation. In this study, it is considered as proportional to the neutrons produced by fission to avoid the production of matrices corresponding to the energy deposition from cell j to cell i . All the fission energy is considered as locally released at the fission position.

The case study is a circulating liquid fuel reactor, thus the precursors motion has to be taken into account. Their transport is performed by OpenFOAM, using the source term calculated by the neutronics.

The algorithm employed in the OpenFOAM solver to obtain the implicit convergence of all the coupled parameters consists in iterative calculations of all the physics during one time step. Those iterations are named "outer" iterations. Then, during transient calculations, the solution obtained respects all the solved equations.

The same approach is used for the coupling, including the neutronics module in those outer iterations since the neutron module is directly implemented in OpenFOAM. However, the time steps used for the neutronics are much smaller than the thermal-hydraulics ones. During one thermal-

hydraulics time step, many neutronics time steps are performed (Fig. 1). Thus the temperature field calculated by the thermal-hydraulics is interpolated during the neutronics calculation, and the energy source term or the precursor source term is deduced for the average production of the neutronics steps.

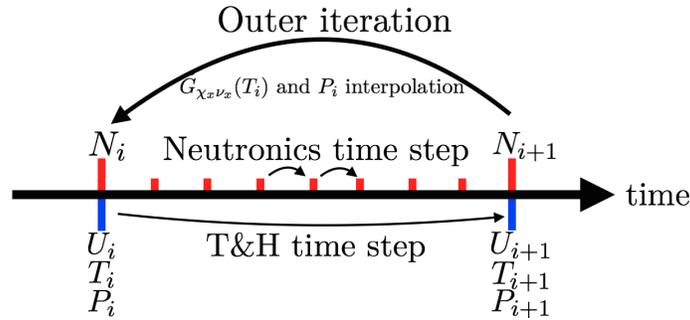


Figure 1. Numerical coupling schema

3 TRANSIENT STUDIES

3.1 Case Presentation

The Molten Salt Fast Reactor (MSFR [7]) is a 3 *GWth* liquid fuel reactor. It is composed of three successive circuits: the fuel circuit, the intermediate circuit cooling the fuel salt in the heat exchangers and the power conversion system.

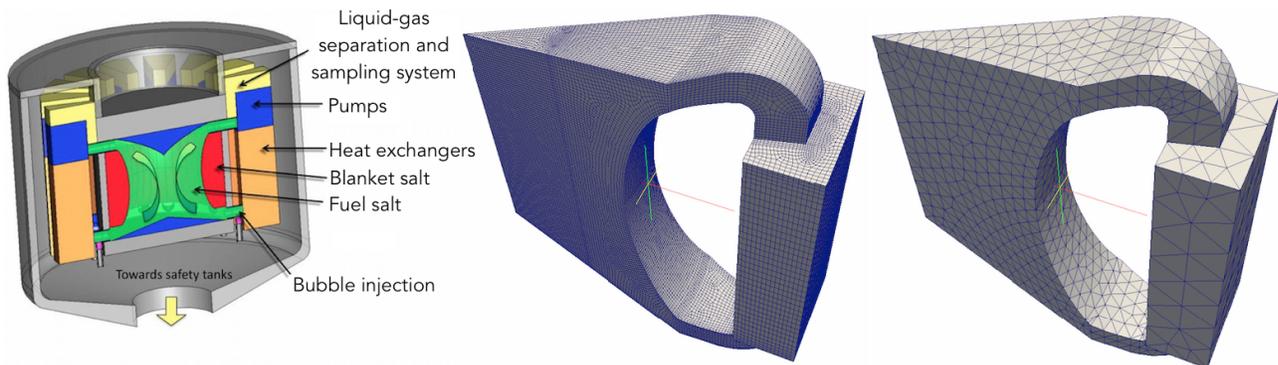


Figure 2. MSFR geometry (left), and CFD (middle) & neutronics (right) mesh on 1/16 of the core

The first circuit (see Fig.2 left) is a molten fluoride salt composed of 77.5mol%*LiF* - 22.5mol%*ThF₄* with some percents of fissile matter (²³³*U* in this case) replacing thorium to reach criticality. A circulation of the fuel salt is required since the molten salt is both the fuel and the coolant. The heat extraction part of the first circuit is composed of 16 loops including a pump, a heat-exchanger, a gas injection and bubble extraction system. The circulation period is around 3 – 4 s (3 s in this study). A radial fertile blanket improves the breeding capabilities of the system. This study is

focused on the fuel salt system. To limit the computational cost and the system complexity, the blanket salt is not modeled in this simulation and symmetric conditions are used to reduce the core simulation to 1/16 of the core (see Fig.2). The fluid is cooled in heat exchangers, modeled as a porous media with a heat transfert coefficient associated to a fixed temperature of the intermediate circuit. The walls are considered as adiabatic with a no-slip condition. A steady state calculation is performed prior to the transient calculation to get an initial converged solution.

Two meshes are used in this simulation, a tetrahedral mesh for the calculation of the neutronics module, and a mostly hexahedral mesh for the thermal-hydraulics (see Fig.2, middle and right). A specific optimisation of the mesh is required for the thermal-hydraulics to capture the flow pattern in the inlet of the core with the detachment of the boundary layer [8].

