

Master CSMI

Méthode Galerkin Discontinu en 1D

Adama DIENG

Superviseur: Joubine Aghili

IRMA - Université de Strasbourg - 2 mois

August 22, 2024

- 1 Introduction
- 2 Méthode de Galerkin Discontinu en 1D
- 3 Résultats numériques en 1D
- 4 Application à l'équation de transport en 2D
- 5 Conclusion et Perspectives
- 6 Références

Introduction

$u'(x) = f(x)$ sur (a, b) avec $u(a) = u_a$ et $u(b) = u_b$

Méthode de Galerkin Continue Classique

- Approximation de $u(x)$ par des fonctions continues;
- Résolution:

$$-\int_a^b u_h(x)v'(x)dx + [u_h(x)v(x)]_a^b = \int_a^b f(x)v(x)dx$$

Méthode de Galerkin Discontinue

- Approximation de $u(x)$ par des fonctions **discontinues**;
- Résolution:

$$-\int_a^b u_h(x)v'(x)dx + \widehat{u}v|_a^b = \int_a^b f(x)v(x)dx$$

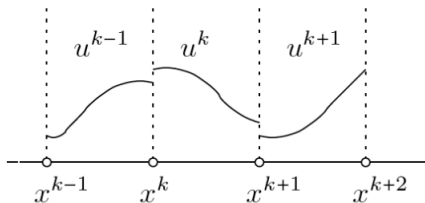
Article de référence

•
Pour l'étude de la méthode de Galerkin Discontinue, on a utilisé l'article de Cockburn[1]
•

Objectif

- Implémenter une methode de type DG pour de type DG pour un problème 1D.
- Comprendre le rôle du flux numérique sur la stabilité d'un schéma DG.

Solution multi-valuée



L'approximation est discontinue !

Figure: approximation discontinue en 1D [2]

Equation Différentielle Ordinaire(EDO)

$$\begin{cases} \frac{d}{dt}u(t) = f(t)u(t) & t \in (0, T) \\ u(0) = u_0 \end{cases} \quad (1)$$

où $f(t)$ est une fonction donnée et u_0 est la condition initiale.

- Partitionner l'intervalle $(0, T)$ en $I^n = (t_n, t_{n+1}) \quad \forall n = 0, \dots, N-2$.
- Trouver u_h de degré au plus k sur chaque intervalle I^n :

$$-\int_{I^n} u_h(s) \frac{d}{ds} v(s) ds + \widehat{u_h} v|_{t_n}^{t_{n+1}} = \int_{I^n} f(s) u_h(s) v(s) ds \quad (2)$$

$$u_h(t) = \sum_{i=0}^k u_i^n \phi_i^n(t) \quad (3)$$

Méthode de Galerkin Discontinu

important!!

- **Problème:** approximations discontinues entre les éléments
- **Conséquence:** solutions instables, incohérentes etc.
- **Solution:** introduire la **trace numérique** \widehat{u}_h

Trace numérique

Définition:

$$\widehat{u}_h(t_n) = \begin{cases} u_0, & \text{si } t_n = 0, \\ \lim_{\epsilon \rightarrow 0} u_h(t_n - \epsilon), & \text{sinon.} \end{cases} \quad (4)$$

Détermination

- Calculée à partir des solutions aux interfaces;
- utilise les valeurs des éléments adjacents;

Stabilité DG

Posant $v = u_h$ dans (2), intégrant par parties et sommant sur n , on obtient:

$$\sum_{n=0}^{N-1} \left(-\frac{1}{2} u_h^2 + \hat{u}_h u_h \right) \Big|_{t_n}^{t_{n+1}} = \frac{1}{2} u_h^2(T^-) + \Theta_h(T) - \frac{1}{2} u_0^2 = \int_0^T f(s) u_h^2(s) d(s)$$

où

$$\Theta_h(T) = -\frac{1}{2} u_h^2(T^-) + \sum_{n=0}^{N-1} \left(-\frac{1}{2} u_h^2 + \hat{u}_h u_h \right) \Big|_{t_n}^{t_{n+1}} + \frac{1}{2} u_0^2$$

Quelle propriété sur $\Theta_h(T)$ pour déterminer explicitement \hat{u}_h ?

$$\frac{1}{2} \|u_h\|_{L^2(0,T)}^2 + \Theta_h(T) \leq \frac{1}{2} \|u_0\|_{L^2(0,T)}^2 \quad (5)$$

[3]

