

1 Présentation de l'IPP

2 B-splines et espaces de Schoenberg

Lors des cours sur les éléments finis que nous avons suivis en MACS2, nous avons uniquement abordé les espaces P1 et P2 pour résoudre les formulations variationnelles. Dans cette section, nous allons introduire les notions de B-splines et d'espaces de Schoenberg que j'ai utilisées lors de ce stage en tant qu'espaces d'éléments finis.

Intuitivement, les espaces de Schoenberg sont des espaces de polymômes par morceaux, de degrés inférieurs ou égaux à un entier p fixé. Ces polynômes sont définis sur des intervalles $[t_i, t_{i+1}]$, et doivent vérifier des conditions de régularité aux points t_i . Mais avant parler plus en détails des espaces de Schoenberg, nous introduisons les B-splines qui permettront par la suite de construire ces espaces.

2.1 Bsplines

Définition 1 (Formule de Cox-DeBoor). *Soit m et p deux entiers naturels, et $T = \{t_i\}_{0 \leq i \leq m}$ une suite de réels croissante.*

La j -ième B-spline de degré p est définie par la relation de récurrence :

$$N_j^p(t) = \frac{t - t_j}{t_{j+p} - t_j} N_j^{p-1}(t) + \frac{t_{j+p+1} - t}{t_{j+p+1} - t_{j+1}} N_{j+1}^{p-1}(t) \quad (1)$$

pour $0 \leq j \leq m - p - 1$ et

$$N_j^0 = \mathbf{1}_{[t_j, t_{j+1}]}$$

On appelle T le vecteur de nœuds.

Remarque 1. *Il est possible que lorsqu'on utilise la relation de récurrence précédente, on rencontre des divisions par 0. En revanche, il est assez facile de montrer que si $t_{j+p} - t_j$ est nul N_j^{p-1} l'est également. Dans ce genre de situation, on utilise donc la convention $\frac{0}{0} = 0$.*

Remarque 2. *A p et T fixés, on note souvent $m = n + p + 1$. De cette manière j varie entre 0 et n , et on obtient $n + 1$ B-splines.*

Nous allons à présent voir quelques exemples afin d'observer comment le choix du vecteur de nœuds et du degré affecte les propriétés des B-splines.

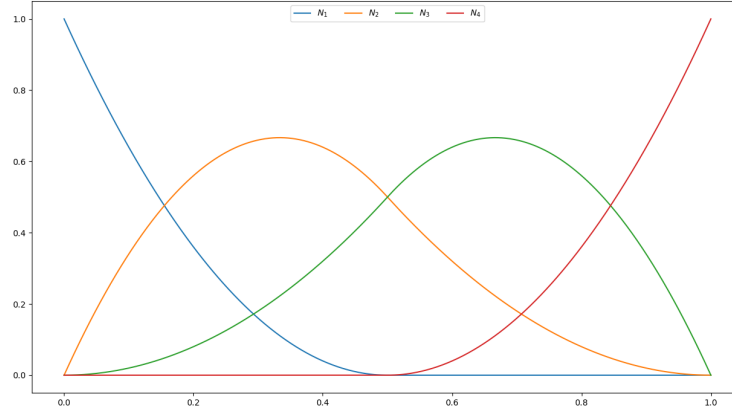


Figure 1: B-splines de degrés 2 obtenues avec $T = \{0, 0, 0, \frac{1}{2}, 1, 1, 1\}$

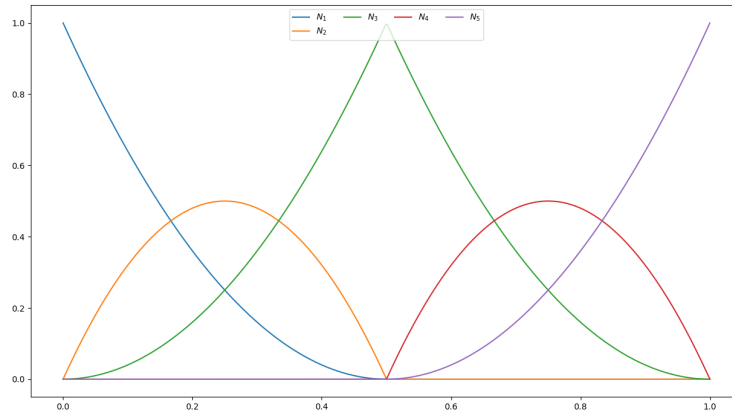


Figure 2: B-splines de degrés 2 obtenues avec $T = \{0, 0, 0, \frac{1}{2}, \frac{1}{2}, 1, 1, 1\}$

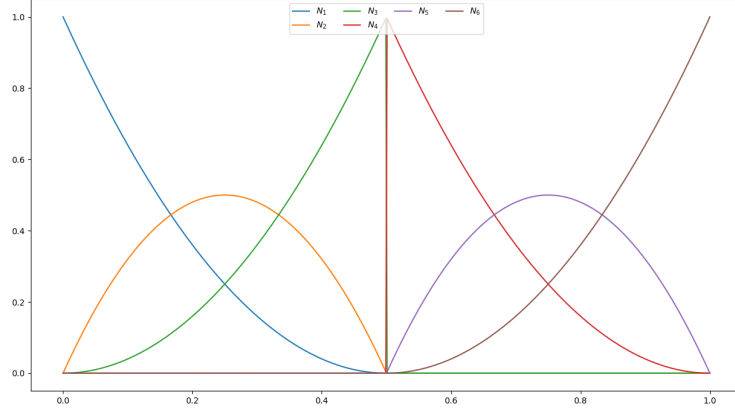


Figure 3: B-splines de degrés 2 obtenues avec $T = \{0, 0, 0, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 1, 1, 1\}$
 Les 3 figures permettent de mettre en évidence l'importance de la multiplicité des éléments d'un vecteur de nœuds. On dit qu'un élément t du vecteur de nœuds est de multiplicité m si cet élément apparait exactement m fois dans le vecteur de nœuds.