3.2 Reactivity insertion

The transient presented here consists of an instantaneous insertion of reactivity of 300 pcm in the reactor. Due to the lack of control rod in the reactor, the introduction of a huge amount of reactivity with a time constant lower than the circulation time is unlikely, thus the reactivity is artificially injected instantaneously. To perform such a reactivity insertion, the neutron production is multiplied by $1 + (300 \text{ pcm})$ to correspond to an artificial increase of $\nu\Sigma_f$ on the whole geometry. This case highlights the good reactor behavior even if the perturbation's origin is not physical.

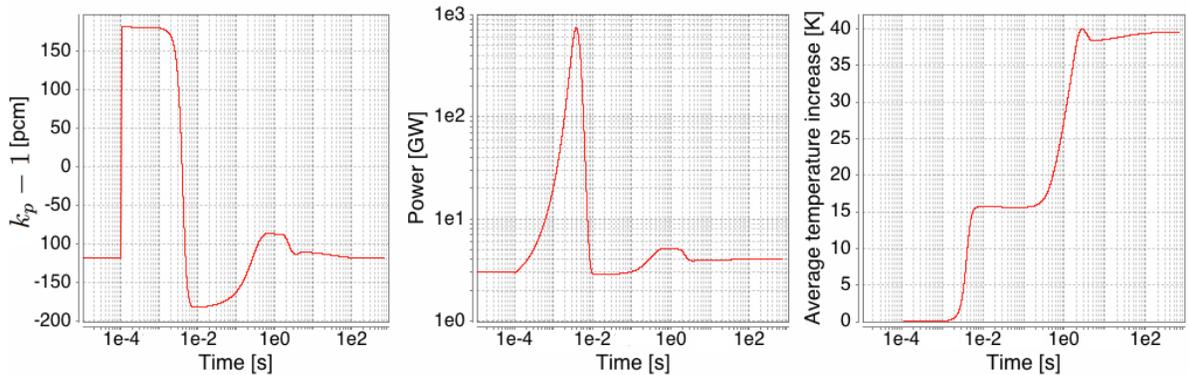


Figure 3. Reactivity insertion: $k_p - 1$ (left), total power (middle) and core average temperature (right)

As presented in Fig.3 after the reactivity insertion at 10^{-4} s, the increase of the reactivity results in a high increase of the power in the reactor. After 4×10^{-3} s, a feedback effect induced by the temperature increase stops the prompt excursion of power. Due to the elevation of the mean temperature at the end of the transient to compensate the reactivity insertion, the heat extraction in the heat exchanger is enhanced and the produced power accordingly increases.

We can notice a rebound oscillation of the reactivity at around 3 seconds, it corresponds to the circulation time of the fuel in the reactor.

3.3 Load following

The second application case is an instantaneous load following transient. The initial condition corresponds to a critical reactor with 1.5 *GWth* power. At the beginning of the simulation, the temperature of the intermediate circuit is reduced to increase the power extracted up to 3*GWth*.

During the first time steps (Fig.5), the salt is cooled in the heat exchanger, a low neutron importance area. For this reason the prompt multiplication factor starts to increase after 0.3 *s*, when the cold salt is transported from the heat exchanger to the core cavity. After one second, the feedback effect stops the increase of the neutron population, and the reactivity progressively goes back to its initial value with a time constant corresponding to the balancing of the delayed neutron precursor population. An oscillation corresponding to the circulating time of the fuel salt can be observed. This application case highlights the good behavior of the reactor to load following transients.

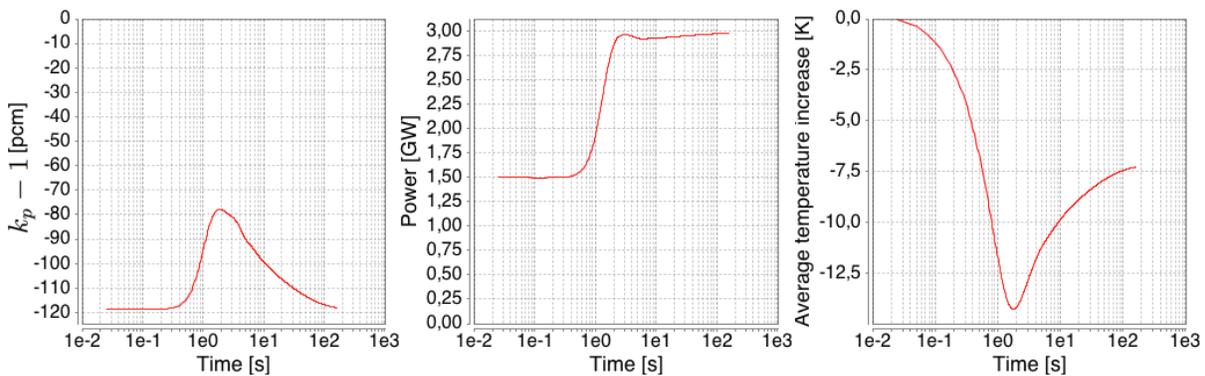


Figure 4. Load following: $k_p - 1$ (left), total power (middle) and average temperature (right)

For a given flow rate, the power increase during the transient results in an increase of the temperature raise between the top and the bottom of the core (see Fig.5). The effect of the complex flow pattern with recirculation localized at the bottom of the core (in blue on the stream line plot of Fig.5) impacts the temperature field.

