Adaptive solution of the domain decomposition+L^2-jumps method applied to the neutron diffusion equation on structured meshes

Mario Gervais[1,*], *François* Madiot[1,**], *Minh-Hieu* Do[1,***], and *Patrick* Ciarlet Jr.[2,****]

[1]Université Paris-Saclay, CEA, Service d'Études des Réacteurs et de Mathématiques Appliquées, 91191, Gif-sur-Yvette, France.
[2]POEMS, CNRS, INRIA, ENSTA Paris, Institut Polytechnique de Paris, 91120 Palaiseau, France.

Abstract. At the core scale, neutron deterministic calculations are usually based on the neutron diffusion equation. Classically, this equation can be recast in a mixed variational form, and then discretized by using the Raviart-Thomas-Nédélec Finite Element. The goal is to extend the Adaptive Mesh Refinement (AMR) strategy previously proposed in [1] to the Domain Decomposition+L^2 jumps which allows non conformity at the interface between subdomains. We are able to refine each subdomain independently, which eventually leads to a more optimal refinement. We numerically investigate the improvements made to the AMR strategy.

1 Introduction

Numerical simulation of the neutron transport equation usually requires a high computational cost. The reason comes from the fact that one has to deal with many variables: the neutron position in space, the velocity direction and the neutron energy. In industrial applications, calculations at core scale are usually modeled by the neutron diffusion equation. Our context is the development of the project APOLLO3®, a shared platform among CEA, EDF and FRAMATOME, which includes different deterministic solvers in order to perform lattice and core calculations [2]. Particularly, we are interested in the MINOS solver based on the mixed Raviart-Thomas-Nédélec finite element method and implemented on Cartesian and hexagonal grids for the multi-group SP_N equation [3]. In this work, we focus on the steady state monokinetic neutron diffusion equation on a reactor core $\mathcal{R}$,

$$\begin{cases} \mathbb{D}^{-1}\mathbf{p} + \nabla\phi = 0 \quad \text{in } \mathcal{R}, \\ \operatorname{div} \mathbf{p} + \Sigma_a\phi = \dfrac{1}{k_{\text{eff}}}\nu\Sigma_f\phi \quad \text{in } \mathcal{R}, \\ \phi = 0 \quad \text{on } \partial\mathcal{R}, \end{cases} \tag{1}$$

where $\mathbb{D}$ is the diffusion coefficient, Σ_a is the absorption cross section, Σ_f is the fission cross section, ν the average number of neutrons emitted per fission, $\mathbf{p}$ is the current, ϕ is the flux,

[*]e-mail: mario.gervais@cea.fr
[**]e-mail: francois.madiot@cea.fr
[***]e-mail: minh-hieu.do@cea.fr
[****]e-mail: patrick.ciarlet@ensta-paris.fr

and k_{eff} is the effective multiplicative factor. Since the cross sections and the diffusion coefficients are usually heterogeneous (piecewise constant), the solution of the diffusion equation may have some singularities which limit the precision and convergence of the solution [4]. It is well known that the mesh subdivision method is one of the most effective ways to treat this problem [1, 5–7].

The domain decomposition is a strategy that is well-suited for parallel computing. It allows to split the problem posed on the whole domain into subproblems posed in subdomains, which may be solved in parallel. Among those domain decomposition methods, the domain decomposition+L^2-jumps method (or DD+L^2-jumps method) is proposed in [4]. The authors show, under some assumptions, the convergence of this method for a non-conforming decomposition. We mention [8] for a successful application of the DD+L^2-jumps method to criticality calculations. In the framework of reactor core simulations, we are interested in Adaptive Mesh Refinement (AMR) strategies. Suitably combined to the DD+L^2-jumps method, a posteriori error estimates allow to adapt independently the mesh of each subdomain.

The purpose of this work is to develop an adaptive solution of the DD+L^2-jumps method applied to the steady state problem. In particular, this article is organized as follows. Section 2 introduces the DD+L^2-jumps method and its discretization. In Section 3, we present our AMR strategy for the source problem based on a posteriori error estimators. Numerical results are shown in Section 4. Finally, Section 5 draws some concluding remarks.

