Time-dependent S_n method in the DOMINO solver of the Cocagne platform

Ansar Calloo, Angélique Ponçot and David Couyras

EDF R&D
7, boulevard Gaspard Monge, 91 120 Palaiseau, France
ansar.calloo@edf.fr, angelique.poncot@edf.fr, david.couyras@edf.fr

Abstract - *EDF R&D is currently working on a new, state-of-the-art calculation chain called ANDROMÈDE. For core computations, it uses the 3D Cartesian code,* Cocagne. Cocagne *is a platform which provides several engineering solutions as well as advanced solutions such as several solvers based on the SP_n (diffusion is a particular case of SP_1 equations) and S_n methods for the time-independent Boltzmann equation. The SP_1 kinetic equations have been implemented for diffusion calculations for engineering purposes. The goal of this paper is to derive and validate the time-dependent Boltzmann equation for* Cocagne. *The long-term aim of this Cartesian transport solver is to provide reference multigroup pin-homogenised kinetic solutions for the transport equation to be used as an industrial reference at EDF with the ANDROMÈDE calculation chain.*

I. INTRODUCTION

The statistical behaviour of a population of neutral particles is governed by the time-dependent Boltzmann equation. It is basically a balance equation that describes the rate of change of the population density between particles that stream into/out of a given control volume, those which disappear by absorption and scattering and those which are emitted by a source [1]. This equation is essential for modelling neutron transport in industrial nuclear reactors.

The solution to the time-dependent problem is still very costly, even today, and for everyday engineering applications, the Boltzmann equation is solved for a steady state by transforming the transient equation into a time-independent equation. The particle balance is ensured by dividing the production term by the *multiplication factor* or k_{eff}. Furthermore, for industrial applications, several approximations are set up to solve the equation in a reasonable amount of time.

However, studies of the neutron population in short time intervals, ranging from fraction of a second to several minutes such as safety engineering accident methodologies (*e.g.* reactivity insertion accident), may require the solution of the time-dependent equation. Given that $\psi(\boldsymbol{r}, E, \boldsymbol{\Omega}, t)$ and $C_p(\boldsymbol{r}, t)$ are respectively the neutron flux and the density of the p^{th} precursor group at time t, position $\boldsymbol{r}$, and for the flux at energy E in direction $\boldsymbol{\Omega}$ of the unit sphere S_2, such that:

$$
\begin{cases}
\mathcal{L}\psi(\boldsymbol{r}, E, \boldsymbol{\Omega}, t) = \boldsymbol{\Omega} \cdot \boldsymbol{\nabla}\psi(\boldsymbol{r}, E, \boldsymbol{\Omega}, t) + \Sigma_t(\boldsymbol{r}, E, t)\psi(\boldsymbol{r}, E, \boldsymbol{\Omega}, t) \\
\mathcal{H}\psi(\boldsymbol{r}, E, \boldsymbol{\Omega}, t) = \\
\dfrac{1}{2\pi}\displaystyle\int_0^\infty dE' \int_{S_2} d\boldsymbol{\Omega}'\Sigma_s(\boldsymbol{r}, E' \to E, \boldsymbol{\Omega} \cdot \boldsymbol{\Omega}', t)\psi(\boldsymbol{r}, E', \boldsymbol{\Omega}', t) \\
\mathcal{F}_{\mathrm{p}}\psi(\boldsymbol{r}, E, \boldsymbol{\Omega}, t) = \\
\dfrac{\chi_0(E)}{4\pi}\displaystyle\int_0^\infty dE' \int_{S_2} d\boldsymbol{\Omega}' \left[1 - \beta(E')\right] \nu\Sigma_f(\boldsymbol{r}, E', t)\psi(\boldsymbol{r}, E', \boldsymbol{\Omega}', t) \\
Q_r(\boldsymbol{r}, E, t) = \dfrac{1}{4\pi}\displaystyle\sum_{p=1}^{P} \lambda_p \chi_p(E) C_p(\boldsymbol{r}, t) \\
\mathcal{F}_{\mathrm{d},p}\phi(\boldsymbol{r}, E, t) = \beta_p(E)\nu\Sigma_f(\boldsymbol{r}, E, t)\phi(\boldsymbol{r}, E, t) \\
\phi(\boldsymbol{r}, E, t) = \displaystyle\int_{S_2} d\boldsymbol{\Omega}\psi(\boldsymbol{r}, E, \boldsymbol{\Omega}, t),
\end{cases}
$$

the time-dependent Boltzmann equation is expressed as:

$$
\frac{1}{v}\frac{\partial}{\partial t}\psi(\boldsymbol{r}, E, \boldsymbol{\Omega}, t) + \underbrace{\mathcal{L}\psi(\boldsymbol{r}, E, \boldsymbol{\Omega}, t)}_{\text{leakage and collisions}} - \overbrace{\mathcal{H}\psi(\boldsymbol{r}, E, \boldsymbol{\Omega}, t)}^{\text{transfers}} - \underbrace{\mathcal{F}_{\mathrm{p}}\psi(\boldsymbol{r}, E, \boldsymbol{\Omega}, t)}_{\text{prompt neutrons}} = \overbrace{Q_r(\boldsymbol{r}, E, t)}^{\text{delayed neutrons}} \qquad (1)
$$

and the P equations to account for the density of precursors that emit delayed neutrons are expressed as:

$$
\forall\, 1 \le p \le P, \quad \frac{\partial}{\partial t}C_p(\boldsymbol{r}, t) = \underbrace{-\lambda_p C_p(\boldsymbol{r}, t)}_{\text{losses}} + \overbrace{\int_0^\infty dE \mathcal{F}_{\mathrm{d},p}\phi(\boldsymbol{r}, E, t)}^{\text{productions}}. \qquad (2)
$$

At EDF, the operating core calculation chain CASSIOPEE solves this problem with the diffusion approximation. EDF R&D is currently working on a new, state-of-the-art calculation chain called *ANDROMÈDE*. For assembly computations, this encompasses the Apollo2 code/JEFF3-based CEA multigroup library/REL2005 scheme package [2], while for core computations, it uses the 3D Cartesian code, Cocagne [3]. Cocagne is a platform which provides several engineering solutions as well as advanced solutions such as several solvers based on the SP_n (diffusion is a particular case of SP_1 equations) and S_n methods for the time-independent Boltzmann equation [4]. The SP_1 kinetic equations have been implemented for diffusion calculations for engineering purposes.