Stabilité DG

Notations:

$$\{u_h\} = \frac{1}{2}(u_h^+ + u_h^-), \quad \llbracket u_h \rrbracket = u_h^+ - u_h^-, \quad u_h^\pm = \lim_{\epsilon \rightarrow 0} u_h(t \pm \epsilon)$$

Définition explicite de \hat{u}_h : Si on prends:

$$\hat{u}_h(t_n) = \begin{cases} u_0, & \text{si } t_n = 0, \\ (\{u_h\} + C^n \llbracket u_h \rrbracket)(t_n), & \text{si } t_n \in (0, T), \\ u_h(T^-), & \text{si } t_n = T, \end{cases} \quad (6)$$

où $C^n \geq 0$, on peut montrer que, avec $C^0 = \frac{1}{2}$:

$$\Theta_h(T) = \sum_{n=1}^{N-1} C^n \llbracket u_h \rrbracket^2(t_n) + \frac{1}{2} \llbracket u_h \rrbracket^2(0) = \sum_{n=0}^{N-1} C^n \llbracket u_h \rrbracket^2(t_n) \geq 0$$

Stabilité DG

Importance des coefficients C^n :

- Rôle des coefficients de C^n :
 - Influence la convergence de la méthode DG;
 - Stabilité garantie pour $C^n \geq 0$;
- Impact sur la Précision:
 - $C^n = 0$: Ordre de précision $2k + 2$;
 - $C^n = \frac{1}{2}$: Ordre de précision $2k + 1$;

discretisation en 1D

- discrétisation de $(0, T)$ en $N - 1$ intervalles I^n :

$$0 = t_0 < t_1 < \dots < t_{N-1} = T$$

$$(0, T) = \bigcup_{n=0}^{N-2} I^n \quad \text{avec } I^n = (t_n, t_{n+1}) \quad \text{pour } n = 0, 1, \dots, N-2$$

- Application de la formulation faible sur chaque élément I^n :

$$\int_{I^n} u_h(s) \frac{d}{ds} v(s) ds + \int_{I^n} f(s) u_h(s) v(s) ds - \hat{u}_h v|_{t_n}^{t_{n+1}} = 0 \quad (7)$$

- Substuer les expressions de $u_h = \sum_{i=0}^P u_i^n \phi_i^n(t)$, de $v = \phi_j^n$ pour $j = 0, 1, 2$ et de \hat{u}_h dans (7).

Matrices Locales

- Sur $I^0 = (t_0, t_1)$, on a :

$$\sum_{i=0}^p \left\{ \int_{I^0} \left(f(s) \phi_i^0(s) + \frac{d}{ds} \phi_i^0(s) \right) \phi_j^0(s) ds + \left(-\frac{1}{2} - C^1 \right) \phi_i^0(t_1^-) \phi_j^0(t_1^-) \right\} \mathbf{u}_i^0$$

$$+ \left(C^1 - \frac{1}{2} \right) \sum_{i=0}^p \phi_i^1(t_1^+) \phi_j^0(t_1^-) \mathbf{u}_i^1 = -u_0 \phi_j^0(t_0^+)$$

Sous forme matricielle, on obtient:

$$\begin{bmatrix} Q_1 & Q_2 & 0 & \dots & 0 \\ 0 & 0 & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \end{bmatrix} \begin{bmatrix} \mathbf{U}^0 \\ \mathbf{U}^1 \\ \vdots \end{bmatrix} = \begin{bmatrix} \mathbf{b} \\ 0 \\ \vdots \end{bmatrix}$$

Remarque: Il faudra noter que Q_1 et Q_2 sont des blocs de matrices $(p \times p)$, et \mathbf{b} est un vecteur de taille p et que U^0 et U^1 sont aussi des vecteurs de taille p .

Matrices Locales

- Sur $I^n = (t_n, t_{n+1})$ pour $n = 1, 2, \dots, N-3$, on a:

$$\begin{bmatrix} 0 & \dots & \dots & \dots & \dots & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & Q1 & Q2 & Q3 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & \dots & \dots & \dots & \dots & 0 \end{bmatrix} \begin{bmatrix} \vdots \\ \mathbf{U}^{n-1} \\ \mathbf{U}^n \\ \mathbf{U}^{n+1} \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ 0 \\ 0 \\ 0 \\ \vdots \end{bmatrix}$$

$$\begin{aligned} (Q_1)_{ij} &= \left\{ \left(\frac{1}{2} + C^n \right) \phi_i^{n-1}(t_n^-) \phi_j^n(t_n^+) \right\} \\ (Q_2)_{ij} &= \left\{ \int_{I^n} \phi_i^n(s) \left(f(s) \phi_j^n(s) + \frac{d}{ds} \phi_j^n(s) \right) ds \right. \\ &\quad \left. + \left(\frac{1}{2} - C^n \right) \phi_i^n(t_n^+) \phi_j^n(t_n^+) - \left(\frac{1}{2} + C^{n+1} \right) \phi_i^n(t_{n+1}^-) \phi_j^{n+1}(t_{n+1}^+) \right\} \\ (Q_3)_{ij} &= - \left(\frac{1}{2} - C^{n+1} \right) \phi_i^{n+1}(t_{n+1}^+) \phi_j^{n+1}(t_{n+1}^+) \end{aligned}$$