2 Variational formulation and discretization

2.1 Monodomain problem

We assume that the domain $\mathcal{R}$ is a bounded, connected and open subset of $\mathbb{R}^d$ for $d = 2, 3$, having a Lipschitz boundary which is piecewise smooth. To be convenient, let us introduce the following spaces :

$$L := L^2(\mathcal{R}), \ V := H_0^1(\mathcal{R}),$$
$$\mathbf{Q} := \mathbf{H}(\text{div}, \mathcal{R}) := \{\mathbf{v} \in (L^2(\mathcal{R}))^d \mid \text{div}\, \mathbf{v} \in L^2(\mathcal{R})\}, \ \mathcal{X} := \mathbf{Q} \times L.$$

Let $(\mathcal{T}_h)_h$ be a family of meshes, made for instance of simplices, or of rectangles ($d = 2$), resp. cuboids ($d = 3$), indexed by a parameter h equal to the largest diameter of elements of a given mesh. We introduce $\mathcal{X}_h = \mathbf{Q}_h \times L_h$, the conformal discrete space associated with the Raviart-Thomas-Nédélec elements, which satisfies

$$\mathbf{Q}_h \subset \mathbf{Q}, \text{ and } L_h \subset L.$$

The discrete variational formulation associated to Problem (1) writes

$$\text{Find } (\zeta_h, k_{\text{eff}}^h) \in \mathcal{X}_h \times \mathbb{R}^+ \text{ such that for all } \xi_h \in \mathcal{X}_h, \quad c(\zeta_h, \xi_h) = \frac{1}{k_{\text{eff}}^h} f(\zeta_h, \xi_h). \quad (2)$$

where the bilinear forms are defined on $\mathcal{X} \times \mathcal{X}$ for all $\zeta := (\mathbf{p}, \phi)$ and $\xi := (\mathbf{q}, \psi) \in \mathcal{X}$,

- $c(\zeta, \xi) := -\int_{\mathcal{R}} \mathbb{D}^{-1} \mathbf{p} \cdot \mathbf{q} + \int_{\mathcal{R}} \phi \, \text{div}\, \mathbf{q} + \int_{\mathcal{R}} \text{div}\, \mathbf{p}\, \psi + \int_{\mathcal{R}} \Sigma_a \phi \psi,$

- $f(\zeta, \xi) := \int_{\mathcal{R}} \nu \Sigma_f \phi \psi.$

Traditionally, this problem is solved by the inverse power iteration algorithm, which relies on the resolution of a *discrete* source problem [9, Chapter 4]. Let $S_f \in L$ be a given source term, the source problem writes

$$\text{Find } \zeta_h \in \mathcal{X}_h \text{ such that for all } \xi_h := (\mathbf{q}_h, \psi_h) \in \mathcal{X}_h, \quad c(\zeta_h, \xi_h) = \int_{\mathcal{R}} S_f \psi_h. \quad (3)$$

2.2 Domain decomposition+L^2-jumps method

In this section, we present the domain decomposition+L^2-jumps method applied to the source problem (3). Let us define a partition $\{\overset{*}{\mathcal{R}}_i\}_{1\le i\le \overset{*}{N}}$ of $\mathcal{R}$. For $\psi \in L$, we will use the notation

$$\psi_i := \psi|_{\overset{*}{\mathcal{R}}_i},$$

for $1 \le i \le \overset{*}{N}$. We will note Ξ_{ij} the interface between two subdomains of $\overset{*}{\mathcal{R}}_i$ and $\overset{*}{\mathcal{R}}_j$. We define the interface Ξ by

$$\Xi := \cup_{i=1}^{\overset{*}{N}} \cup_{j=i+1}^{\overset{*}{N}} \overline{\Xi_{ij}}.$$

We now define the following spaces

$$P\overset{*}{H}{}_0^1 (\mathcal{R}) := \{\psi \in L^2(\mathcal{R}) | \psi_i \in H^1(\overset{*}{\mathcal{R}}_i), \psi|_{\partial \overset{*}{\mathcal{R}}_i \setminus \Xi} = 0, 1 \le i \le \overset{*}{N}\},$$

$$\mathbf{P}\overset{*}{\mathbf{H}}(\mathrm{div}\,, \mathcal{R}) := \{\mathbf{q} \in L^2(\mathcal{R}) \mid \mathbf{q}_i \in \mathbf{H}(\mathrm{div}\,, \overset{*}{\mathcal{R}}_i), 1 \le i \le \overset{*}{N}\},$$

$$M := \{\psi_I \in \prod_{i<j} L^2(\Xi_{ij})\},$$

$$\overset{*}{\mathbf{Q}} := \{\mathbf{q} \in \mathbf{P}\overset{*}{\mathbf{H}}(\mathrm{div}\,, \mathcal{R}) \mid [\mathbf{q} \cdot \mathbf{n}] \in M\},$$

$$\mathbb{W} := \overset{*}{\mathbf{Q}} \times L^2(\mathcal{R}) \times M,$$

where $[\mathbf{q} \cdot \mathbf{n}]$ is called the global jump of the normal component and is defined by

$$[\mathbf{q} \cdot \mathbf{n}]|_{\Xi_{ij}} := \mathbf{q}_i \cdot \mathbf{n}_i + \mathbf{q}_j \cdot \mathbf{n}_j, \text{ for } 1 \le i < j \le \overset{*}{N}.$$