Nonetheless, transport methods are becoming more and more prevalent in the nuclear engineering community, and the use of best-estimate calculations for validation is more common. Although high-fidelity calculations with heterogeneous cells in unstructured meshes (*e.g.* MoC) are becoming widespread, their industrial application on an everyday basis to deal with full-core industrial reactor for time-dependent applications (*e.g.* RIA) is still not available. At EDF, within the Cocagne platform, there exists a framework to deal with core calculations from homogeneous assemblies to pin-resolved calculations for diffusion up to full transport calculations. The goal of this paper is to derive and validate the time-dependent Boltzmann equation for Cocagne. The aim of this Cartesian transport solver is to provide reference multigroup and pin-homogenised kinetic solutions for the transport equation for its use as an industrial reference for EDF.

This paper first describes the theoretical background for the time-dependent transport equation along with its numerical integration using the θ-scheme. Furthermore, a preliminary method to optimize the solver will also be discussed in this section. Afterwards, we shall provide the numerical time convergence of the solver using a benchmark case, as well as validation results where the power of the core is compared to published solutions. Besides, we have also set up a small benchmark with multigroup cross sections for further validation purposes.

II. THEORY

This section describes the equations which have been implemented in Cocagne for the time-dependent S_n solver.

1. Brief overlook of the existing $\mathrm{S_n}$ solver, DOMINO

Today, the Cocagne platform has a steady-state S_n solver called DOMINO. DOMINO is based on a classical multigroup approach for the energy variable. The angular flux is solved using *Level-Symmetric Quadratures* ranging from S_2 to S_{16}. DOMINO solves the spatial part using a *diamond-differencing scheme* which is implemented for DD_0 only at present.

Past works have shown that for PWR cases S_8 quadratures are largely efficient and satisfactory. Furthermore, DD_0 spatial discretization is also acceptable for our in-house studies since they are carried at pin-cell level which are further refined up to 2×2 meshes and by applying an SPH equivalence factor.

Besides, for convergence and acceleration purposes, DOMINO uses a *Diffusion Synthetic Acceleration* scheme based on the SP_1 solver which is implemented within the same platform.

2. Principles of the θ-scheme

The time-dependent neutron transport equation is integrated using the θ-scheme method. Let us consider the following time-dependent problem, where f is positive and defined on the $[0, T]$ range:

$$\begin{cases} \frac{\partial}{\partial t}u + \alpha(t)u(t) = f(t), \\ u(0) = u_0 \end{cases}$$

It is integrated from t_n to t_{n+1} as follows:

$$u(t_{n+1}) - u(t_n) + \int_{t_n}^{t_{n+1}} \alpha(t)u(t)dt = \int_{t_n}^{t_{n+1}} f(t)dt.$$

The θ-scheme is based on the following approximation:

$$\forall\theta \in [0;1], \quad \int_{t_n}^{t_{n+1}} f(t)dt \approx \int_{t_n}^{t_{n+1}} \left[f(t_n)w_n^\theta(t) + f(t_{n+1})w_{n+1}^\theta(t)\right] dt$$

where the functions $w_n^\theta(t)$ and $w_{n+1}^\theta(t)$ are quadratic polynomials defined as:

$$\begin{aligned} w_n^\theta(t) &= \frac{6(\theta - 1/2)}{\Delta t^2}(t_{n+1} - t)^2 + \frac{6}{\Delta t}(2/3 - \theta)(t_{n+1} - t) \\ w_{n+1}^\theta(t) &= \frac{6(1/2 - \theta)}{\Delta t^2}(t - t_n)^2 + \frac{6}{\Delta t}(\theta - 1/3)(t - t_n) \end{aligned}$$

and such that:

$$\begin{cases} w_n^\theta(t_n) = 1, \quad w_n^\theta(t_{n+1}) = 0 \quad \text{et} \displaystyle\int_{t_n}^{t_{n+1}} w_n^\theta(t)\, dt = (1-\theta)\Delta t, \\ w_{n+1}^\theta(t_n) = 0, \quad w_{n+1}^\theta(t_{n+1}) = 1 \quad \text{et} \displaystyle\int_{t_n}^{t_{n+1}} w_{n+1}^\theta(t)\, dt = \theta\Delta t. \end{cases}$$

Thus, the integral is approximated as follows:

$$\int_{t_n}^{t_{n+1}} f(t)dt \approx \left[\theta f(t_{n+1}) + (1-\theta)f(t_n)\right]\Delta t,$$

thereby, leading to the time scheme expressed as:

$$\forall\theta \in]0;1], \quad \frac{u_{n+1} - u_n}{\theta\Delta t} + \left[\alpha(t_{n+1})u_{n+1} + \frac{1-\theta}{\theta}\alpha(t_n)u_n\right] = f(t_{n+1}) + \frac{1-\theta}{\theta}f(t_n), \quad (3)$$

where u_n and u_{n+1} are the approached values for $u(t_n)$ and $u(t_{n+1})$.