Matrices Locales

- Sur le dernier intervalle $I^{N-2} = (t_{N-2}, t_{N-1})$, on a:

Sous forme matricielle, on obtient:

$$\begin{bmatrix} 0 & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & Q1 & Q2 \end{bmatrix} \begin{bmatrix} \vdots \\ \vdots \\ \vdots \\ \mathbf{U}^{N-3} \\ \mathbf{U}^{N-2} \end{bmatrix}$$

$$(Q1)_{ij} = \left\{ \left(\frac{1}{2} + C^{N-2} \right) \phi_i^{N-3}(t_-^{N-2}) \phi_j^{N-2}(t_+^{N-2}) \right\}$$

$$\begin{aligned} (Q2)_{ij} = & \int_{I^{N-2}} \phi_i^{N-2}(s) \left(f(s) \phi_j^{N-2}(s) + \frac{d}{ds} \phi_j^{N-2}(s) \right) ds \\ & + \left(\frac{1}{2} - C^{N-2} \right) \phi_i^{N-2}(t_+^{N-2}) \phi_j^{N-2}(t_+^{N-2}) \\ & - \phi_i^{N-2}(T^-) \phi_j^{N-2}(T^-) \end{aligned}$$

Implémentation pratique

Voici les différentes étapes de l'implémentation:

- **Etape 1:** Initialisation d'une classe, définition des paramètres du problème et des fonctions de base.
- **Etape 2:** Construction des matrices locales $Q1$, $Q2$ et $Q3$ selon chaque intervalle I^n .
- **Etape 3:** Assemblage des matrices locales pour obtenir une matrice globale A .
- **Etape 4:** Résolution du systeme lineaire $A\mathbf{u} = \mathbf{b}$ et estimation des erreurs.

Résultats

Dans le problème (1), on considère les données suivantes:

- $f(t) = -t^2$;
- $u_0 = 1$;
- $T = 5$;
- solution exacte: $u(t) = u_0 e^{\int_0^t f(s) ds}$.

Pour chaque valeur de C^n , on calcule l'erreur L^2 selon le degré k du polynome de base.