The DD+L^2-jumps method writes,

$$\begin{cases}
\text{Find } (\mathbf{p}, \phi, \phi_I) \in \overset{*}{\mathbf{Q}} \times P\overset{*}{H}{}_0^1 (\mathcal{R}) \times M \text{ such that} \\
-\mathbb{D}_i^{-1}\mathbf{p}_i - \mathbf{grad}\phi_i = 0 \quad \text{in } \overset{*}{\mathcal{R}}_i, \quad \text{for } 1 \le i \le \overset{*}{N}, \\
\mathrm{div}\,\mathbf{p}_i + \Sigma_{a,i}\phi_i = S_{f,i} \quad \text{in } \overset{*}{\mathcal{R}}_i, \quad \text{for } 1 \le i \le \overset{*}{N}, \\
\phi_i = \phi_I \quad \text{on } \partial \overset{*}{\mathcal{R}}_i \cap \Xi, \quad \text{for } 1 \le i \le \overset{*}{N}, \\
[\mathbf{p} \cdot \mathbf{n}]_{\Xi_{ij}} = 0 \quad \text{for } 1 \le i < j \le \overset{*}{N}.
\end{cases}$$

The flux and the current respectively belong to the broken spaces $P\overset{*}{H}{}_0^1 (\mathcal{R})$ and $\mathbf{P}\overset{*}{\mathbf{H}}(\mathrm{div}\,, \mathcal{R})$. The space M is the space of the Lagrange multipliers, defined on the interface. They are introduced in order to impose the constraint on the jump of the normal component of the current. The key observation in the domain decomposition+L^2-jumps method is that the jump of the normal component of the current belongs to M. The associated variational formulation writes

$$\text{Find } \mathbf{u} = (\mathbf{p}, \phi, \phi_I) \in \mathbb{W} \text{ such that } \forall\, \mathbf{w} = (\mathbf{q}, \psi, \psi_I) \in \mathbb{W}, \quad c_S(\mathbf{u}, \mathbf{w}) = \int_{\mathcal{R}} S_f \psi, \tag{4}$$

where

$$c_S(\mathbf{u}, \mathbf{w}) := c((\mathbf{p}, \phi), (\mathbf{q}, \psi)) + \int_{\Xi} [\mathbf{p} \cdot \mathbf{n}]\psi_I - \int_{\Xi} [\mathbf{q} \cdot \mathbf{n}]\phi_I.$$

The discrete variational formulation writes,

$$\text{Find } \mathbf{u}_h = (\mathbf{p}_h, \phi_h, \phi_{I,h}) \in \mathbb{W}_h \text{ such that } \forall \mathbf{w}_h = (\mathbf{q}_h, \psi_h, \psi_{I,h}) \in \mathbb{W}_h, \; c_S(\mathbf{u}_h, \mathbf{w}_h) = \int_{\mathcal{R}} S_f \psi_h, \tag{5}$$

where $\mathbb{W}_h$ is a conformal discrete approximation space of $\mathbb{W}$. A priori estimates have been derived for low regularity solutions in [4].

3 Adaptive mesh refinement strategies

3.1 Generalities

In this paper, we aim to illustrate an AMR strategy for the problems (3) and (5). The general method generates a sequence $\mathcal{T}_{h_k}$ from the initial mesh $\mathcal{T}_{h_0}$ by using the following iterative loop, which is divided into four modules as presented in Figure 1.
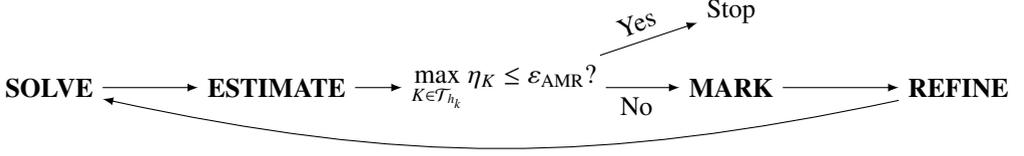
Figure 1: Description of the AMR process.

The module **SOLVE** amounts to solving the source problem (3) (respectively Problem (5)). In module **ESTIMATE**, the η_K local error indicator on each element is computed from a posteriori error estimate for the discrete solution. The stopping criterion is defined as $\max_{K \in \mathcal{T}_{h_k}} \eta_K \leq \varepsilon_{\mathrm{AMR}}$ for a user-defined $\varepsilon_{\mathrm{AMR}} > 0$. In Section 4, we use a relative stopping criterion $\varepsilon_{\mathrm{AMR}} = \varepsilon_{\mathrm{AMR,\,rel}} \|\phi_h\|_{L^2(\mathcal{R})}$, where $\varepsilon_{\mathrm{AMR,\,rel}} > 0$. The purpose of the module **MARK** is to select a set of elements with large error to be refined. For a user-defined parameter θ, the marking strategy consists in finding an optimal set of cells S such that one has

$$\eta(S) \leq \theta\,\eta(\mathcal{T}_{h_k}), \quad \text{where} \quad \eta(S) := \left(\sum_{K \in S} \eta_K^2 \right)^{1/2}.$$