3. Integration of the precursor density equation

The precursor concentration equation is solved by an exact integration of equation (2). Integrating that equation by parts over $[t_n; t_{n+1}]$ leads to:

$$\frac{\partial d}{\partial t}(C_p(\boldsymbol{r}, t)\exp^{-\lambda_p(t_{n+1}-t)}) = \int_{t_n}^{t_{n+1}} \int_0^{+\infty} \mathcal{F}_{\mathrm{d},p}\psi(\boldsymbol{r}, E, \boldsymbol{\Omega}, t)\exp^{-\lambda_p(t_{n+1}-t)}\, dEdt. \quad (4)$$

The θ-scheme is not used for the precursor equation, it is integrated analytically as:

$$C_p(\boldsymbol{r}, t_{n+1}) = C_p(\boldsymbol{r}, t_n)\exp^{-\lambda_p\Delta t} + W_n^p\mathcal{F}_{\mathrm{d},p}\psi(\boldsymbol{r}, E, \boldsymbol{\Omega}, t_n) + W_{n+1}^p\mathcal{F}_{\mathrm{d},p}\psi(\boldsymbol{r}, E, \boldsymbol{\Omega}, t_{n+1}), \quad (5)$$

with

$$W_n^p\mathcal{F}_{\mathrm{d},p}\psi(\boldsymbol{r}, E, \boldsymbol{\Omega}, t_n) + W_{n+1}^p\mathcal{F}_{\mathrm{d},p}\psi(\boldsymbol{r}, E, \boldsymbol{\Omega}, t_{n+1}) = \int_{t_n}^{t_{n+1}} \exp^{-\lambda_p(t_{n+1}-s)}\mathcal{F}_{\mathrm{d},p}\psi(\boldsymbol{r}, E, \boldsymbol{\Omega}, s)ds,$$

and

$$\begin{aligned} W_n^p &= \int_{t_n}^{t_{n+1}} \exp^{-\lambda_p(t_{n+1}-s)} w_n^\theta(s)\, ds, \\ W_{n+1}^p &= \int_{t_n}^{t_{n+1}} \exp^{-\lambda_p(t_{n+1}-s)} w_{n+1}^\theta(s)\, ds. \end{aligned}$$

4. Integration of the neutron transport equation

Integrating equation (1) over $[t_n; t_{n+1}]$ results into:

$$\frac{1}{v}\left(\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t_{n+1})-\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t_n)\right) + \int_{t_n}^{t_{n+1}}\left(\mathcal{L}-\mathcal{H}-\mathcal{F}_\mathrm{p}\right)\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t)dt = \int_{t_n}^{t_{n+1}} Q_r(\boldsymbol{r},E,t)dt. \quad (6)$$

The θ-scheme (3) is applied to the previous equation, thereby leading to:

$$\frac{\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t_{n+1})-\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t_n)}{v\theta\Delta t} + \left(\mathcal{L}-\mathcal{H}-\mathcal{F}_\mathrm{p}\right)\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t_{n+1}) + \frac{1-\theta}{\theta}\left(\mathcal{L}-\mathcal{H}-\mathcal{F}_\mathrm{p}\right)\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t_n) = Q_r(\boldsymbol{r},E,t_{n+1}) + \frac{1-\theta}{\theta}Q_r(\boldsymbol{r},E,t_n). \quad (7)$$

The expression for the precursor density at times t_n and t_{n+1} and the following notations are introduced for simplification:

$$\begin{cases} \widetilde{\mathcal{F}}^{n+1}_{d,p}\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t_{n+1}) & = & \frac{1}{4\pi}\lambda_p\chi_p(E)W^p_{n+1}\mathcal{F}_{d,p}\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t_{n+1}), \\ \widetilde{\mathcal{F}}^{n}_{d,p}\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t_{n}) & = & \frac{1}{4\pi}\lambda_p\chi_p(E)W^p_{n}\mathcal{F}_{d,p}\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t_{n}). \end{cases}$$

Thus, the following source problem has to be solved at t_{n+1}:

$$\left(\frac{1}{v\theta\Delta t}+\mathcal{L}-\mathcal{H}\right)\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t_{n+1}) = Q(\boldsymbol{r},E,\boldsymbol{\Omega},t_n) + \left(\mathcal{F}_p+\sum_{p=1}^{P}\widetilde{\mathcal{F}}^{n+1}_{d,p}\right)\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t_{n+1}), \quad (8)$$

with

$$Q(\boldsymbol{r},E,\boldsymbol{\Omega},t_n) = \frac{1}{4\pi}\sum_{p=1}^{P}\lambda_p\chi_p(E)\left(\frac{1-\theta}{\theta}+\exp^{-\lambda_p\Delta t}\right)C_p(\boldsymbol{r},t_n) + \left\{\frac{1}{v\theta\Delta t}+\frac{\theta-1}{\theta}\left(\mathcal{L}-\mathcal{H}-\mathcal{F}_p\right)+\sum_{p=1}^{P}\widetilde{\mathcal{F}}^{n}_{d,p}\right\}\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t_n).$$

If $\theta = 1$, the time scheme thus obtained is an implicit Euler scheme which is unconditionally stable. For $\theta = 0.5$, the θ scheme yields the Crank-Nicolson scheme [5] which converges with a second order scheme precision in time. The Crank-Nicolson scheme is unconditionally stable and is implicit in time. The explicit integration scheme is obtained for $\theta = 0.$. However, although it is more easily implemented, the explicit scheme does not converge unconditionally since it requires very small time steps, below 10^{-10} s using the CFL condition [6], for stability. Thus, it is not very useful in practice for the time-dependent neutron transport equation and we shall not use this integration scheme in DOMINO (thereby, allowing for θ in the denominator term in the previous equations).

5. Time resolution scheme

The goal is to solve the coupled Boltzmann equation discretized in time with the θ-scheme along with the precursor equation at each time step. This coupled problem is initialized by solving the time-independent problem to obtain the initial equilibrium flux. The latter is used to initialize the equilibrium precursor concentration (equation 2 without the time derivative).

The quantities C_p and ψ are then computed at time t_{n+1} using the following algorithm:

- calculation of W^p_n and W^p_{n+1}, and the source term $Q(\boldsymbol{r},E,\boldsymbol{\Omega},t_n)$,
- flux calculation at time t_{n+1}, which is solution to the source problem (equation 8),
- calculation of the precursor concentration at t_{n+1}.