On remarque en particulier qu'augmenter la multiplicité d'un élément diminue la régularité des B-splines à cet élément. En effet les B-splines des figures 1 et 2 sont respectivement de régularité \mathcal{C}^1 et \mathcal{C}^0 , alors que les B-splines de la figure 3 ne sont pas continues.

Pour la plupart des codes implémentés lors de ce stage, j'ai donc utilisé des vecteurs de nœuds du type.

$$T = \{x_0, \dots, x_0, x_1, \dots, x_{n-1}, x_n, \dots, x_n\}$$

où les points x_0 et x_n sont répétés $p + 1$ fois, et les points $x_1 \dots x_{n-1}$, dits intérieurs, sont de multiplicité 1. Ce type de vecteurs de nœuds permet en particulier de retrouver les fonctions de base des éléments finis P1 en 1D.

Nous allons à présent introduire quelques propriétés utiles des B-splines.

Proposition 1. *Les B-splines de degré p sont des polynômes par morceaux de degré p .*

Proposition 2. *Les B-splines sont des fonctions positives sur $[t_0, t_m]$*

Proposition 3. $N_j^p(t) = 0 \ \forall t \notin [t_j, t_{j+p+1}[$

Ces propositions se montrent facilement par récurrence sur les degrés. La proposition suivante se déduit de la proposition 4.

Proposition 4. $t \in [t_i, t_{i+1}[$ et $N_j^p(t) \neq 0 \Rightarrow j \in \{i - p, \dots, i\}$

Cette proposition sur le nombre de B-splines non nulles sur un certain intervalle est très utile lors de l'implémentation. En effet, au lieu d'évaluer toutes les B-splines à un certain point t , on peut se contenter d'en évaluer $p + 1$.

Proposition 5 (Partition de l'unité). *Soit $T = \{t_i\}_{0 \leq i \leq n+p+1}$ un vecteur de nœuds et p un entier naturel. On a alors l'égalité :*

$$\sum_{j=0}^n N_j^p(t) = 1$$

$$\forall t \in [t_0, t_{n+p+1}]$$

La démonstration de cette proposition est assez représentative des méthodes généralement utilisées, sans être trop compliquée. Je vais donc la détailler.

Proof. On procède par récurrence. Le cas $p = 0$ est trivial. Soit p tel que :

$$\sum_{j=0}^{m-p-1} N_j^p(t) = 1$$

où $m = n + p + 1$ soit $t \in [t_i, t_{i+1}[$. D'après la proposition 4, on a :

$$\sum_{j=0}^{m-p-1} N_j^p(t) = \sum_{j=i-p}^i N_j^p(t)$$

(On suppose ici que si $j > m - p - 1$ ou $j < 0$ alors $N_j^p(t) = 0$).

On cherche maintenant à calculer $\sum_{j=0}^{m-p-2} N_j^{p+1}(t)$

$$\begin{aligned} \sum_{j=0}^{m-p-2} N_j^{p+1}(t) &= \sum_{j=i-p-1}^i N_j^{p+1}(t) \\ &= \sum_{j=i-p-1}^i \frac{t - t_j}{t_{j+p+1} - t_j} N_j^p(t) + \frac{t_{j+p+2} - t}{t_{j+p+2} - t_{j+1}} N_{j+1}^p(t) \\ &= \sum_{j=i-p-1}^i \frac{t - t_j}{t_{j+p+1} - t_j} N_j^p(t) + \sum_{j=i-p}^{i+1} \frac{t_{j+p+1} - t}{t_{j+p+1} - t_j} N_j^p(t) \\ &= \sum_{j=i-p}^i N_j^p(t) + N_{i-p-1}^p(t) + N_{i+1}^p(t) \\ &= 1 \end{aligned}$$

□

Proposition 6 (Dérivées de B-splines). *La dérivée de la j -ième B-spline de degré p est donnée par la formule :*

$$N_j^{p'} = \frac{p}{t_{j+p} - t_j} N_j^{p-1} - \frac{p}{t_{j+p+1} - t_{j+1}} N_{j+1}^{p-1} \quad (2)$$

La proposition suivante formalise les propriétés obserées précédement pour les figures 1 à 3.

Proposition 7 (Régularité des B-splines). *Soit T un vecteur de nœuds et p un entier naturel. On note $T^* = \{t_i^*\}$ l'ensemble des éléments de T . On note m_i la multiplicité de t_i^* . Alors, $N_j^{p(r)}$ est continue en t_i^* pour tout $0 \leq r \leq p - m_i$ et ce pour tout j . On note $r_i = p - m_i$ la régularité associée à t_i^* .*

2.2 Espaces de Schoenberg

Maintenant que nous avons définis les B-splines et donné quelques unes de leurs propriétés, nous introduisons les espaces de Schoenberg.

Théorème 1. *Sur chaque intervalle $[t_i, t_{i+1}[$, les B-splines non nulles sont linéairement indépendantes.*

Théorème 2. *Sur chaque intervalle $[t_i, t_{i+1}[$, les B-splines de degrés p sont une famille génératrice des polynômes de degrés inférieurs ou égaux à p .*

Les deux théorèmes précédents permettent de montrer que, à T et p fixés, les B-splines forment une base d'un espace de fonction "plus grand" que l'espace des polynômes de degrés inférieurs ou égaux à p , ce qui permet de donner la définition suivante.

Définition 2 (Espaces de Schoenberg). *Soit T un vecteur de nœuds et p un entier naturel. On définit l'espace de Schoenberg $S^p(T)$ par :*

$$S^p(T) = \text{Vect}\{N_j^p, 0 \leq j \leq n\}$$

Remarque 3. *Une définition équivalente des espaces de Schoenberg est donnée par la proposition 7 et les théorèmes 1 et 2. En effet, on peut en déduire assez facilement que $S^p(T)$ est en fait l'ensemble des polynômes par morceaux sur les intervalles $[t_i^*, t_{i+1}^*]$ ayant une régularité supérieure ou égale à r_i en t_i^* . Bien que cette définition soit plus simple et ne fasse pas appel aux B-splines, la définition 2 va nous être très utile. En effet, les B-splines forment des bases naturelles des espaces de Schoenberg, ayant de bonnes propriétés (comme la compacité de leurs supports).*

Nous introduisons à présent une définition des espaces de Schoenberg en 2D.