According to [10, Section 6], an efficient strategy which preserves the Cartesian structure of the mesh is the *direction* marker strategy. One selects for each direction $\mathbf{e}_x$, $x = 1, \ldots, d$, the smallest set of lines $L_x \subset \mathcal{T}_{h_k}$ such that $\eta(L_x) \geq \theta\eta(\mathcal{T}_{h_k})$. The resulting selected set is $S = \cup_{x=1,\ldots,d} L_x$. We will use this marking strategy through the rest of this paper.

Finally, the module **REFINE** refines the mesh. In the case of monodomain approach, the marker strategy is designed such that it preserves the Cartesian structure of the mesh. This constraint is somehow relaxed in the case of the DD+L^2-jumps method. Since the DD+L^2-jumps method allows non conformity at the interface between two subdomains, the marker strategy can be applied *independently* on each subdomain.

In the case of generalized eigenvalue problem, we simply replace the definition of the **SOLVE** module in Figure 1 by the resolution of Problem (2). We refer to [1, Section 3] for a generalized algorithm.

3.2 A posteriori error estimators for the source problem

3.2.1 Reconstruction

Since the discrete flux is computed with Raviart-Thomas-Nédélec elements, it is possible that $\phi_h \notin V$. We introduce a *reconstruction* $\tilde{\zeta}_h = (\tilde{\mathbf{p}}_h, \tilde{\phi}_h)$ of $\zeta_h = (\mathbf{p}_h, \phi_h)$ defined as

$$\tilde{\mathbf{p}}_h = \mathbf{p}_h \in \mathbf{Q}_h \subset \mathbf{H}(\mathrm{div}\,, \mathcal{R}), \quad \tilde{\phi}_h \in V.$$

Note that the current is not reconstructed. For different reconstruction methods in the monodomain case, we refer to [1, Section 5]. For the DD+L^2-jumps method, a *reconstruction* $\tilde{\mathfrak{u}}_h = (\tilde{\mathbf{p}}_h, \tilde{\phi}_h, \tilde{\phi}_{I,h})$ of $\mathfrak{u}_h = (\mathbf{p}_h, \phi_h, \phi_{I,h})$ is defined as

$$\tilde{\mathbf{p}}_h = \mathbf{p}_h \in \mathbf{Q}_h \subset \mathbf{H}(\mathrm{div}\,, \mathcal{R}), \quad \tilde{\phi}_h \in V, \quad \tilde{\phi}_{I,h} = \tilde{\phi}_h \text{ on } \Xi.$$

3.2.2 Monodomain problem

In [10], the authors show the existence of a posteriori error estimates for Problem (3). For all $\zeta = (\mathbf{p}, \phi), \xi = (\mathbf{q}, \psi) \in \mathcal{X}$, we define

- $d(\zeta, \xi) := \int_{\mathcal{R}} \mathbb{D}^{-1} \mathbf{p} \cdot \mathbf{q} + \int_{\mathcal{R}} \Sigma_a \phi \psi + \int_{\mathcal{R}} \psi \, \mathrm{div}\, \mathbf{p} - \int_{\mathcal{R}} \phi \, \mathrm{div}\, \mathbf{q} = c(\zeta, (-\mathbf{q}, \psi)),$

- $\|\zeta\|_+ := \int_{\mathcal{R}} \mathbb{D}^{-1} \mathbf{p} \cdot \mathbf{p} + \int_{\mathcal{R}} \Sigma_a \phi \phi + \sum_{K \in \mathcal{T}_{h_k}} h_K^2 (\mathbb{D}_K^{\min})^{-1} \|\mathrm{div}\, \mathbf{p}\|_{L^2(K)}^2.$

For $K \in \mathcal{T}_h$, we define $N(K) := \{K' \in \mathcal{T}_h \mid \dim_H(\partial K' \cap \partial K) = d - 1)\}$, where $\dim_H$ is the Hausdorff's dimension. $N(K)$ corresponds to the set of the neghbours of K, including K. We let

- $\mathcal{X}_K := \{\zeta = (\mathbf{p}, \phi) \in \mathcal{X} \mid \mathrm{Supp}(\phi) \subset K, \mathrm{Supp}(\mathbf{p}) \subset N(K)\}.$

- For $\zeta \in \mathcal{X}$, the $\mathcal{X}_K$-local norm is then defined by

$$|\zeta|_{+,K} := \sup_{\xi \in \mathcal{X}_K, \|\xi\|_+ \leq 1} d(\zeta, \xi).$$

