



**HAL**  
open science

# Approximation diffuse Hermite et ses applications

Jean-Michel Savignat

► **To cite this version:**

Jean-Michel Savignat. Approximation diffuse Hermite et ses applications. Autre. École Nationale Supérieure des Mines de Paris, 2000. Français. NNT : 2000ENMP0953 . pastel-00577930

**HAL Id: pastel-00577930**

**<https://pastel.hal.science/pastel-00577930>**

Submitted on 17 Mar 2011

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



**ECOLE DES MINES  
DE PARIS**

Collège doctoral

*N° attribué par la bibliothèque*

|  |  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|--|
|  |  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|--|

## **THESE**

pour obtenir le grade de  
**Docteur de l'Ecole Nationale Supérieure des Mines de Paris**  
Spécialité « Technique et Economie de l'Exploitation du Sous-sol »

présentée et soutenue publiquement par

Jean-Michel SAVIGNAT

le 6 octobre 2000

|   |
|---|
| <p><b>APPROXIMATION DIFFUSE HERMITE<br/>ET SES APPLICATIONS</b></p> |
|---|

*Directeurs de thèse: Olivier STAB et Pierre VILLON*

### **Jury**

|               |                  |                          |            |
|---------------|------------------|--------------------------|------------|
| P. BECKERS    | Professeur       | Univ. de Liège           | Rapporteur |
| W. K. LIU     | Professeur       | Northwestern Univ.       | Rapporteur |
| J. BONET      | Reader           | Univ. of Wales, Swansea  | Examineur  |
| G. CAILLETAUD | Professeur       | Ecole des Mines de Paris | Examineur  |
| P. LADEVEZE   | Professeur       | LMT-Cachan               | Examineur  |
| O. STAB       | Maître assistant | Ecole des Mines de Paris | Examineur  |
| P. VILLON     | Professeur       | UTC Compiègne            | Examineur  |



*à mon grand-père, Gaston*



# Remerciements

Je remercie Monsieur Wing Kam Liu et Monsieur Pierre Beckers d'avoir accepté d'être rapporteurs pour ce travail, Monsieur Javier Bonnet, Monsieur George Cailletaud et Monsieur Pierre Ladevèze de participer au jury.

La collaboration avec Olivier Stab qui m'a proposé ce sujet, Pierre Villon dont les conseils me furent précieux, Michel Tijani qui m'a apporté toute sa connaissance, Piotr Breitung et Alain Rassineux, a toujours été pour moi un plaisir.

Je remercie les membres du CGES pour les bons moments que nous avons passés ensemble et en particulier Ielena pour son écoute attentive, Zakaria pour sa grande tolérance et Edith pour son aide précieuse dans la réalisation de ce document.

Je voudrais insister sur le soutien permanent de ma famille et de mes amis sans lequel je n'aurais certainement pas mené à bien mon travail. Je leur en suis très reconnaissant.



# Avant Propos

L'analyse physique de nombreux phénomènes, comme la conduction de la chaleur ou la mécanique des milieux continus (solide ou fluide), mène à des systèmes d'équations différentielles. L'équation de Laplace régit la conduction de la chaleur, les équations de Navier–Stokes l'écoulement des fluides en régime laminaire par exemple. Le problème de l'ingénieur est de trouver la solution de ces équations pour résoudre ses problèmes pratiques : épaisseur d'isolant, dimensionnement d'une canalisation ou d'un pilier de mines, ... Cependant, on ne sait pas, en général, résoudre analytiquement les équations dans les cas réels. Il nous faut donc calculer une solution approchée des équations pour résoudre le problème pratique. Le développement de l'informatique a permis l'emploi de modèles numériques de plus en plus complexes, et les numériciens assurent que la solution numérique approchée est effectivement proche de la solution réelle. Il existe plusieurs techniques, les plus couramment utilisées sont les différences finies (pour simuler l'écoulement de pétrole dans le sous-sol par exemple), les éléments finis (mécanique du solide), les volumes finis (mécanique des fluides) ou les éléments frontières (conduction de la chaleur). Ces quatre techniques s'appuient sur un même formalisme mathématique.