Résultats numériques

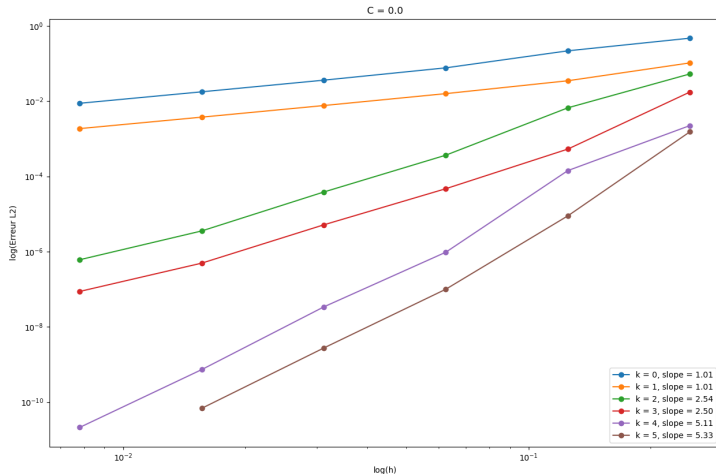


Figure: Erreurs et Pentes pour $C = 0$ pour chaque ordre polynomial k

Resultats numériques

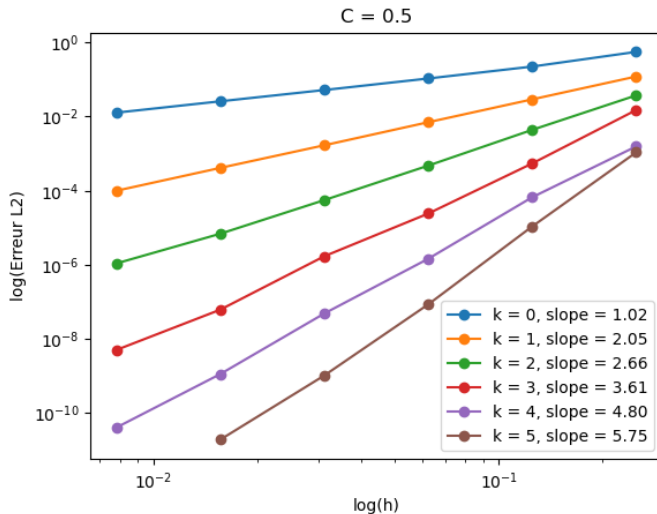


Figure: Erreurs et Pentes pour $C = 0.5$ pour chaque ordre polynomial k

Présentation du problème

$$\begin{cases} \frac{\partial u}{\partial t} + \nabla \cdot (\mathbf{a}u) = 0, & \text{dans } \Omega \times (0, T) \\ u(t=0) = u_0, & \text{dans } \Omega \end{cases} \quad (8)$$

où $\mathbf{a} = (a_x, a_y)$ est un vecteur fixé, $\Omega = [0, 1] \times [0, 1]$.

Formulation faible:

$$\int_K (u_h)_t v - \int_K \mathbf{a}u_h \cdot \nabla v + \int_{\partial K} \widehat{\mathbf{a}u_h} \cdot \hat{\mathbf{n}}^k v = 0 \quad (9)$$

où K est un élément du maillage, $\hat{\mathbf{n}}^k$ est la normale à un coté de K et v est une fonction test.

Maillage du domaine

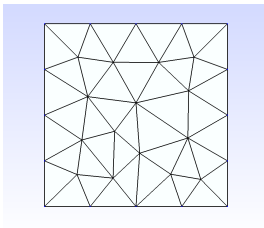


Figure: Maillage en 2D

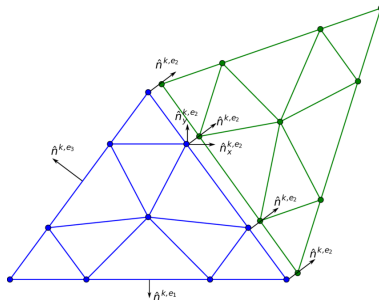


Figure: Normales aux cotés de l'élément [1]

Choix du flux et Stabilité

$$\Theta_h(t) = \sum_{K \in \mathcal{T}_h} \left(-\frac{1}{2} \int_K \nabla \cdot (\mathbf{a} u_h)(x, y, t) dx dy + \int_{\partial K} \widehat{\mathbf{a} u_h} \cdot \mathbf{n} u_h(x, y, t) ds \right) \quad (10)$$

$$\widehat{\mathbf{a} u_h^n} = \mathbf{a} \{u_h\} + C \llbracket u_h \rrbracket$$

$$\text{avec } \llbracket u_h \rrbracket = u_h^- \mathbf{n}^- + u_h^+ \mathbf{n}^+ \quad \text{et} \quad \{u_h\} = \frac{1}{2}(u_h^+ + u_h^-)$$

On obtient alors que:

$$\Theta_h(t) = \sum_{e \in \mathcal{E}_h} \int_e C \llbracket u_h \rrbracket^2 ds \geq 0$$

Ce qui assure la stabilité de la méthode pour le problème (7).

Assemblage des matrices élémentaires

Dans un element K , à un instant t^n , on considère:

$$u_h^n(x, y) = \sum_{i=1}^3 \alpha_i^{K,n} \phi_i^K(x, y) \quad \text{et} \quad v = \phi_j^K(x, y)$$

- **Matrice de masse:**

$$M = (M_{ij}) \quad \text{où} \quad M_{ij} = \int_K \phi_i^K(x, y) \phi_j^K(x, y) dx dy$$

- **Matrice de convection:**

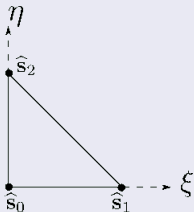
$$S = (S_{ij}) \quad \text{où} \quad S_{ij} = \int_K \mathbf{a} \cdot \nabla \phi_i^K(x, y) \phi_j^K(x, y) dx dy$$

- **Matrice de flux:** Pour chaque bord e_i ($i = 1, 2, 3$), on a:

$$F = (F_{e_i}) \quad \text{où} \quad F_{e_i} = \int_{e_i} \left(\widehat{\mathbf{a} u_h^n} \cdot \hat{\mathbf{n}}^k \right) \cdot \phi_j^K(x, y) ds$$