The residual and flux estimators write

$$\eta_{r,K} := \|\Sigma_a^{-1/2}(S_f - \mathrm{div}\, \mathbf{p}_h - \Sigma_a \tilde{\phi}_h)\|_{L^2(K)} \quad \text{and} \quad \eta_{f,K} := \|\mathbb{D}^{1/2}(\mathbb{D}^{-1}\mathbf{p}_h + \mathbf{grad}\, \tilde{\phi}_h)\|_{L^2(K)}. \quad (6)$$

According to [10, Theorem 5.6], we have the reliability of the previous estimators

$$|\zeta - \tilde{\zeta}_h|_{+,K} \leq \left(\eta_{r,K}^2 + \sum_{K' \in N(K)} \eta_{f,K}^2 \right)^{1/2}.$$

Under some hypothesis on the data, we also have the efficiency of these estimators [10, Theorem 5.7].

3.2.3 DD+L^2-jumps method

For the DD+L^2-jumps method, we can follow the steps for the monodomain case. Similarly as above, we let $\mathfrak{u} = (\mathbf{p}, \phi, \phi_I) \in \mathbb{W}$, and

- $\|\mathfrak{u}\|_{+,*} := \int_{\mathcal{R}} \mathbb{D}^{-1} \mathbf{p} \cdot \mathbf{p} + \int_{\mathcal{R}} \Sigma_a \phi \phi + \sum_{K \in \mathcal{T}_h} h_K^2 (\mathbb{D}_K^{\min})^{-1} \|\mathrm{div}\, \mathbf{p}\|_{L^2(K)}^2,$

- $\mathbb{W}_K := \{\zeta = (\mathbf{p}, \phi) \in \mathcal{X} \mid \mathrm{Supp}(\phi) \subset K, \mathrm{Supp}(\mathbf{p}) \subset \overset{*}{N}(K)\}$, where $\overset{*}{N}(K) := N(K) \cap \overset{*}{\mathcal{R}}_K$ with $\overset{*}{\mathcal{R}}_K$, the subdomain which contains K,

- For $\zeta \in \mathcal{X}$,
$$|\zeta|_{+,K,*} := \sup_{\mathfrak{w} = (\xi, \psi_I) \in \mathbb{W}_K, \|\mathfrak{w}\|_{+,*} \leq 1} d(\zeta, \xi).$$
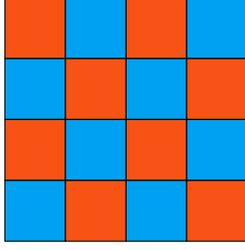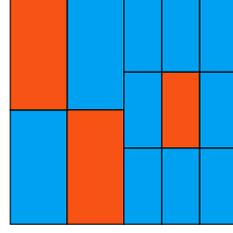
(a) Checkerboard test case.   (b) Dauge-Center test case.

Figure 2: The geometry for the test cases.

One can prove that if $\tilde{\phi}_{I,h} = \tilde{\phi}_h$ on Ξ, then

$$|\zeta - \tilde{\zeta}_h|_{+,K,*} \leq \left(\eta_{r,K}^2 + \sum_{K' \in \overset{*}{N}(K)} \eta_{f,K'}^2 \right)^{1/2}.$$

Under the same hypothesis as above on the data, one can also prove the efficiency of the residual and flux estimators. The proof is left to a publication in preparation.

4 Numerical results

4.1 Comparison of estimators

In this subsection, we would like to show numerically the improvements of the estimators proposed in [10] compared to those presented in [1]. The AMR generates a mesh which contains eventually less elements compared to the results presented in [1].
The estimator defined in [1] is denoted

$$\eta_{K,0} = \sqrt{\hat{\eta}_{r,K}^2 + \eta_{f,K}^2 + 5\eta_{nc,K}^2},$$

where

$$\hat{\eta}_{r,K} = \|S_f - \operatorname{div} \mathbf{p}_h - \Sigma_a \phi_h\|_{L^2(K)}, \quad \eta_{nc,K} = \|\Sigma_a^{1/2}(\tilde{\phi}_h - \phi_h)\|_{L^2(K)}.$$

In this paper, we use the estimator presented in [10]. The estimator is defined by

$$\eta_{K,1} = \sqrt{\eta_{r,K}^2 + \sum_{K \in N(K)} \eta_{f,K}^2}.$$

We perform the checkerboard test case [1, Section 4.1] for the one-group diffusion equation on the domain of computation $\mathcal{R} = (0, 100)^2$. The aim is to compare the AMR process using the a posteriori error estimates presented in [1] with the AMR process using the ones introduced in [10].
The diffusion coefficient D is given as in Figure 2(a) with $D = 5$ in the red region and $D = 1$ in the blue region. We set $\nu\Sigma_f = 1$ and the absorption cross section $\Sigma_a = \Sigma_t - \Sigma_s = 1$. We impose Dirichlet boundary condition on the flux. The initial mesh is a 12×12 uniform mesh. The **SOLVE** module is performed by a RTN$_0$ finite element method. We apply the averaging method [1, Section 5.1.1] for the reconstruction. The ESTIMATE module differs on

the choice of the a posteriori error estimator, $\eta_{K,0}$ or $\eta_{K,1}$. The **MARK** module applies the direction marker strategy and we fix $\theta = 0.5$.