L'équation différentielle est transformée par la technique des résidus pondérés (ou approche variationnelle, [DT84, ZT94]) où la solution est cherchée dans un espace de solutions admissibles tel que la formulation variationnelle soit vérifiée pour toutes les fonctions d'un espace test. Cette technique a été proposée pour la première fois par Galerkin dans [Gal15]. Un choix judicieux des espaces de solutions admissibles ( $\mathcal{H}$ ) et de test ( $\mathcal{V}$ ) permet de construire chacune des quatre méthodes citées comme le montre [DT84]. Cependant, le problème variationnel ne se résout facilement que lorsque les deux espaces fonctionnels sont de dimensions finies. On doit donc souvent construire des sous-espaces de dimensions finies des espaces  $\mathcal{H}$  et  $\mathcal{V}$  (respectivement  $\mathcal{H}_n$  et  $\mathcal{V}_n$ ), et calculer une solution approchée de la solution réelle dans ces sous-espaces. L'erreur sur la solution du problème variationnel est directement liée à la capacité de l'espace  $\mathcal{H}_n$  à approximer la solution. La construction de  $\mathcal{H}_n$  est donc un point crucial. La technique de construction de  $\mathcal{H}_n$  la plus employée est l'interpolation par sous-domaines; le choix  $\mathcal{V}_n = \mathcal{H}_n$  conduit dans ce cas à la méthode des éléments finis.

Depuis le début des années 90, des techniques alternatives de construction de  $\mathcal{H}_n$  sont en vogue. Ces techniques trouvent leurs origines à la fin des années 70 et les principales sont les méthodes *meshless* ou sans-maillage :

- FDM (Finite Difference Method), avec semis de nœuds arbitraire [LO80].
- SPH (Smooth Particle Hydrodynamics) [Luc77].
- DEM (Diffuse Element Method) [NTV92].
- EFG (Element Free Galerkin method) [BLG94].
- RPKM (Reproducing Kernel Particle Method) [LJZ95].



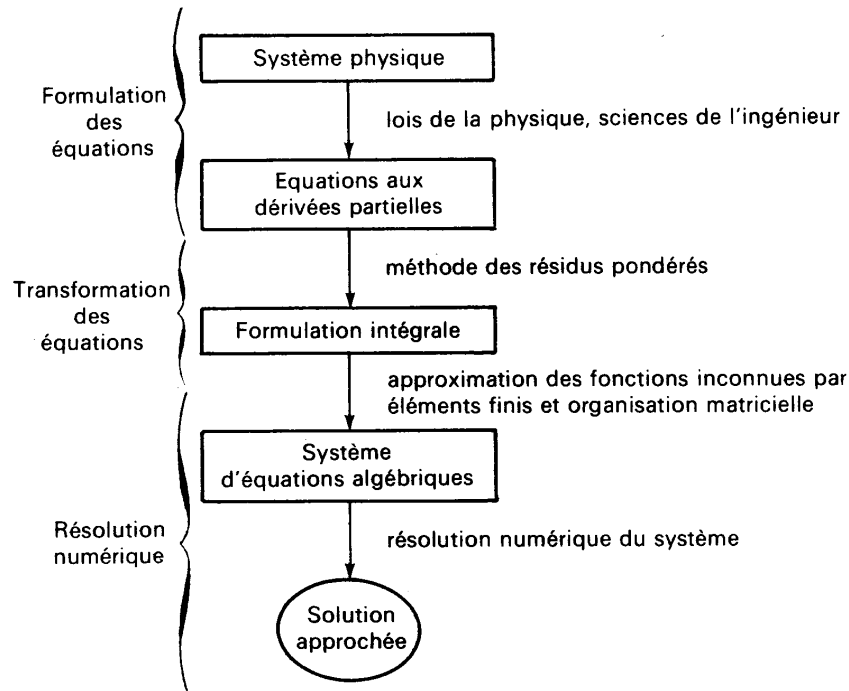


FIG. 0.1: Du système physique à une solution approchée de son comportement (d'après [DT84]). L'approximation est le pivot qui permet de passer du problème continu au problème discret que l'on sait résoudre numériquement.

- PUFEM (Partition of Unity Finite Element Method) [Mel95].
- Hp-Clouds [LDT96].
- et d'autres ...

Toutes ces méthodes s'appuient sur une partition de l'unité pour construire un espace  $\mathcal{H}_n$  avec de « bonnes » propriétés. Dans ses travaux, Melenk [Mel95] insiste particulièrement sur ce point pour introduire une connaissance *a priori* sur la solution de l'équation à résoudre. En général, la technique sous-jacente est l'approximation diffuse (ou MLS : *moving least square approximation*). Cette technique a été développée à plusieurs reprises dans divers domaines<sup>1</sup> :

- Pour l'interpolation par Lancaster et Salkauskas [LS80].
- Pour les statistiques comme une composante de l'algorithme LOWESS [FG96].
- Pour les équations aux dérivées partielles par [LO80], [Vil91] et [LJZ95].