Triangle de référence

Triangle de référence et fonctions de base



$$\begin{cases} \hat{\phi}_1(\xi, \eta) = 1 - \xi - \eta \\ \hat{\phi}_2(\xi, \eta) = \xi \\ \hat{\phi}_3(\xi, \eta) = \eta \end{cases}$$

Figure: Triangle de référence [1]

Transformation et matrice Jacobienne

$$\begin{pmatrix} x \\ y \end{pmatrix} = T \begin{pmatrix} \xi \\ \eta \end{pmatrix} + \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}$$

$$T = \begin{pmatrix} x_1 - x_0 & x_2 - x_0 \\ y_1 - y_0 & y_2 - y_0 \end{pmatrix}$$

Matrices élémentaires

- **Matrice de masse:**

$$\hat{M} = (\hat{M}_{ij}) \quad \text{où} \quad \hat{M}_{ij} = |J_K| \int_{\hat{K}} \hat{\phi}_i(\xi, \eta) \hat{\phi}_j(\xi, \eta) d\xi d\eta$$

- **Matrice de convection:**

$$\hat{S} = (\hat{S}_{ij}) \quad \text{où} \quad \hat{S}_{ij} = |J_K| \int_{\hat{K}} \mathbf{a} \cdot (T^{-1} \nabla_{\xi, \eta} \hat{\phi}_j(\xi, \eta)) \hat{\phi}_i(\xi, \eta) d\xi d\eta$$

Système linéaire

On arrive au système linéaire suivant:

$$A\mathbf{U}^{n+1} = F$$

- $\mathbf{U}^{n+1} = [\alpha^{K,n+1}, \beta^{K,n+1}, \gamma^{K,n+1}]^T$: vecteur solution à l'instant t^{n+1} ,
- A est la matrice de taille $3N \times 3N$,
- F est le vecteur de taille $3N$ donné par:

$$F = \begin{pmatrix} F_1 \\ F_2 \\ F_3 \end{pmatrix} \qquad F_i = \begin{pmatrix} F_{e_1} \\ F_{e_2} \\ F_{e_3} \end{pmatrix}$$
$$A = \begin{pmatrix} M_1 + S_1 & -F_1 & 0 \\ -F_1 & M_2 + S_2 & -F_2 \\ 0 & -F_2 & M_3 + S_3 \end{pmatrix}$$

où M_i et S_i sont les matrices de masse et d'advection associées à l'élément i respectivement.

Implémentation

- **Calculs sur le triangle de référence**
- **Assemblage des matrices**
- **Calculs des flux** : Difficultés à assembler les matrices de flux aux interfaces.

Codes sources

- https://gitlab.math.unistra.fr/aghili/dg/-/blob/dieng-main-patch-68158/EquationTransport.py?ref_type=heads

Résumé des Contributions :

- Théorie DG : Étude approfondie de la méthode DG, avec focus sur la stabilité L^2 et les flux aux interfaces.
- Implémentation 1D : Reproduction des ordres d'erreurs, résultats sous-optimaux dus au choix des fonctions de base.
- Extension 2D : Défis rencontrés lors de l'extension à l'équation de transport en 2D.

Perspectives :

- Optimisation 1D : Améliorer le choix des fonctions de base pour une meilleure précision.
- Finalisation 2D : Valider la méthode sur des cas simples comme l'advection scalaire constante.
- Applications Futures : Potentiel pour des simulations en mécanique des fluides et autres domaines complexes.



Bharat Tripathi.

Discontinuous Galerkin Method for Propagation of Acoustical Shock Waves in Complex Geometry.

Acoustics [physics.class-ph]. Université Pierre et Marie Curie - Paris VI, 2015.

English. ffNNT : 2015PA066344ff. fftel-01297050f



Bernardo Cockburn.

Discontinuous Galerkin methods.

Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik, 2003, 83 (11), pp.731-754.

10.1002/zamm.200310088. hal-01352444



Delfour, M., W. Hager, and F. Trochu.

“Discontinuous Galerkin Methods for Ordinary Differential Equations.” Mathematics of Computation 36, no. 154 (1981): 455–73.

<https://doi.org/10.2307/2007652>



https://bthierry.pages.math.cnrs.fr/course-fem/_images/reference-triangle.svg



<https://perso.uclouvain.be/vincent.legat/documents/epl1110/annotated-2122-slides/epl1110-cours8-dg-2122.pdf>



Bernardo Cockburn.

An Introduction to the Discontinuous Galerkin Method for
Convection-Dominated Problems
School of Mathematics, University of Minnesota,