As far as the spatial and angular sweep are concerned, the kinetic solver hinges on the same approaches as for the steady-state DOMINO solver. Thus, only a few terms are changed to allow for the time-dependence of cross sections: the source term is computed with the updated precursor concentration at times t_n and t_{n+1}. In the present implementation, the angular flux at time step t_n are stored to compute the source at step t_{n+1}, and thus will be costly in terms of the memory peak for very large cases. The next section proposes a simple optimization strategy for this particular problem.

6. Preliminary optimization

In this section, we shall consider a preliminary numerical optimization of the solver. The goal is to use a low-order approximation of the angular flux time derivatives as proposed by [7].

Let us consider equation 8. Part of the angular source term from that equation can be cancelled for the t_n term if an implicit scheme ($\theta = 1$) is applied.

The angular flux is expanded on a basis of appropriate functions. For that purpose, they are projected on the basis of real spherical harmonics as described in [1]:

$$\begin{cases} \psi(\boldsymbol{r},E,\boldsymbol{\Omega},t) = \sum_{l=0}^{L}\frac{2l+1}{4\pi}\sum_{m=-l}^{m=l}\phi_{lm}(\boldsymbol{r},E,t)R_{lm}(\boldsymbol{\Omega}), \\ \phi_{lm}(\boldsymbol{r},E,t) = \int_{S_2} d\boldsymbol{\Omega}' R_{lm}(\boldsymbol{\Omega}')\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t). \end{cases}$$

Afterwards the time derivative given by:

$$\frac{1}{v}\frac{\partial}{\partial t}\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t)$$

is subsituted by the following expression

$$\frac{1}{v}\sum_{l=0}^{L}\frac{2l+1}{4\pi}\sum_{m=-l}^{m=l}R_{lm}(\boldsymbol{\Omega})\frac{\partial}{\partial t}\phi_{lm}(\boldsymbol{r},E,t).$$

and thus leads to the numerical scheme given by the equation written:

$$\begin{aligned}&\left\{\mathcal{L}-\mathcal{H}-\mathcal{F}_p-\sum_{p=1}^{P}\widetilde{\mathcal{F}}_{d,p}^{n+1}\right\}\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t_{n+1})\\&+\tfrac{1}{v\theta\Delta t}\textstyle\sum_{l=0}^{L}\tfrac{2l+1}{4\pi}\sum_{m=-l}^{m=l}\phi_{lm}(\boldsymbol{r},E,t_{n+1})R_{lm}(\boldsymbol{\Omega})\\&=\sum_{p=1}^{P}\widetilde{\mathcal{F}}_{d,p}^{n}\phi(\boldsymbol{r},E,t_n)\\&+\tfrac{1}{v\theta\Delta t}\textstyle\sum_{l=0}^{L}\tfrac{2l+1}{4\pi}\sum_{m=-l}^{m=l}\phi_{lm}(\boldsymbol{r},E,t_n)R_{lm}(\boldsymbol{\Omega})\\&+\tfrac{1}{4\pi}\textstyle\sum_{p=1}^{P}\lambda_p\chi_p(E)\exp^{-\lambda_p\Delta t}C_p(\boldsymbol{r},t_n)\end{aligned}$$

Therefore, at time t_{n+1}, the following source problem is obtained:

$$\begin{aligned}&(\mathcal{L}-\mathcal{H})\,\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t_{n+1})\\&+\frac{1}{v\theta\Delta t}\sum_{l=0}^{L}\frac{2l+1}{4\pi}\sum_{m=-l}^{m=l}R_{lm}(\boldsymbol{\Omega})\phi_{lm}(\boldsymbol{r},E,t_{n+1})=\\&\left(\mathcal{F}_p+\sum_{p=1}^{P}\widetilde{\mathcal{F}}_{d,p}^{n+1}\right)\psi(\boldsymbol{r},E,\boldsymbol{\Omega},t_{n+1})\\&+Q(\boldsymbol{r},E,t_n),\qquad(9)\end{aligned}$$

where the angular source at time t_n is defined by the following expression:

$$\begin{aligned}Q(\boldsymbol{r},E,\boldsymbol{\Omega},t_n)&=\sum_{p=1}^{P}\widetilde{\mathcal{F}}_{d,p}^{n}\phi(\boldsymbol{r},E,t_n)\\&+\frac{1}{v\theta\Delta t}\sum_{l=0}^{L}\frac{2l+1}{4\pi}\sum_{m=-l}^{m=l}\phi_{lm}(\boldsymbol{r},E,t_n)R_{lm}(\boldsymbol{\Omega})\\&+\frac{1}{4\pi}\sum_{p=1}^{P}\lambda_p\chi_p(E)\exp^{-\lambda_p\Delta t}C_p(\boldsymbol{r},t_n).\qquad(10)\end{aligned}$$

This term has the corresponding form

$$A\Psi=F\Psi+Q,$$

and is usually solved using the Gauss-Seidel algorithm:

$$A\Psi^{k+1}=F\Psi^{k}+Q.$$

In the rest of this paper, the Gauss-Seidel iterations are called "external iterations". For each of the latter, the following source problem should be solved (by dropping the time indices, and expressing the problem as a multigroup form:

$$\begin{aligned}&\boldsymbol{\Omega}\cdot\boldsymbol{\nabla}\psi_g(\boldsymbol{r},\boldsymbol{\Omega})+\Sigma_{tg}(\boldsymbol{r})\psi_g(\boldsymbol{r},\boldsymbol{\Omega})=\\&\sum_{g'=1}^{G}\frac{2l+1}{4\pi}\sum_{l=0}^{L_\Sigma}\Sigma_{sl}^{g'\to g}(\boldsymbol{r})\sum_{m=-l}^{m=l}\phi_{g',lm}(\boldsymbol{r})R_{lm}(\boldsymbol{\Omega})\\&-\tfrac{1}{v\theta\Delta t}\textstyle\sum_{l=0}^{L_\phi}\tfrac{2l+1}{4\pi}\sum_{m=-l}^{m=l}R_{lm}(\boldsymbol{\Omega})\phi_{g,lm}(\boldsymbol{r})=Q(\boldsymbol{r},\boldsymbol{\Omega}).\end{aligned}$$

where L_Σ corresponds to the expansion order of the differential scattering cross section and L_ϕ that of the angular flux for approximating the time terms. These two orders may not be strictly the same, even though usually $L_\phi = L_\Sigma$, to avoid increasing the memory peak compared to a static problem.