In order to compare the AMR processes using either $\eta_{K,0}$ or $\eta_{K,1}$, we compare the number of elements needed to reach a given precision, relatively to a reference $k_{\text{eff,ref}}$. The latter is obtained from a RTN$_0$ computation on a uniform 1000 x 1000 mesh, which gives $k_{\text{eff,ref}}$ = 0, 995194.

It is observed in Figure 3 that in order to reach a 1 pcm precision relative to $k_{\text{eff,ref}}$, the $\eta_{K,0}$-AMR needs 6 refinements, ending up with a mesh containing 12100 elements. On the other hand, the $\eta_{K,1}$-AMR generates 6084 elements in 5 refinements. Finally, in order to reach the same precision with a uniform mesh, one needs more than 30 000 elements. The total AMR process is presented in Table 1.
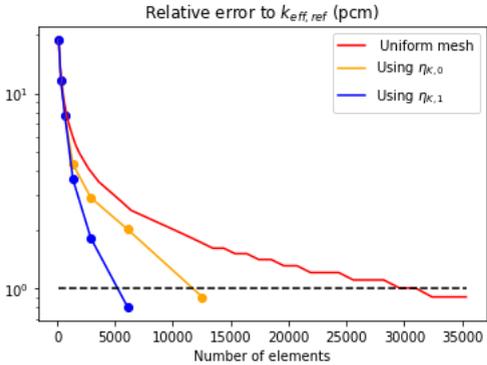
Iter	Number of elements		$\lvert\mathcal{R}\rvert / \min\limits_{K \in \mathcal{T}_{h_k}} \lvert K \rvert$	
	With $\eta_{K,0}$	With $\eta_{K,1}$	With $\eta_{K,0}$	With $\eta_{K,1}$
0	144	144	144	144
1	324	324	829	829
2	676	676	3317	3317
3	1444	1369	3317	13271
4	2916	2916	13271	53084
5	6084	6084	13271	212336
6	12100	-	53084	-

Figure 3: Comparison of the estimators $\eta_{K,0}$ and $\eta_{K,1}$ and a uniform refinement. The dashed line represents the stopping criterion for the splitting algorithm.

Table 1: Adaptive mesh refinement for the checkerboard test case.

4.2 DD+L^2-jumps method for the source and eigenvalue problem

In this subsection, we would like to assess our AMR strategy on the DD+L^2-jumps method applied to a specific one-group diffusion equation. We study the resolution of the source problem and the eigenvalue problem.

The test cases are performed on the square $\mathcal{R} = (0, 100)^2$ where the diffusion coefficient $D = 10$ in the red region and $D = 1$ in the blue region of the geometry as illustrated in Figure 2(b). The absorption cross section $\Sigma_a = 1$. The domain decomposition is fixed by dividing the domain at $x = 50$, resulting in a configuration containing two subdomains : a two by two checkerboard for the first subdomain, and a second subdomain where the material corresponding to the red region is located at the center of the subdomain.

For the reconstruction of the discrete solution of the DD+L^2-jumps method, it is necessary to tackle non conformities at the interface. The method we use here is an extension of the averaging method. Let us describe it in the case of a RTN$_0$ discretization on a two-dimensional mesh.

First, we create a mesh by projecting the nodes located at the non-conformities of the interface onto the mesh of the corresponding subdomain. Theses new nodes are represented by yellow rectangles in Figure 4(b). This defines a new mesh called $\overset{*}{\mathcal{T}}_h$.
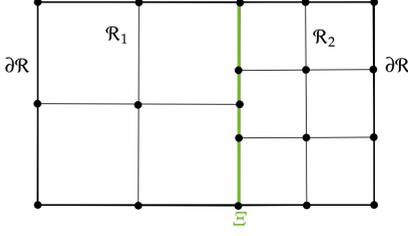
(a) Computational mesh.
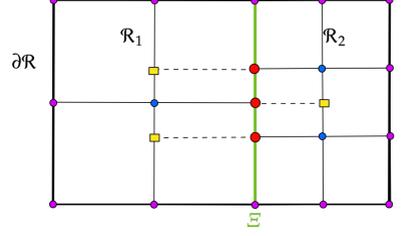
(b) Reconstruction at interface for the DD+L^2-jumps case.