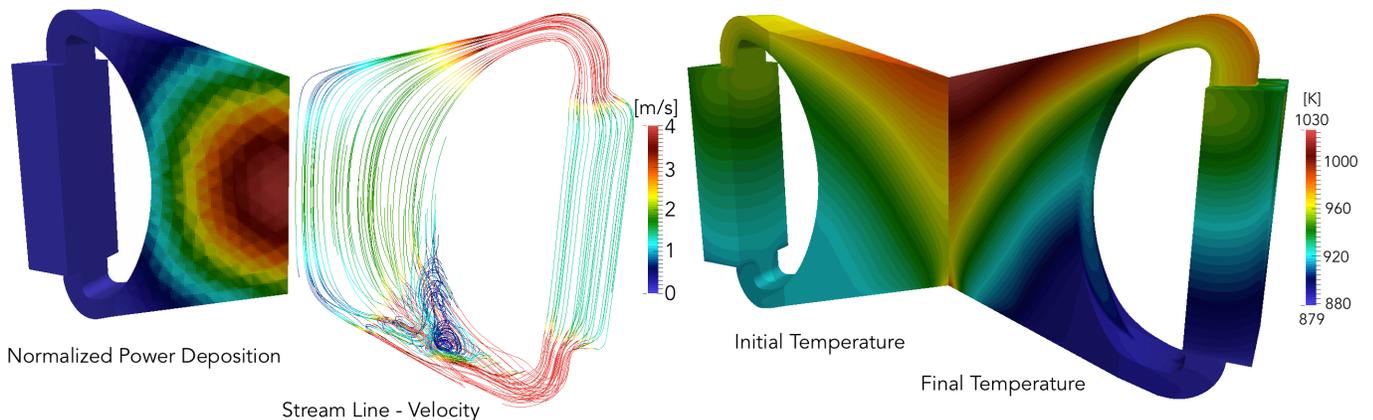


Figure 5. Distribution of power, velocity and temperature

4 CONCLUSION

A good estimation of the time response of the neutron behavior in a reactor can be provided using the Transient Fission Matrix Approach. The major advantage of the method proposed is to reduce the Monte Carlo calculation to the generation of a database of matrices prior to the transient calculation, no more calculation being required during the transient. The neutronic part of the coupling is reduced to matrix products. The associated computational cost is reduced, even if it is directly linked to the spatial discretisation required and to the neutron transport characteristics of the reactor (sparse matrix for a reactor with a small migration area).

Good preliminary results with the expected magnitude response have been obtained with the first transient calculations performed on the MSFR reactor. Future development will be focused on a benchmark to validate the coupled module developed. Other future ways of development would be the addition of the sensibility to the crossed cells' temperature on the transient fission matrices, and the application of the TFM approach on different nuclear systems.

5 ACKNOWLEDGMENTS

The authors wish to thank the NEEDS French program, the IN2P3 department of the CNRS, Grenoble Institute of Technology, and the European program EVOL of FP7 for their support.

6 REFERENCES

- [1] A. Laureau et al., "Coupled neutronics and thermal-hydraulics numerical simulations of a Molten Salt Fast Reactor (MSFR)," *Joint International Conference on Supercomputing in Nuclear Applications+ Monte Carlo (SNA & MC 2013)*, 2013.
- [2] S. Carney, F. Brown, B. Kiedrowski, and W. Martin, "Theory and Applications of the Fission Matrix Method for Continuous-Energy Monte Carlo," *Annals of Nuclear Energy*, **73**, pp. 423–431 (2014).
- [3] J. Dufek and W. Gudowski, "Fission Matrix Based Monte Carlo Criticality Calculations," *Annals of Nuclear Energy*, **36**, 8, pp. 1270–1275 (2009).
- [4] S. E. Carney, F. B. Brown, B. C. Kiedrowski, and W. R. Martin, "Fission Matrix Capability for MCNP Monte Carlo," Los Alamos National Laboratory (LANL) (2012).
- [5] J. Leppänen, "Serpent—a Continuous-energy Monte Carlo Reactor Physics Burnup Calculation Code," *VTT Technical Research Centre of Finland* (2012).
- [6] H. Jasak, A. Jemcov, and Z. Tukovic, "OpenFOAM: A C++ Library for Complex Physics Simulations," *International workshop on coupled methods in numerical dynamics*, volume 1000, pp. 1–20, 2007.
- [7] M. Brovchenko et al., "Design-Related Studies for the Preliminary Safety Assessment of the Molten Salt Fast Reactor," *Nuclear Science and Engineering*, **175**, pp. 329–39 (2013).
- [8] H. Rouch et al., "Preliminary thermal–hydraulic core design of the Molten Salt Fast Reactor (MSFR)," *Annals of Nuclear Energy*, **64**, pp. 449–456 (2014).