Définition 3. *Soit T_x et T_y deux vecteurs de nœuds, p_x et p_y deux entiers naturels. On définit l'espace de Schoenberg 2D $S^{p_x, p_y}(T_x, T_y)$ par :*

$$S^{p_x, p_y}(T_x, T_y) = S^{p_x}(T_x) \otimes S^{p_y}(T_y)$$

Cette définition s'étend facilement en n'importe quelle dimension.

3 Projecteurs

Maintenant que nous avons introduit les B-splines et les espaces de Schoenberg, nous allons voir dans cette section plusieurs méthodes pour projeter des fonctions dans ces espaces, et comment utiliser ces projections pour obtenir des propriétés algébriques intéressantes.

3.1 Quasi interpolateurs locaux

Définition 4. Soit T un vecteur de nœuds, j un indice tel que $t_{j+1} > t_j$, μ et ν deux indices tels que $I \cap [t_j, t_{j+1}]$ n'est pas vide, avec $I = [t_\mu, t_\nu]$. On note P^I une méthode d'approximation locale sur I (par exemple Lagrange) tel que :

$$P^I f = \sum_{i=\mu-p}^{\nu-1} b_i N_i^p \quad (3)$$

Le quasi interpolateur \mathcal{Q} est alors défini par :

$$\mathcal{Q}f = \sum_{j=0}^n \lambda_j f N_j^p$$

où $\lambda_j f$ est donné par b_j dans l'équation précédente

Exemple 1 (Quasi interpolateur quadratique à 3 points). On considère $T = [t_0^*, t_0^*, t_0^*, t_1^*, \dots, t_{a-1}^*, t_a^*, t_a^*, t_a^*]$ et $p = 2$. Pour $j \leq n$ on pose $I = [t_{j+1}, t_{j+2}]$, puis :

$$\begin{cases} x_0^j = t_{j+1} \\ x_1^j = \frac{t_{j+1} + t_{j+2}}{2} \\ x_2^j = t_{j+2} \end{cases}$$

On cherche P^I tel que $P^I f$ interpole f aux points x_i^j . Une fois les calculs effectués, on trouve :

$$\lambda_j f = \begin{cases} f(t_0), & j = 0 \\ -\frac{1}{2}f(x_0^j) + 2f(x_1^j) - \frac{1}{2}f(x_2^j), & 1 \leq j \leq n \\ f(t_{n+1}), & j = n \end{cases}$$

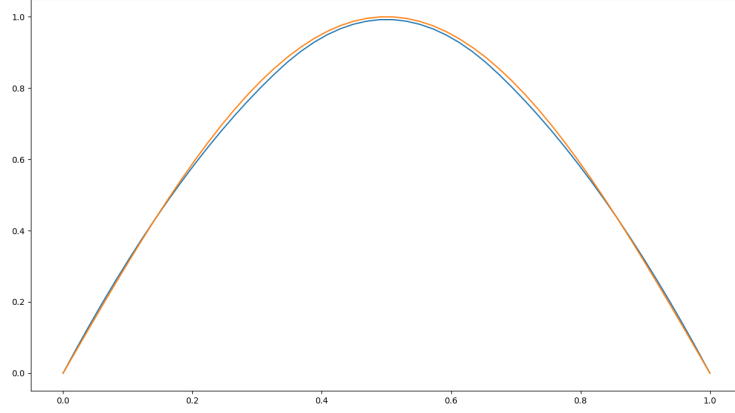


Figure 4: Quasi interpolation de $\sin(\pi x)$ entre 0 et 1 avec $n = 5$. La courbe orange est le graphe de $\sin(\pi x)$ et la courbe bleue est celui de la quasi interpolation.

3.2 Histopolations

Définition 5. Soit P une projection d'un espace de fonctions dérivables (au sens faible ou fort) E vers $\mathcal{S}^p(T)$. On dit que H_P est une histopolation associée à P si :

$$D(Pf) = H_P(Df), \quad \forall f \in E \quad (4)$$

où D est l'opérateur dérivation. On peut alors définir H_P par :

$$H_P f = D \left(P \left(x \mapsto \int_{\tau}^x f(t) dt \right) \right)$$

Si P est linéaire et exact pour les constantes, la borne inférieure d'intégration τ n'intervient pas dans la définition. On peut donc la choisir afin de simplifier les calculs.

Remarque 4. Lors de la définition du quasi interpolateur quadratique à trois points, nous avons vu qu'il suffit d'évaluer f à différents points pour obtenir les coefficients. En revanche, la définition précédente nous fait comprendre que pour l'histopolation associée, le calcul des coefficients fait intervenir des intégrales de f sur différents intervalles. D'un point de vue numérique, la relation de commutativité (4) ne sera donc pas exacte.

Proposition 8. Si P est une projection vers $\mathcal{S}^p(T)$, H_P est une projection vers $\mathcal{S}^{p-1}(T)$.

Cette proposition est une conséquence de la formule de dérivation des B-splines.