This method has been implemented recently in the DOMINO solver and is currently under validation. Results on the LMW3D benchmark are very satisfactory and yield a discrepancy of 0.01% compared to the exact method using a P_1 expansion of the angular flux. Future work requires the testing of the method of more anisotropic cases with UOX-MOX interfaces to ensure that the P_1 expansion is sufficient even for more heterogeneous situations.

III. VALIDATION ON LMW3D

This section higlights the verification and validation work on the newly-implemented time-dependent S_n solver. Firstly verification tests were done to prove the very good performances of this solver for unitary tests. Secondly a benchmark published by the end of the 1970's by Langenbuch, Maurer and Werner in [8] was studied.

1. The LMW3D benchmark

This benchmark hereafter called *LMW3D* is based on a 3D geometry representing a small core with two rodbanks that are being simultaneously inserted and removed over a time interval of 60 seconds. [1] The test geometry is that of a quarter core as shown in Figure 1 and the positions of the rodbanks are described in the same figure on the right.

The neutronic data for this geometry are provided in table I. The latter correspond to the cross sections for diffusion solvers: transport solvers such as DOMINO in Cocagne do not use absorption or slowing-down cross sections $\Sigma_{1\to2}$. Thus, the total and scattering cross sections are defined as:

$$\left\{\begin{array}{ll}\Sigma_{t1}=\frac{1}{3D_1}, & \Sigma_{t2}=\frac{1}{3D_2},\\ \Sigma_s^{1\to1}=\frac{1}{3D_1}-\Sigma_{a1}-\Sigma_r, & \Sigma_s^{2\to1}=0,\\ \Sigma_s^{1\to2}=\Sigma_{1\to2}, & \Sigma_s^{2\to2}=\frac{1}{3D_1}-\Sigma_{a1}.\end{array}\right.$$

Furthermore, the paper [8] also provides power values for:

- the total core power with time,
- the local power in the three coloured cells of Figure 1 with time.

The solution from [8] constitutes our "reference" solution based on a diffusion approximation calculation with a semi-implicit time discretisation. Their spatial discretisation is based on the nodal elements with polynomials of order 3 (for a mesh with 20 cm cells).

In this work, the newly-implemented transient capabilities of DOMINO will be tested on this benchmark and compared to the diffusion (SP_1) solver of Cocagne. These results will also be compared to the reference calculations of [8] as well as those obtained by Mula in [9]. This benchmark is used for elementary validation purposes as cross sections are constant

[1] It is often (wrongly) referred to as the TWIGL benchmark, which is a 2D test with two rods which are inserted in a core.

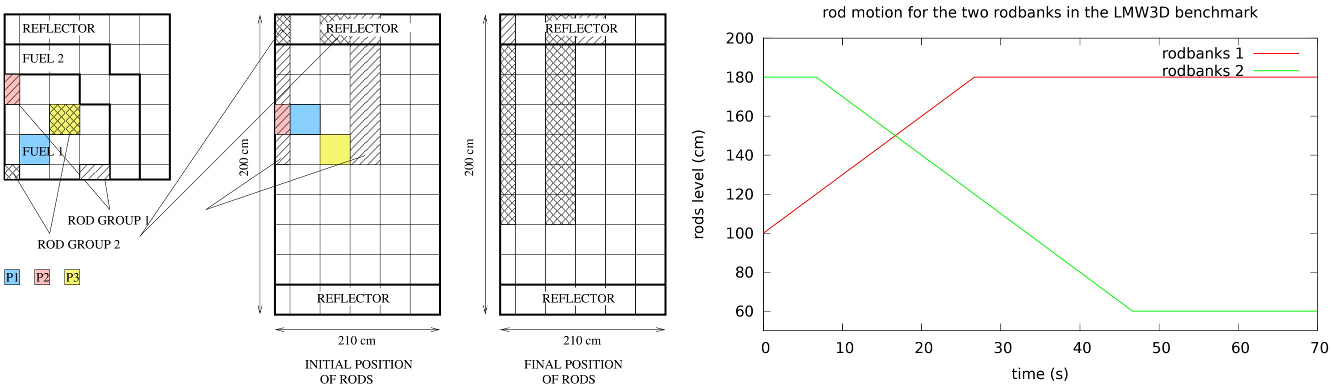

Fig. 1. The 3D geometry with a quarter core for the LMW3D benchmark (left). The colored cells represent those for which the authors of the initial paper provide local power values. Rod motion for the two rodbanks in the LMW3D benchmark. The y-axis corresponds to the number of extracted rod steps, each step being equal to 1 cm (right).