Figure 4: Computational mesh and reconstruction. The interface is noted Ξ, the two subdomains are $\mathcal{R}_1$ and $\mathcal{R}_2$, and the border is $\partial\mathcal{R}$.

It is sufficient to define the reconstruction on the nodes of $\overset{*}{\mathcal{T}}_h$. We use the boundary condition for the nodes located at the border (the purple circles in Figure 4(b)). We now consider the inner nodes.

For a given node v of $\overset{*}{\mathcal{T}}_h$, we call $d(v) \in \mathbb{N}$ the minimal number of edges between the node and the interface.

For the nodes v such that $d(v) = 0$ (the red circles in Figure 4(b)), we take the averaged value of the Lagrange multipliers corresponding to the edges which contain v. On the nodes v such that $d(v) \geq 2$, the reconstruction is computed in the same way as the monodomain case. For the Lagrange nodes $v \in \overset{*}{\mathcal{T}}_h$ such that $d(v) = 1$, two cases occur :

- the case where the node v is in $\mathcal{T}_h$ (the blue circles in Figure 4(b)). In this case, the reconstruction is computed with the averaging method . Let us suppose that all these nodes have been computed before the second kind of nodes.

- the case where the node v is in $\overset{*}{\mathcal{T}}_h$, but not in $\mathcal{T}_h$ (the yellow rectangles in Figure 4(b)). In this case, the reconstruction is computed by interpolation. Since all circle nodes have been computed, we use the value of the reconstruction at theses nodes to compute its value at the retangular nodes.

In general, AMR is applied to increase the accuracy of solution. The AMR stopping criterion is set at $\varepsilon_{\text{AMR,rel}} = 4.3 \times 10^{-3}$ for both test cases.

4.2.1 Source problem

For the source problem, we consider the simple constant source term $S_f = 1$. In this test, we would like to emphasize that the DD+L^2-jumps method enhances the efficiency of the AMR when applied to structured meshes. It is also interesting to point out that the non-conforming grid at the subdomain interface allows local independent refinement on each subdomain, clearly demonstrated in Figure 5. Hence, the AMR strategy performs much better with DD+L^2-jumps than in the monodomain case. More importantly, one can choose independently one value for the threshold θ on each subdomain to reduce the total number of elements on each subdomain as illustrated in Table 2.

(a) Monodomain, $\theta = 0.5$. (b) DD+L^2-jumps, $\theta_1 = \theta_2 = 0.5$. (c) DD+$L^2$-jumps, $\theta_1 = 0.35, \theta_2 = 0.5$.
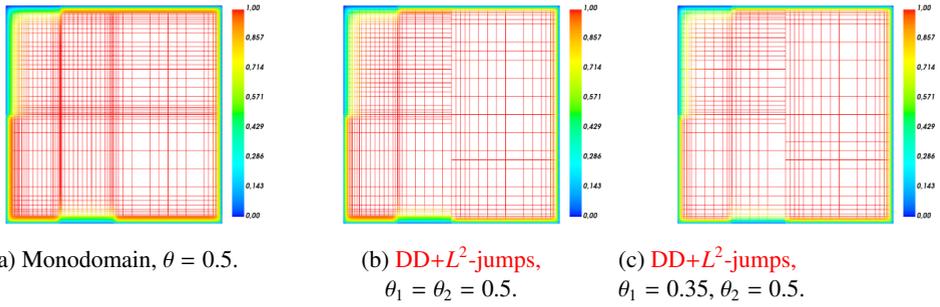
Figure 5: The discrete solution of the AMR on the source problem.

Iter	Monodomain, $\theta = 0.5$		DD+L^2-jumps, $\theta_1 = \theta_2 = 0.5$				DD+L^2-jumps, $\theta_1 = 0.35, \theta_2 = 0.5$			
	N	**max η_K**	N_1	**max η_K^1**	N_2	**max η_K^2**	N_1	**max η_K^1**	N_2	**max η_K^2**
0	144	6.250	72	6.246	72	6.250	72	6.246	72	6.250
1	225	3.523	128	3.251	128	3.519	98	3.523	120	3.519
2	361	2.252	200	2.252	190	2.280	153	2.252	190	2.28
3	576	1.267	338	0.179	299	1.180	231	1.128	231	1.180
4	1024	0.585	630	0.543	493	0.543	364	0.888	493	0.543
5	1936	0.429	1274	0.502	874	0.225	594	0.712	874	0.225
6	3844	0.521	2484	0.225	-	-	1032	0.476	-	-
7	7395	0.408	-	-	-	-	1705	0.408	-	-

Table 2: AMR with the DD+L^2-jumps method for the source problem.