Exemple 2 (Histopolation associée au quasi interpolateur quadratique à trois points). *Pour trouver les formules d'une Histopolation, il faut se rapeller de la formule donnant la dérivée d'une B-spline. Soit ϕ une fonction de $\mathcal{S}^p(T)$:*

$$\phi = \sum_{i=0}^n \phi_i N_i^p$$

Alors ϕ' est donné par :

$$\phi' = \sum_{i=0}^{n-1} \frac{\phi_{i+1} - \phi_i}{t_{i+p+1} - t_{i+1}} N_i^{p-1}$$

Cette identité nous permet de trouver les formules pour l'histopolation. Soit $H_Q f = \sum_{i=0}^n \lambda_i f N_i^{p-1}$. On a alors :

$$\lambda_i f = \frac{1}{t_{i+p+1} - t_{i+1}} \begin{cases} 2 \int_{x_{-2}^i}^{x_0^i} f(t) dt + \frac{3}{2} \int_{x_0^i}^{x_1^i} f(t) dt - \frac{1}{2} \int_{x_1^i}^{x_2^i} f(t) dt, & i = 0 \\ -\frac{1}{2} \int_{x_{-2}^{i-1}}^{x_{-1}^{i-1}} f(t) dt + \frac{3}{2} \int_{x_{-1}^{i-1}}^{x_1^i} f(t) dt - \frac{1}{2} \int_{x_1^i}^{x_2^i} f(t) dt, & 1 \leq i \leq n-1 \\ -\frac{1}{2} \int_{x_{-2}^0}^{x_0^i} f(t) dt + \frac{3}{2} \int_{x_{-1}^i}^{x_1^i} f(t) dt, & i = n \end{cases}$$

Où x_0^i , x_{-1}^i et x_{-2}^i ont été définis dans l'exemple précédent et :

$$\begin{cases} x_{-2}^i = t_i \\ x_{-1}^i = \frac{t_{i+1} + t_i}{2} \end{cases}$$

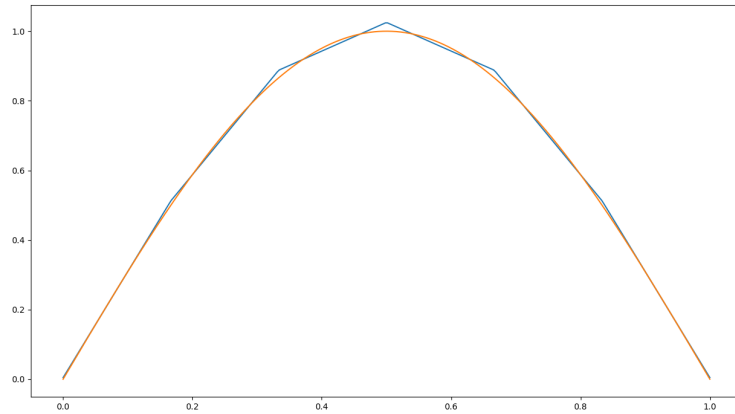


Figure 5: Histopolation associée au quasi interpolateur quadratique pour $\sin(\pi x)$ entre 0 et 1 avec $n = 7$. La courbe orange est le graphe de $\sin(\pi x)$ et la courbe bleue est celui de l'histopolation.

3.3 Séquences de DeRham et diagrammes commutatifs

Maintenant que nous avons introduit différents projecteurs pour des fonctions à 1 variable, nous allons voir comment construire des projecteurs sur des espaces de Schoenberg en 2D vérifiant des propriétés de commutation. Mais tout d'abord, nous devons introduire la notion de séquence exacte, puis les séquences de DeRham.

Définition 6. Soient E_1, E_2, \dots, E_n des espaces vectoriels et f_1, \dots, f_{n-1} des applications linéaires avec $f_i : E_i \rightarrow E_{i+1}$. On dit que la séquence

$$E_1 \xrightarrow{f_1} E_2 \xrightarrow{f_2} \dots \xrightarrow{f_{n-2}} E_{n-1} \xrightarrow{f_{n-1}} E_n \quad \text{est exacte si :}$$

$$Im(f_i) = ker(f_{i+1}), \quad \forall i$$

Proposition 9 (Séquence de DeRham en 2D). Si $\Omega \subset \mathbb{R}^2$ est homotope à un disque (sans trous) alors la séquence suivante est exacte:

$$H^1(\Omega) \xrightarrow{\nabla} \mathbf{H}(curl, \Omega) \xrightarrow{\nabla \times} L^2(\Omega)$$

Remarque 5. Ici l'opérateur $\nabla \times$ est défini par $\nabla \times \mathbf{u} = \partial_x u_y - \partial_y u_x$ et $\mathbf{H}(curl, \Omega) = \{\mathbf{u} \in (L^2(\Omega))^2, \nabla \times \mathbf{u} \in L^2(\Omega)\}$

La démonstration de ce théorème fait appel à des notions de Géométrie différentielle et d'analyse fonctionnelle que je n'ai pas abordées lors de ce stage. Il est en revanche quasiment immédiat de montrer une première inclusion. En effet les définitions des différents espaces de Sobolev et opérateurs différentiels donnent directement $Im(\nabla) \subset ker(\nabla \times)$ et $Im(\nabla \times) \subset L^2(\Omega)$. Nous allons à présent introduire la version discrete de cette séquence.

Proposition 10. Soient T_x, T_y des vecteurs de nœuds, p_x et p_y des entiers naturels. On définit les espaces de Schoenberg suivants :

$$V_h(grad, \Omega) = \mathcal{S}^{p_x}(T_x) \otimes \mathcal{S}^{p_y}(T_y)$$

$$V_h(curl, \Omega) = \begin{pmatrix} \mathcal{S}^{p_x-1}(T_x) \otimes \mathcal{S}^{p_y}(T_y) \\ \mathcal{S}^{p_x}(T_x) \otimes \mathcal{S}^{p_y-1}(T_y) \end{pmatrix}$$

$$V_h(L^2, \Omega) = \mathcal{S}^{p_x-1}(T_x) \otimes \mathcal{S}^{p_y-1}(T_y)$$

La séquence suivante est alors exacte :

$$V_h(grad, \Omega) \xrightarrow{\nabla} V_h(curl, \Omega) \xrightarrow{\nabla \times} V_h(L^2, \Omega)$$

Comme précédemment, la première inclusion se trouve directement. Pour trouver les égalités, il suffit donc de montrer $rg(\nabla) = dim(ker(\nabla \times))$ et $rg(\nabla \times) = dim(V_h(L^2, \Omega))$.

Maintenant que nous avons introduit les séquences de DeRham continues et discrètes, nous allons définir des projecteurs afin de "lier" ces deux séquences dans un diagramme commutatif. Pour se faire, nous allons tout d'abord introduire une notation qui nous sera utile.