Il existe deux formes d'approximation diffuse, l'une discrète [LO80, LS80, Vil91] utilisée dans les méthodes FDM, DEM, EFG. L'autre continue [Vil91, LJZ95] utilisée dans RPKM par exemple.

Le grand intérêt de l'approximation diffuse est de cacher la phase de découpage de la géométrie (maillage) derrière la notion de partition de l'unité avec pour objectif de construire une méthode aussi flexible que les éléments finis mais plus facile d'emploi. Bien entendu, ces nouvelles définitions de  $\mathcal{H}_n$  demandent d'adapter les

1. Il semble que ces divers développements ont été indépendants.

différentes étapes et les différents outils de construction d'un modèle numérique, comme le maillage, l'intégration, ou la prise en compte des conditions aux limites.

Une grande part de l'activité de recherche du CGES<sup>2</sup> est basée sur la simulation numérique de phénomènes physiques (mécanique des roches, abattage mécanique ou par explosif par exemple). Aussi ce centre de recherche est-il un gros consommateur de modèles numériques et le logiciel VIPLEF concentre plus de 20 ans d'expérience dans le développement de code éléments finis. Nous avons donc décidé de nous pencher sur les nouvelles méthodes de simulation en collaboration avec l'équipe de recherche qui a développé l'approximation diffuse et les éléments diffus [NTV92] à l'UTC<sup>3</sup>.

L'approximation diffuse est le point central de la plupart des méthodes, aussi la première étape a été de l'analyser. L'absence de maillage et les fonctions de pondération nous ont poussés à la comparer au krigeage et ses covariances généralisées. La comparaison emploie le théorème d'équivalence splines-krigeage: on s'appuie sur les splines et le formalisme de Wahba [Wah90] pour construire une nouvelle méthode d'approximation. L'*approximation myope* est une méthode hybride dont des cas particuliers sont les splines d'ajustement ou l'approximation diffuse. Ce travail s'inspire de l'article de Salkauskas [Sal92], mais nous insistons ici sur le contexte de construction de l'interpolateur myope et les transformations nécessaires pour obtenir une interpolation locale. Les propriétés de l'approximation myope sont étudiées et on montre qu'elle suit les mêmes schémas que l'approximation diffuse (consistance, fonctions de forme, interpolation).

Nous avons aussi étudié l'approximation diffuse de type Hermite proposée pour la première fois pour la prise en compte de conditions aux limites de type Neumann dans [LDT96]. Nous montrons que cette forme de l'approximation diffuse admet les propriétés usuelles (consistance, fonction de forme, interpolation) et démontrons un théorème de convergence.

Dans un second temps, nous avons mis en œuvre l'approximation diffuse et sa pseudo-dérivation. Cette technique est très efficace pour les lissages et dérivations numériques comme le montre l'usage quotidien du logiciel réalisé `adlis` par les membres du CGES. Ces résultats nous ont poussés à utiliser la pseudo-dérivation et l'approximation Hermite pour estimer la courbure de surfaces triangulées [SRSV99]. La qualité des résultats obtenus nous permet de reconnaître un certain nombre de surfaces élémentaires grâce à un algorithme simple et rapide [SSRV99]. L'originalité de l'algorithme tient dans l'utilisation active de la topologie (maillage); ceci nous permet de spécifier les hypothèses sous lesquelles il reconnaît toute la surface. Le calcul diffus de la courbure est aussi à la base d'un algorithme de remaillage [RSSV99].

Enfin, nous avons développé une méthode d'intégration numérique robuste pour les équations aux dérivées partielles. Comme la prise en compte des conditions aux limites reste un problème ouvert, nous travaillons ici en dimension 1. L'analyse du patch test de [Lao96, Kro96] conduit à la définition d'une intégration du type Gauss où les points d'intégration sont définis comme les points de Legendre et les poids d'intégration sont calculés sur chaque support d'intégration. Le schéma de base est décrit dans [BTV99c] et nous l'étendons pour le calcul de poutre sous les hypothèses de Kirchhoff (la section droite reste droite) avec l'approximation diffuse de type Hermite. Ceci nous a permis de développer une librairie de simulation des corps élancés basée sur l'approximation diffuse. Cette librairie est ensuite utilisée

---

2. Centre de Géotechnique et d'Exploitation du Sous-sol

3. Université de technologie de Compiègne

pour résoudre des problèmes réels posés au CGES.