4.2.2 Eigenvalue problem

For the eigenvalue problem, we use the same domain, with the same cross section values and the geometry as in Figure 2(b). We set $\nu\Sigma_f = 1$. Figure 6 and Table 3 show the efficiency of AMR with DD+L^2-jumps method for the eigenvalue problem. As can be seen, the conclusions are similar to those of the source problem.

(a) Monodomain, $\theta = 0.5$. (b) DD+L^2-jumps, $\theta_1 = \theta_2 = 0.5$. (c) DD+$L^2$-jumps, $\theta_1 = 0.35, \theta_2 = 0.5$.
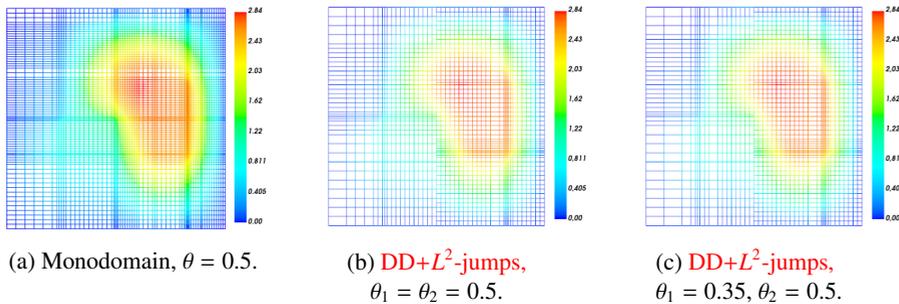
Figure 6: The discrete solution of the AMR on the eigenvalue problem.

Iter	Monodomain, $\theta = 0.5$		DD+L^2-jumps, $\theta_1 = \theta_2 = 0.5$				DD+L^2-jumps, $\theta_1 = 0.35, \theta_2 = 0.5$			
	N	$\max \eta_K$	N_1	$\max \eta_K^1$	N_2	$\max \eta_K^2$	N_1	$\max \eta_K^1$	N_2	$\max \eta_K^2$
0	144	3.335	72	1.929	72	3.335	72	1.929	72	3.335
1	306	2.340	153	1.451	162	2.426	120	1.447	162	2.421
2	648	1.869	325	0.608	338	1.080	200	0.724	338	1.079
3	1326	1.253	684	0.449	722	0.603	351	0.622	722	0.603
4	2688	0.989	-	-	1458	0.360	595	0.531	1458	0.360
5	5293	0.637	-	-	-	-	-	-	-	-
6	10509	0.431	-	-	-	-	-	-	-	-

Table 3: AMR with the DD+L^2-jumps method for the eigenvalue problem.

5 Conclusion

In this work, we propose an adaptive mesh refinement strategy for the Domain Decomposition+L^2 jumps method for the neutron diffusion equation, which relies on a posteriori error error estimators for the source problem and a splitting approach to handle the generalized eigenvalue problem. We have shown numerically that this approach is more effective than in the monodomain case. Future work will be dedicated to the extension of this method to the SP$_N$ model.

References

[1] M.H. Do, P. Ciarlet, F. Madiot, *Adaptive solution of the neutron diffusion equation with heterogeneous coefficients using the mixed finite element method on structured meshes*, in *EPJ Web of Conferences* (EDP Sciences, 2021), Vol. 247, p. 02002

[2] P. Mosca, L. Bourhrara, A. Calloo, A. Gammicchia, F. Goubioud, L. Lei-Mao, F. Madiot, F. Malouch, E. Masiello, F. Moreau et al., *APOLLO3®: Overview of the new code capabilities for reactor physics analysis*, in *Proceeding of International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering, M&C 2023* (2023)

[3] A.M. Baudron, J.J. Lautard, Nuclear Science and Engineering **155**, 250 (2007)

[4] P. Ciarlet, E. Jamelot, F. Kpadonou, Computers & Mathematics with Applications **74**, 2369 (2017)

[5] Y. Wang, W. Bangerth, J. Ragusa, Progress in Nuclear Energy **51**, 543 (2009)

[6] Y. Wang, J. Ragusa, Nuclear Science and Engineering **161**, 22 (2009)

[7] M. Vohralík, SIAM Journal on Numerical Analysis **45**, 1570 (2007)

[8] L. Giret, P. Jr. Ciarlet, E. Jamelot, *Criticality Computation with Finite Element Method on Non-Conforming Meshes*, in *Proceeding of International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering, M&C 2017* (2017)

[9] J. Planchard, *Méthodes mathématiques en neutronique* (Eyrolles, EDF, 1995)

[10] P. Ciarlet, M.H. Do, F. Madiot, ESAIM: M2AN **57**, 1 (2023)