Définition 7. Soient Π_x et Π_y deux projecteurs vers des espaces de Schoenberg en 1D. On note :

$$\Pi_x \odot \Pi_y f = \Pi_y(y \mapsto \Pi_x(x \mapsto f(x, y)))$$

Définition 8. Soient Π_x et Π_y deux projecteurs vers des espaces de Schoenberg en 1D \mathcal{S}^{p_x} et \mathcal{S}^{p_y} , et H_x, H_y leurs histopolations associées. On considère les espaces $V_h(\text{grad}, \Omega)$, $V_h(\text{curl}, \Omega)$ et $V_h(L^2, \Omega)$ définis dans la proposition 10. On introduit les projecteurs suivants :

$$\Pi^1 = \Pi_x \odot \Pi_y$$

$$\Pi^{\text{curl}} = \begin{pmatrix} H_x \odot \Pi_y \\ \Pi_x \odot H_y \end{pmatrix}$$

$$\Pi^2 = H_x \odot H_y$$

Proposition 11. Le diagramme suivant est commutatif :

$$\begin{array}{ccccc} H^1(\Omega) & \xrightarrow{\nabla} & \mathbf{H}(\text{curl}, \Omega) & \xrightarrow{\nabla \times} & L^2(\Omega) \\ \downarrow \Pi^1 & & \downarrow \Pi^{\text{curl}} & & \downarrow \Pi^2 \\ V_h(\text{grad}, \Omega) & \xrightarrow{\nabla} & V_h(\text{curl}, \Omega) & \xrightarrow{\nabla \times} & V_h(L^2, \Omega) \end{array}$$

Cette proposition se montre aisément par la définition d'une histopolation. Maintenant que nous avons défini ce diagramme commutatif, il est possible d'écrire un code implémentant les différents projecteurs afin de tester les relations de commutation.

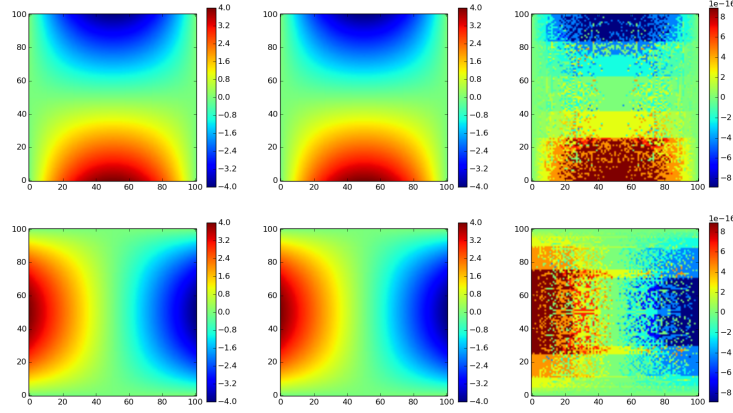


Figure 6: Test de commutativité pour la fonction $\sin(\pi x)\sin(\pi y)$ pour Π_x et Π_y quasi interpolateurs quadratiques à 3 points. La première colonne représente $\nabla(\Pi^1 f)$, la deuxième colonne $\Pi^{curl}(\nabla f)$, et la troisième colonne est la différence entre les deux premières.

La figure précédente n'utilise que quelques B-splines selon x et y . La précision machine est donc bien due à la relation de commutation, et non à l'exactitude des projections. En revanche, pour obtenir de tels résultats, il a fallu utiliser des méthodes de quadratures suffisamment précises lors de l'implémentation des histopolations. En effet, la relation de commutativité ne fonctionne pas réellement d'un point de vue numérique, car les formules d'histopolations font intervenir des intégrales de f .

4 Equations de l'acoustique

Maintenant que nous avons introduit les outils nécessaires, nous pouvons nous lancer dans l'étude de schémas pour les équations de l'acoustique :

$$\begin{cases} \partial_t p + \nabla \cdot (\alpha u) = 0, & \text{dans } \Omega \\ \partial_t u + \alpha \nabla p = 0, & \text{dans } \Omega \\ u(t=0) = u_0, & \text{dans } \Omega \\ p(t=0) = p_0, & \text{dans } \Omega \\ p = 0, & \text{sur } \partial\Omega \end{cases} \quad (5)$$

4.1 Equations homogènes

Dans un premier temps, intéressons nous au cas α constant sur Ω . Il faut tout d'abord trouver une formulation variationnelle du problème (5).

$$\begin{cases} \int_{\Omega} \partial_t p q dx + \int_{\Omega} \nabla \cdot (\alpha u) q dx = 0 \\ \int_{\Omega} \partial_t u \cdot v dx + \int_{\Omega} \alpha \nabla p \cdot v dx = 0 \end{cases}$$

$$\begin{cases} \int_{\Omega} \partial_t p q dx - \int_{\Omega} \alpha \nabla q \cdot u dx = 0 \\ \int_{\Omega} \partial_t u \cdot v dx + \int_{\Omega} \alpha \nabla p \cdot v dx = 0 \end{cases} \quad (6)$$

On remarque ici qu'en choisissant $q = p$ et $v = u$, on obtient l'égalité :

$$\partial_t \left(\frac{1}{2} \int_{\Omega} p^2 dx + \frac{1}{2} \int_{\Omega} u^2 dx \right) = 0$$

On définit donc l'énergie E (ou l'Hamiltonien) du système par :

$$E(u, p) = \frac{1}{2} \int_{\Omega} p^2 dx + \frac{1}{2} \int_{\Omega} u^2 dx$$

On peut également remarquer le fait que cette quantité est aussi préservée dans le cas α non constant. En étudiant la formulation variationnelle (6), on se rend compte que la deuxième équation, inchangée, est en fait équivalente à l'équation forte de base. Pour simplifier les notations, on note $u(t)$ la fonction u définie dans l'espace au temps t , et $p(t)$ de la même manière. On a donc la formulation variationnelle suivante :