Region	Fuel 1	Fuel 2	Refl
D_1	1.423913	1.423913	1.425611
D_2	0.3563060	0.3563060	0.3505740
Σ_{a1}	1.040206e-2	1.095206e-2	1.099263e-2
Σ_{a2}	8.766217e-2	9.146217e-2	9.925634e-2
$\nu\Sigma_{f1}$	6.477691e-3	6.477691e-3	7.503284e-3
$\nu\Sigma_{f2}$	1.127328e-1	1.127328e-1	1.378004e-1
$\Sigma_{1\to2}$	0.0175555	0.0175555	0.01717768

Precursor groups	β_i	λ_i
group 1	0.000247	0.0127
group 2	0.0013845	0.0317
group 3	0.001222	0.115
group 4	0.0026455	0.311
group 5	0.000832	1.4
group 6	0.000169	3.87

$v_1 = 1.25 \cdot 10^7\ cm/s$, $v_2 = 2.5 \cdot 10^5\ cm/s$
$\kappa = 3.204 \cdot 10^{-11}\ J$,
$\nu = 2.5$ neutrons/fission

TABLE I. Neutronic data for LMW3D.

as there are no thermalhydraulic feedback, thereby leading to results that can be analysed more easily.

It should be noted that in the benchmark, the rods are inserted/removed at a speed of 3 cm/s. Thus, it must be ensured that the rod positions always correspond to a defined cell in the geometry mesh to avoid rod cusping effects [10][2]. Each non-refined axial cell has a height of 20 cm. Thus, the time steps are chosen such that $\Delta t = \dfrac{20}{3 \times N_t}$, where N_t is a refinement factor. Hence, the axial cells are refined by $20/N_z$ with $N_t \leq N_z$. The S_4 quadrature is used for all S_n calculations with $N_z = 24$.

2. Time convergence

A convergence study of the solvers is first carried out to analyse the time convergence of the S_n and SP_1 solvers on this benchmark for $\theta = 0.5$ (Crank-Nicolson scheme) and $\theta = 1$ (implicit scheme). The time step is varied from 6.66 s to 0.28 s. For the convergence studies, the calculation with time step 0.28 s is used as the reference calculation. Figure 2 shows the time convergence for the total core power.

From Figure 2, it can be observed that for both the Crank-Nicolson and implicit schemes, the solution converges as the time step is decreased, thereby leading to smaller discrepancies. The solution is assumed to be sufficiently converged with a time step below 1 s, since the discrepancies are under 1%. The solution for the Crank-Nicolson scheme converges faster than that of the implicit scheme since the errors are below 1% for a time step of 0.83 s. Indeed, the Crank-Nicolson scheme is a second order scheme in time and hence, converges faster than the implicit scheme (order one in time). The latter provides errors below 1% for time steps below 0.83 s. The same convergence study has been carried out for the diffusion solver with the same conclusions and will not be presented in this paper.

[2] The SP_1 kinetic solver has strategies implemented for such cases, *e.g.* on-the-fly mesh creation. However, such strategies has not been defined yet for the DOMINO solver

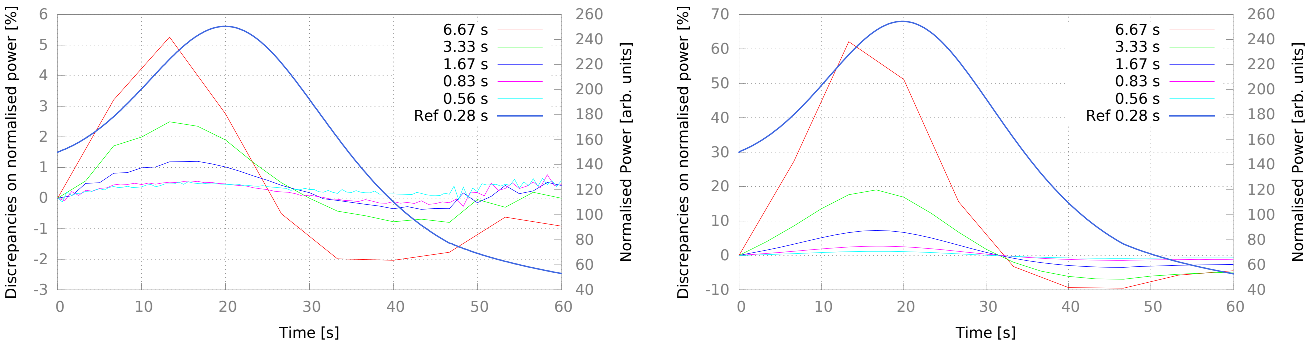

Fig. 2. Time convergence results for the LMW benchmark with the DOMINO solver for both the Crank-Nicolson scheme (left) and the implicit scheme (right).

Figure 3 shows the total core power variation over the simulation time range (60 s) for both the Crank-Nicolson and implicit schemes. Both schemes with both DOMINO and the diffusion solvers have the same behaviour and with very small differences for the peak amplitude (less than 1.5%). This step provides a preliminary verification of the kinetic DOMINO solver.

3. Comparison to published solutions

These results are then compared to the reference values provided in [8] and in [9]. The results for DOMINO will be compared for $\Delta t = 0.28$ s. Figure 4 illustrates the core power for the various Cocagne calculations and reference values. For the initial 18 s, all the plots are similar. However, afterwards, the Cocagne solvers have a lower maximum and thus, the power levels are below the reference values [8]. Yet, if compared to the recent results obtained by [9], the Cocagne solvers have a similar trend. Local power values have also been compared and the same conclusions can be affirmed for positions P1, P2 and P3 as shown on the Figure 1.

Bearing in mind that the time-dependent DOMINO solver has not been fully optimised, the running time for the execution of the solution compared here is 8 000 s whereas the diffusion calculation time is 600 s. Although this is a very high running time, it corresponds to a refined spatial case where the axial direction has been meshed finely to avoid rod-cusping effects. Nevertheless, to tackle industrial problems, numerical strategies as well as parallel computational considerations are further required.

Since the Cocagne results are very close to those published in [8, 9], it can be assumed that the DOMINO solver behaves satisfactorily for this benchmark. In the next section, we will present further validation work which has been carried out for a multigroup case, which is based on the same type of transient as the LMW3D benchmark.

IV. VALIDATION FOR A MULTIGROUP CASE

In this section, we consider a 5 × 5 cluster of UOX assemblies with a 3.7% enrichment. This case is set up as the LMW3D benchmark with two rodbanks which are moved correspondingly, *i.e.*, one of them is removed, R1, as the other, R2, is being inserted. The benchmark geometry is given in Figure 5. No reflector assemblies are used in this benchmark where the boundary conditions are set as reflective ones.