Les contributions à la connaissance de l'approximation diffuse et son utilisation touchent ainsi plusieurs domaines :

- **L'approximation** avec la construction et l'étude de l'approximation myope, et l'étude de l'approximation diffuse Hermite.
- **La modélisation géométrique** avec le calcul de courbure et l'algorithme de reconnaissance des surfaces.
- **La construction de logiciel de simulation** avec la mise en œuvre pratique (choix techniques, étude de sensibilité) du schéma d'intégration.
- **La simulation pratique** avec la librairie poutre.

Le plan de ce mémoire suit les différentes étapes que nous venons de décrire.

Dans une première partie, l'approximation diffuse est décrite en détail dans ses formulations continue et discrète. Nous présentons ensuite la famille splines–krigeage en insistant plus sur les idées à la base de ces méthodes.

La seconde partie regroupe les travaux sur la méthode d'approximation elle-même. L'interpolateur myope est présenté et permet de mieux comprendre la profonde différence qui sépare moindres carrés glissants et splines. L'approximation diffuse Hermite est construite et analysée en suivant le schéma de présentation de la première partie.

La troisième partie montre l'utilisation de l'approximation Hermite pour la modélisation surfacique. Le mode de calcul de la courbure est présenté et analysé. Nous développons ensuite l'algorithme de reconnaissance des surfaces. Cette partie contient aussi la bibliographie spécifique à la courbure et son utilisation pour la reconnaissance des surfaces.

La quatrième partie aborde les équations aux dérivées partielles. Nous commençons par une étude bibliographique synthétique des méthodes sans maillage existantes, et dégageons deux faiblesses communes à toutes les méthodes : l'intégration numérique d'une part et la prise en compte des conditions aux limites d'autre part. Nous avons choisi de nous pencher sur l'intégration numérique car on peut facilement tester les idées en 1d ; de plus l'étude des conditions aux limites suppose que l'on dispose déjà d'un schéma d'intégration performant ce qui n'était pas le cas avant ce travail. Nous décrivons dans un premier temps le schéma d'intégration et les tests effectués. Ceci nous permet de construire une librairie pour le calcul de corps élancés et de l'appliquer à des problèmes réels.

Chaque étape de cette étude a demandé des développements informatiques. Après trois ans de thèse, la librairie de recherche ressemble un peu à la tour de Babel, mais l'expérience acquise dans son écriture nous a permis d'en extraire une librairie d'approximation DIAM et la librairie de calcul de poutre ADHEL.

Les annexes au volume principal regroupent à la fois des résultats numériques (chapitres D, E, I) et quelques développements importants (chapitres A, B, F, H, G). L'annexe C référence les surfaces utilisées dans la troisième partie.

# Table des matières

|   |           |
|---|-----------|
| Remerciements   | v         |
| Avant Propos  | vii       |
| <b>I Approximation sans maillage</b>                          | <b>1</b>  |
| <b>1 Introduction</b>   | <b>3</b>  |
| <b>2 Approximation Diffuse</b>                                | <b>13</b> |
| 2.1 Définitions et notations . . . . .                        | 13        |
| 2.1.1 Opérateurs de Lissage . . . . .                         | 13        |
| 2.1.2 Fonction d'atténuation ou d'influence . . . . .         | 14        |
| 2.2 Lissage par moindres carrés – Cas continu . . . . .       | 15        |
| 2.2.1 Opérateur $\Pi$ . . . . .                               | 15        |
| 2.2.2 Expression centrée . . . . .                            | 16        |
| 2.2.3 Propriétés . . . . .                                    | 17        |
| 2.2.4 Pseudo-dérivation . . . . .                             | 17        |
| 2.2.5 Convergence . . . . .                                   | 18        |
| 2.2.6 <i>Reproducing Kernel Particle</i> . . . . .            | 18        |
| 2.3 Lissage par moindres carrés – Cas discret . . . . .       | 19        |
| 2.3.1 Introduction . . . . .                                  | 19        |
| 2.3.2 Lisseur discret . . . . .                               | 21        |
| 2.3.3 Interpolation . . . . .                                 | 21        |
| 2.3.4 Forme centrée, propriétés . . . . .                     | 22        |
| 2.3.5 Pseudo-dérivation . . . . .                             | 23        |
| 2.3.6 Convergence . . . . .                                   | 25        |
| 2.3.7 Fonctions de forme de l'approximation diffuse . . . . . | 26        |
| 2.3.8 Fonction d'atténuation . . . . .                        | 27        |
| 2.4 Conclusion . . . . .                                      | 27        |
| <b>3 Interpolateurs Radiaux</b>                               | <b>31</b> |
| 3.1 Introduction . . . . .                                    | 31        |
| 3.1.1 Radial Basis Functions . . . . .                        | 31        |
| 3.1.2 Le krigeage ponctuel . . . . .                          | 32        |
| 3.1.3 Splines . . . . .                                       | 32        |
| 3.1.4 Fonctions de type positif conditionnel . . . . .        | 33        |
| 3.2 Choix du noyau . . . . .                                  | 33        |
| 3.2.1 Cas du krigeage . . . . .                               | 33        |
| 3.2.2 Splines . . . . .                                       | 35        |
| 3.2.3 Radial Basis Functions . . . . .                        | 35        |
| 3.2.4 Validation croisée . . . . .                            | 35        |