En forme faible ; pour tout t , trouver $u(t)$ dans $\mathbf{H}(\text{curl}, \Omega)$ et $p(t)$ dans $H_0^1(\Omega)$ tels que, pour tout $v(t)$ dans $\mathbf{H}(\text{curl}, \Omega)$ et $q(t)$ dans $H_0^1(\Omega)$, on a :

$$\begin{cases} \int_{\Omega} \partial_t p q dx - \int_{\Omega} \alpha \nabla q \cdot u dx = 0 \\ \int_{\Omega} \partial_t u \cdot v dx + \int_{\Omega} \alpha \nabla p \cdot v dx = 0 \end{cases}$$

En forme forte ; pour tout t , trouver $u(t)$ dans $\mathbf{H}(\text{curl}, \Omega)$ et $p(t)$ dans $H_0^1(\Omega)$ tels que, pour tout $q(t)$ dans $H_0^1(\Omega)$, on a :

$$\begin{cases} \int_{\Omega} \partial_t p q dx - \int_{\Omega} \alpha \nabla q \cdot u dx = 0 \\ \partial_t u + \alpha \nabla p = 0 \end{cases}$$

Le choix de $\mathbf{H}(\text{curl}, \Omega)$ comme espace de u est justifié par la seconde équation du système. En effet, α étant constant, celle ci nous dit que $\partial_t u$ et ∇p sont dans le même espace. En utilisant la séquence de DeRham, et en observant que p doit être au moins dans $H^1(\Omega)$, on en déduit que $\mathbf{H}(\text{curl}, \Omega)$ est un choix naturel.

De plus, en réutilisant les notations de la section précédente, on se rend compte qu'en cherchant $p_h(t)$ dans $V_h(\text{grad}, \Omega)$ et $u_h(t)$ dans $V_h(\text{curl}, \Omega)$, l'équation forte reste vraie dans le cas discret. C'est à dire :

$$\partial_t u_h + \alpha \nabla p_h = 0$$

En revanche, pour α non constant, même en admettant que $\alpha \nabla p$ est dans $\mathbf{H}(\text{curl}, \Omega)$, rien ne nous permet de dire que $\alpha \nabla p_h$ est dans $V_h(\text{curl}, \Omega)$. Il va donc falloir modifier les formulations variationnelles pour conserver une équation forte dans le cas discret.

4.2 Equations non homogènes

Afin de conserver une equation forte, nous allons légèrement modifier les formulations variationnelles dans le cas discret. Pour se faire, nous allons utiliser des projecteurs. Considérons la formulation variationnelle suivante : trouver u_h dans $V_h(\text{curl}, \Omega)$ et p_h dans $V_h^0(\text{grad}, \Omega)$ tels que, pour tout v_h dans $V_h(\text{curl}, \Omega)$ et q_h dans $V_h^0(\text{grad}, \Omega)$, on a :

$$\begin{cases} \int_{\Omega} \partial_t p_h q_h dx - \int_{\Omega} \Pi^{\text{curl}}(\alpha \nabla q_h) \cdot u_h dx = 0 \\ \int_{\Omega} \partial_t u_h \cdot v_h dx + \int_{\Omega} \Pi^{\text{curl}}(\alpha \nabla p_h) \cdot v_h dx = 0 \end{cases} \quad (7)$$

Où $V_h^0(\text{grad}, \Omega) = V_h(\text{grad}, \Omega) \cap H_0^1(\Omega)$.

Intuitivement lorsque h tend vers 0, les espaces $V_h(\text{curl}, \Omega)$ et $V_h^0(\text{grad}, \Omega)$ "convergent" vers $\mathbf{H}(\text{curl}, \Omega)$ et $H_0^1(\Omega)$ respectivement. Si les projections sont définies de manière raisonnable, on peut donc s'attendre à ce que la solution à ce problème converge vers la solution au problème initial lorsque h tend vers 0.

On peut également remarquer que l'énergie définie précédemment est toujours conservée par cette formulation variationnelle.

Nous pouvons à présent réécrire cette formulation variationnelle sous forme matricielle.

Soit $V_h^0(\text{grad}, \Omega) = \text{Vect}\{\phi_1, \dots, \phi_{n_1}\}$ et $V_h(\text{curl}, \Omega) = \text{Vect}\{\psi_1, \dots, \psi_{n_{\text{curl}}}\}$. Concrètement, les fonctions ϕ_i et ψ_i sont construites avec des produits tensoriels de B-splines. On définit les matrices suivantes :

$$\begin{aligned} M_{i,j}^1 &= \int_{\Omega} \phi_j \phi_i dx \\ M_{i,j}^{\text{curl}} &= \int_{\Omega} \psi_j \cdot \psi_i dx \\ K_{i,j} &= \int_{\Omega} \psi_j \cdot \Pi^{\text{curl}}(\alpha \nabla \phi_i) dx \end{aligned}$$

La formulation variationnelle (7) s'écrit alors :

$$\partial_t \begin{pmatrix} M^{\text{curl}} u_h \\ M^1 p_h \end{pmatrix} = \begin{pmatrix} 0 & K \\ -K^T & 0 \end{pmatrix} \begin{pmatrix} u_h \\ p_h \end{pmatrix} \quad (8)$$

u_h et p_h ne désignant plus les fonctions elles mêmes, mais les coefficients de ces fonctions dans les bases définies plus haut. Les matrices M^1 et M^{curl} étant des matrices de masses, elles sont inversibles et on peut écrire :

$$\partial_t \begin{pmatrix} u_h \\ p_h \end{pmatrix} = \begin{pmatrix} 0 & (M^{\text{curl}})^{-1} K \\ -(M^1)^{-1} K^T & 0 \end{pmatrix} \begin{pmatrix} u_h \\ p_h \end{pmatrix} \quad (9)$$

Etant donné deux conditions initiales u_h^0 et v_h^0 , la formulation variationnelle (7) admet donc une unique solution. En revanche, cette solution s'écrit grace à une exponentielle de matrice. D'un point de vue implémentation, le calcul d'exponentielles de matrices peut s'avérer couteux en temps (d'autant plus que lorsqu'on affine notre discrétisation en espace, les matrices M^1 , M^{curl} et K peuvent avoir de très grandes dimensions). Nous allons donc à présent étudier des schémas en temps pour résoudre l'équation (9) plus rapidement.