The cross sections are generated using the Gabv2 tool which implements the Apollo2.8-REL2005 computation scheme [2]. Thus, for a given assembly, the cross sections will be generated at two and eight energy groups as for previous works [11]. The goal of this section is to examine the impact of the energy discretisation on the kinetic calculation. Thus, it will be possible to compute two-group diffusion calculation up to multigroup transport ones on a homogenised assembly. As for the previous studies cited, diffusion calculation are carried without any equivalence factor but with Selengut normalisation factors for the two-group calculation. On the other hand, the multigroup calculations use flux-volume normalisation factors [12].

Figure 6 illustrates the power released in the core as previously shown for LMW3D. First, it is observed that there is a multigroup effect by increasing the number of groups while keeping the diffusion approximation. Although the initial behavior during the initial 15 s are very similar, the multigroup diffusion case has a higher peak that the diffusion calculation. Secondly, the S_n DOMINO solver is applied for the eight-group calculation. From this calculation, it can be observed that for a given spatial mesh and energy discretization, using the transport solver leads to significantly lower peaks. Besides, it can also be noted that the power peaks are slightly shifted in time for the different simulations.

These results are very interesting and promising as for exactly the same problem (geometry, materials and nuclear data), both the transport and diffusion solutions for different energy groups - with corresponding cross sections generated by Gabv2 - are available. Indeed, literature is rife with two-group diffusion kinetic problems on structured meshed geometries or fully heterogeneous kinetic transport problems. Yet, the comparison of different solvers on a given problem set for a time-dependent situation is not available.

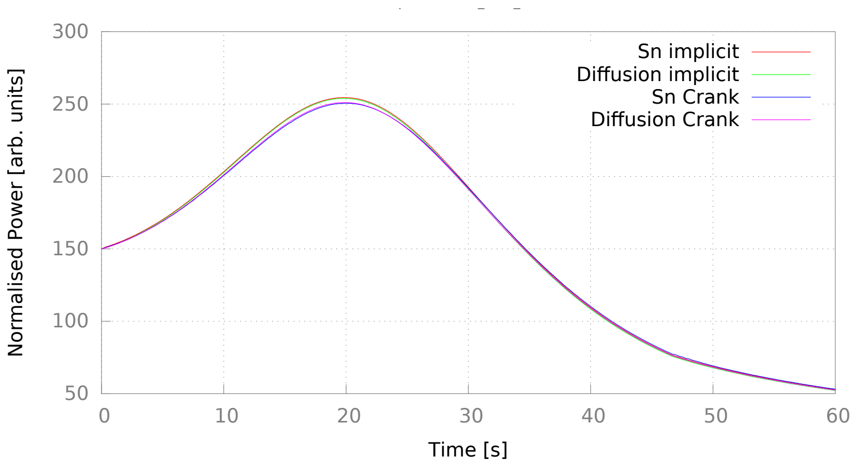

Fig. 3. Core power as computed with the Crank-Nicolson and implicit schemes for the S_4 and SP_1 solvers (N_z = 24, Δt = 0.28 s).

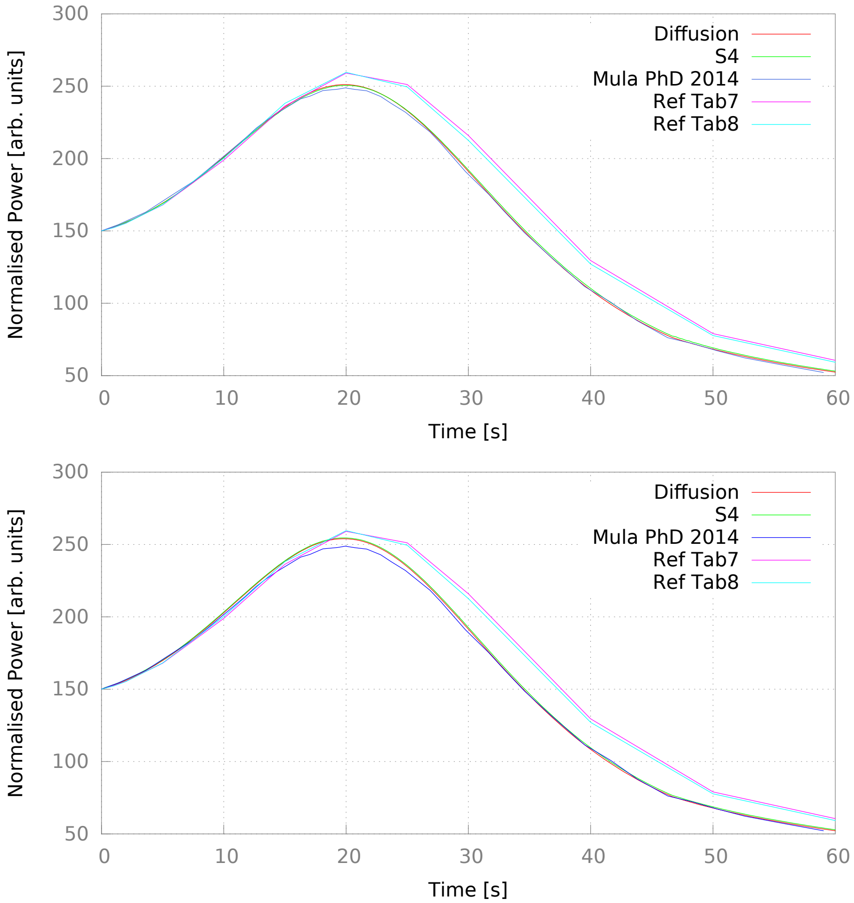

Fig. 4. Core power computed by diffusion (SP_1), S_4 and the references for the LMW benchmark (N_z = 24, Δt = 0.28 s, with both the Crank-Nicolson scheme with θ = 0.5 and the implicit scheme with θ = 1 (below) for Cocagne solvers).