4.3 Schémas en temps

L'idée du schéma que nous allons introduire est de séparer chaque incrément de temps en deux étapes. On fait premièrement évoluer uniquement la pression, puis, uniquement la vitesse. On pose :

$$\begin{aligned} A &= \begin{pmatrix} 0 & (M^{curl})^{-1}K \\ -(M^1)^{-1}K^T & 0 \end{pmatrix} \\ A_1 &= \begin{pmatrix} 0 & (M^{curl})^{-1}K \\ 0 & 0 \end{pmatrix} \\ A_2 &= \begin{pmatrix} 0 & 0 \\ -(M^1)^{-1}K^T & 0 \end{pmatrix} \end{aligned}$$

On remarque que $A = A_1 + A_2$, $e^{A_1} = I + A_1$ et $e^{A_2} = I + A_2$. Puis, on définit :

$$\begin{aligned} \phi_u^{\Delta t}(u, p) &= e^{\Delta t A_1} \begin{pmatrix} u \\ p \end{pmatrix} \\ &= \begin{pmatrix} u + \Delta t (M^{curl})^{-1} K p \\ p \end{pmatrix} \\ \phi_p^{\Delta t}(u, p) &= e^{\Delta t A_2} \begin{pmatrix} u \\ p \end{pmatrix} \\ &= \begin{pmatrix} u \\ p - \Delta t (M^1)^{-1} K^T u \end{pmatrix} \end{aligned}$$

Le schéma de Lie splitting est défini de la manière suivante :

$$\begin{aligned} \begin{pmatrix} u_h^{n+1} \\ p_h^{n+1} \end{pmatrix} &= \phi_u^{\Delta t} \circ \phi_p^{\Delta t}(u_h^n, p_h^n) \\ &= (I + \Delta t A_1)(I + \Delta t A_2) \begin{pmatrix} u_h^n \\ p_h^n \end{pmatrix} \end{aligned}$$

Proposition 12. *Le schéma de Lie splitting est d'ordre 1.*

Proof. Soit $T > 0$ et $N \in \mathbb{N}$. On pose $\Delta t = \frac{T}{N}$

$$\begin{aligned}
e^{\Delta t A} &= e^{\Delta t(A_1 + A_2)} \\
&= I + \Delta t A_1 + \Delta t A_2 + \Delta t^2(A_1^2 + A_1 A_2 + A_2 A_1 + A_2^2) + \sum_{i=3}^{\infty} \Delta t^i (A_1 + A_2)^i \\
&= (I + \Delta t A_1)(I + \Delta t A_2) + \Delta t^2 A_2 A_1 + \sum_{i=3}^{\infty} \Delta t^i (A_1 + A_2)^i \\
&= (I + \Delta t A_1)(I + \Delta t A_2) + \mathcal{O}(\Delta t^2)
\end{aligned}$$

On en déduit :

$$\begin{aligned}
\begin{pmatrix} u_h(T) \\ p_h(T) \end{pmatrix} &= e^{TA} \begin{pmatrix} u_h^0 \\ p_h^0 \end{pmatrix} \\
&= (e^{\Delta t A})^N \begin{pmatrix} u_h^0 \\ p_h^0 \end{pmatrix} \\
&= (\phi_u^{\Delta t} \circ \phi_p^{\Delta t})^N (u_h^0, p_h^0) + N \mathcal{O}(\Delta t^2) \\
&= \begin{pmatrix} u_h^N \\ p_h^N \end{pmatrix} + \mathcal{O}(\Delta t)
\end{aligned}$$

□

Proposition 13. Soit $T > 0$ et $N \in \mathbb{N}$. On pose $\Delta t = \frac{T}{N}$. On a alors :

$$|E^N - E^0| = \mathcal{O}(\Delta t)$$

pour $E^n = E(u_h^n, p_h^n)$

La preuve de cette proposition est similaire à la précédente. On commence par montrer que $E^{n+1} - E^n = \mathcal{O}(\Delta t^2)$, puis on en déduit le résultat par inégalité triangulaire sur la somme des $E^{n+1} - E^n$.

Il est également possible de montrer que l'énergie est bornée en temps. i.e :

$$\forall \Delta t, \exists C(\Delta t), \forall n \in \mathbb{N} \quad |E^n - E^0| \leq C(\Delta t)$$

Avec $C(\Delta t) \xrightarrow{h \rightarrow 0} 0$. En revanche, pour démontrer ce résultat, il faut introduire les concepts d'équations Hamiltoniennes et de schémas symplectiques. J'ai brièvement abordé ces concepts lors de ce stage, mais bien trop peu pour être capable de les présenter dans ce rapport. De ce que j'en ai compris, l'idée de la preuve est de montrer que le schéma conserve une autre énergie $\tilde{E}_{\Delta t}$ et que la fonctionnelle $E - \tilde{E}_{\Delta t}$ est bornée sur $V_h(curl, \Omega) \times V_h^0(grad, \Omega)$.

Maintenant que nous avons un schéma en temps, nous pouvons essayer de simplifier les équations obtenues. On a :

$$\begin{aligned} \begin{pmatrix} u_h^{n+1} \\ p_h^{n+1} \end{pmatrix} &= (I + \Delta t A_1)(I + \Delta t A_2) \begin{pmatrix} u_h^n \\ p_h^n \end{pmatrix} \\ \Rightarrow \begin{cases} M^1 p_h^{n+1} &= M^1 p_h^n - \Delta t K^T u_h^n \\ M^{curl} u_h^{n+1} &= M^{curl} u_h^n + \Delta t K p_h^{n+1} \end{cases} \end{aligned}$$

La deuxième équation peut encore être simplifiée. Pour le mettre en évidence, il est préférable de revenir à une forme intégrale de l'équation.

$$\int_{\Omega} u_h^{n+1} \cdot \psi_i dx = \int_{\Omega} u_h^n \cdot \psi_i dx + \Delta t \int_{\Omega} \Pi^{curl}(\alpha \nabla p_h^{n+1}) \cdot \psi_i dx$$

Cette équation étant vraie pour tout i , et $\Pi^{curl}(\alpha \nabla p_h^{n+1})$ étant dans $V_h(curl, \Omega)$ on en déduit :

$$u_h^{n+1} = u_h^n + \Delta t \Pi^{curl}(\alpha \nabla p_h^{n+1})$$

Une fois que l'on connaît p_h^{n+1} , il est donc relativement facile de trouver u_h^{n+1} .