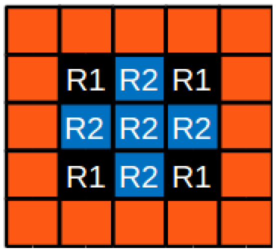

Fig. 5. The geometry for the 5 × 5 cluster of UOX assemblies. The assemblies with R1 and R2 are rodded.

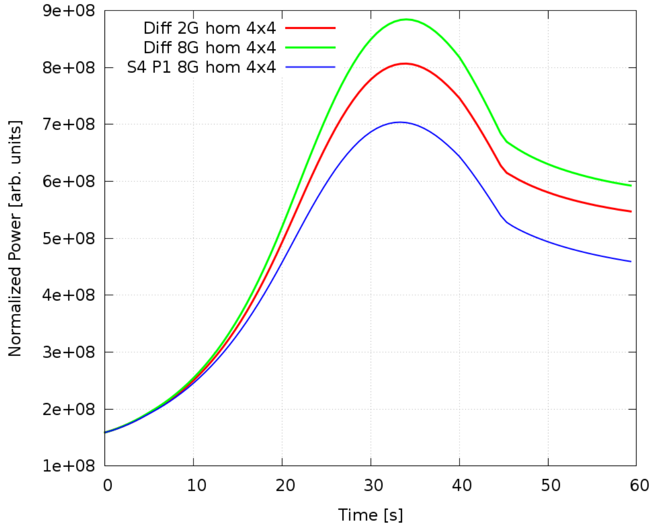

Fig. 6. Core power computed by diffusion (SP_1) and S_4 ($N_z = 15$, $\Delta t = 0.67$ s, with the Crank-Nicolson scheme for the two-group and 8-group cases. For all cases, it is ensured that the problem is converged spatially.

V. CONCLUSIONS

The time-dependent S_n method has been successfully implemented within the DOMINO framework of the Cocagne platform. Verification tests have proved the very good performances of this solver for unitary cases. The solver has also been optimized for memory requirements using low-order approximations for angular flux terms as described in the previous section.

Furthermore, the LMW benchmark provided preliminary validation of the solver. The results were compared to those obtained by [8] and [9]. It showed that both the implicit and the Crank-Nicholson schemes are satisfactory on this case and that the time-dependent DOMINO solver yields good results. Further validation work has been carried out on an in-house case with a cluster of assemblies whereby two-group cross sections for diffusion problem as well as the corresponding multigroup cross sections for homogenised transport problems are available. Results have shown that transport solution gives much lower power peaks than diffusion solutions, and that these are slightly shifted in time. Future work on such validation cases includes the calculation of the given problem with pin-homogenised problems for both the transport and diffusion solvers with Cocagne. The long-term aim of this Cartesian transport solver is to provide reference multigroup pin-homogenised kinetic solutions for the transport equation as an industrial reference at EDF with the ANDROMÈDE calculation chain.

REFERENCES

1. HEBERT, A., *Applied Reactor Physics*, Presses Internationales Polytechnique (2009).
2. VIDAL, J.-F., LITAIZE, O., BERNARD, D., ET AL., “New modelling of LWR assemblies using the APOLLO2 code package,” in “Proceedings of Joint International Topical Meeting on Mathematics and Computations and Supercomputing in Nuclear Applications, M & C + SNA,” Monterey, CA (April 2007).
3. T. COURAU, S. MOUSTAFA, L. PLAGNE, and A. PONÇOT, “DOMINO: A fast 3D cartesian discrete ordinates solver for reference PWR simulations and SPN validation,” in “International Conference on Mathematics and Computational Methods Applied to Nuclear Science & Engineering,” USA (May 2013).
4. CALLOO, A., COUYRAS, D., FÉVOTTE, F., GUILLO, M., “Cocagne: EDF new neutronic core code for ANDROMEDE calculation chain,” in “Proceedings of *this conference*,” Jeju, Korea (2017).
5. CRANK, J., NICOLSON P. , “A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type,” *Cambridge Philosophical Society*, **43**, 50–67 (1947).
6. COURANT, R., FRIEDRICHS, K., LEWY, H. , “On the Partial Difference Equations of Mathematical Physics,” *Mathematische Annalen*, **100**, 32–74 (1928).
7. HOFFMAN, A. J. AND LEE, J. C. , “Low-Order Approximations to the Angular Flux Time Derivative for Transport-based Reactor Kinetics,” *Trans. Am. Nucl. Soc.*, **108**, 777–780 (2013).
8. LANGENBUCH, S., MAURER W. AND WERNER W., “Coarse Mesh Flux-Expansion Method for the Analysis of Space-Time Effects in Large Light Water Reactor Cores,” *Nucl. Sci. Eng.*, **63**, 437–456 (1977).
9. MULA, O., *Some contributions towards the parallel simulation of time-dependent neutron transport and the integration of observed data in real time*, Ph.D. thesis, Université Pierre et Marie Curie (2014).
10. JOO, H-S., *Resolution of the control rod cusping problem for nodal methods*, Ph.D. thesis, Massachussetts Institute of Technology (1984).
11. CALLOO, A., HUY, S., COUYRAS, D., BROSSELARD, C., FLISCOUNAKIS, M., “Validation of the SP_n depletion schemes of the EDF Gabv2-Cocagne tools using the KAIST 1A benchmark,” in “Proceedings of PHYSOR 2016,” Sun Valley, Idaho, USA (May 2016).
12. BROSSELARD, C. AND LEROYER, H. AND GIRARDI, E. AND FLISCOUNAKIS, M., “Normalization methods for diffusion calculations with various assembly homogenizations,” in “Proceedings of PHYSOR 2014,” Kyoto, Japan (2014).