Voici à présent quelques résultats obtenus à partir du schéma de Lie splitting.

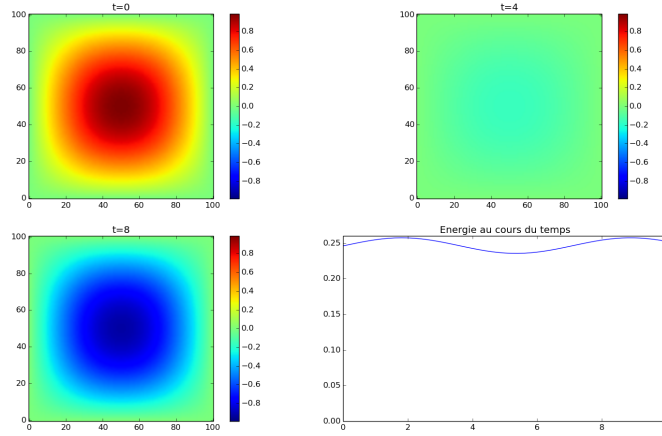


Figure 7: Résolution des équations de l'acoustique pour une onde stationnaire $u_0 = 0$, $p_0 = \sin(\pi x)\sin(\pi y)$, $\alpha = 0.1$, $T = 10$ et $dt = 0.2$

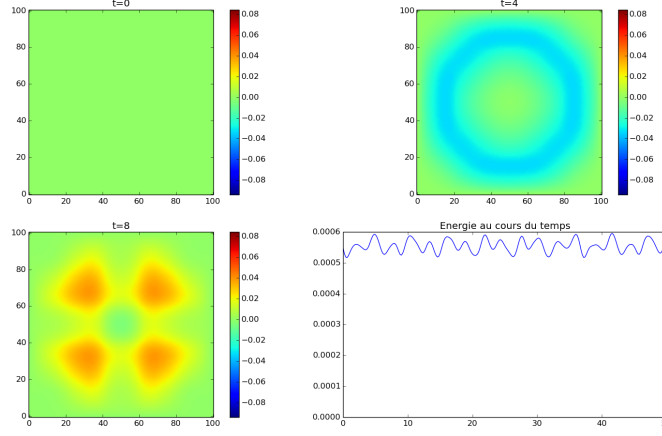


Figure 8: Résolution des équations de l'acoustique pour $u_0^x = -\frac{(x-0.5)}{(1+30((x-0.5)^2+(y-0.5)^2))^2}$, $u_0^y = -\frac{(y-0.5)}{(1+30((x-0.5)^2+(y-0.5)^2))^2}$, $p_0 = 0$, $\alpha = 0.1(1 + (x - 0.5)^2 + (y - 0.5)^2)$, $T = 50$ et $dt = 0.2$

5 Perspectives de travail

5.1 Tests de convergence

Premièrement, il serait intéressant d'effectuer des tests de convergence pour trouver l'ordre en espace et en temps des méthodes utilisées. En revanche, mes codes sont pour l'instant trop lents pour effectuer des tests de convergence corrects sur mon ordinateur.

Pour être capable d'accélérer les calculs, il serait possible d'utiliser l'outil pyccl, développé par mon responsable de stage. Il s'agit d'un outil permettant de convertir des fonctions Python en fonctions Fortran lors de la compilation, pouvant dans beaucoup de cas diminuer considérablement les temps de calcul. J'ai déjà réussi à faire fonctionner pyccl sur certaines fonctions, mais je rencontre encore des problèmes avec les fonctions les plus coûteuses de mes codes.

5.2 Implémentation de mappings

Pour l'instant, ma méthode fonctionne uniquement avec des domaines rectangulaires. En effet, mes espaces de fonctions étant des produits tensoriels, le domaine ne peut être qu'un produit cartésien de deux intervalles. L'idée serait donc de trouver une méthode pour obtenir plus de liberté dans le choix des domaines.

Pour se faire, on introduit un difféomorphisme F , défini sur $\Omega_0 = [0, 1]^2$ à valeurs dans un domaine Ω . Cette fonction permet de passer du domaine physique Ω au domaine logique Ω_0 . De cette manière, il est possible de définir des fonctions de

bases pour nos espaces de fonctions sur Ω , ainsi que de ramener les intégrales sur Ω à des intégrales sur Ω_0 . On remarque cependant que cette méthode permet uniquement d'obtenir des domaines difféomorphes à Ω_0 .

5.3 Généralisation des quasi interpolateurs

Dans ce rapport, j'ai décidé de présenter uniquement les projecteurs construits à partir du quasi interpolateur quadratique à 3 points. Il est en fait possible de définir des quasi interpolateurs pour des splines de degré quelconque, quitte à choisir suffisamment de points pour l'interpolateur local.

Concrètement, il s'agirait de poser le système d'interpolation local pour chacune des B-splines, afin de pouvoir stocker les poids associés à chaque point dans un tableau. On obtiendrait par exemple pour le quasi interpolateur quadratique le tableau suivant :

$$\begin{bmatrix} 1 & -\frac{1}{2} & \dots & -\frac{1}{2} & 0 \\ 0 & 2 & \dots & 2 & 0 \\ 0 & -\frac{1}{2} & \dots & -\frac{1}{2} & 1 \end{bmatrix}$$