|   |  |           |
|---|--|-----------|
| 3.3   | Données bruitées . . . . .   | 36        |
| 3.3.1   | Cokrigage d'erreur . . . . .   | 37        |
| 3.3.2   | Splines d'ajustement . . . . .   | 38        |
| 3.4   | Fonctions de forme . . . . .   | 38        |
| 3.4.1   | Existence des fonctions de forme . . . . .                                     | 40        |
| 3.4.2   | Comportement des poids de krigeage . . . . .                                   | 40        |
| 3.4.3   | Quelques fonctions de forme en 2d . . . . .                                    | 42        |
| 3.5   | Conclusion . . . . .   | 42        |
| <b>II Extensions de l'approximation diffuse</b> |  | <b>45</b> |
| <b>1</b>  | <b>Approximation Myope</b>   | <b>47</b> |
| 1.1   | Cadre Fonctionnel . . . . .  | 47        |
| 1.2   | Structure de Hilbert des espaces de Sobolev $\mathcal{H}^m$ . . . . .          | 48        |
| 1.2.1   | Définitions . . . . .  | 48        |
| 1.2.2   | Produits scalaires de $\mathcal{H}_1^m$ et $\mathcal{H}_2^m$ . . . . .         | 49        |
| 1.2.3   | Somme directe de $\mathcal{H}_1^m$ et $\mathcal{H}_2^m$ . . . . .              | 50        |
| 1.2.4   | $\mathcal{H}^m$ est un espace de Hilbert . . . . .                             | 50        |
| 1.3   | Structure de r.k.h.s. des espaces de Sobolev $\mathcal{H}^m$ . . . . .         | 51        |
| 1.3.1   | $\mathcal{H}^m$ est un r.k.h.s. . . . .  | 51        |
| 1.3.2   | Représentant des formes linéaires $L_i$ . . . . .                              | 52        |
| 1.4   | Splines d'ajustement . . . . .   | 54        |
| 1.4.1   | Introduction . . . . .   | 54        |
| 1.4.2   | Transformation de $\mathcal{E}_W^\lambda$ et calcul de $f_W^\lambda$ . . . . . | 54        |
| 1.4.3   | Régularité du système d'optimalité . . . . .                                   | 56        |
| 1.4.4   | Cas particuliers, consistance . . . . .  | 56        |
| 1.4.5   | Système équivalent . . . . .   | 57        |
| 1.5   | Approximation Myope . . . . .  | 60        |
| 1.5.1   | Définitions . . . . .  | 60        |
| 1.5.2   | Propriétés . . . . .   | 61        |
| 1.5.3   | Interpolation myope . . . . .  | 62        |
| 1.5.4   | Approximation Myope et Approximation Diffuse . . . . .                         | 63        |
| 1.5.5   | Approximation diffuse pénalisée . . . . .                                      | 63        |
| 1.6   | Expérimentation numérique . . . . .  | 65        |
| 1.6.1   | Fonctions de forme . . . . .   | 66        |
| 1.6.2   | Fonction sinus . . . . .   | 68        |
| 1.6.3   | Influence de $\lambda$ . . . . .   | 70        |
| 1.7   | Conclusion . . . . .   | 73        |
| <b>2</b>  | <b>Approximation Hermite</b>   | <b>75</b> |
| 2.1   | Construction et propriétés . . . . .   | 75        |
| 2.1.1   | Construction de l'approximation . . . . .                                      | 75        |
| 2.1.2   | Fonctions de forme . . . . .   | 77        |
| 2.1.3   | Interpolation . . . . .  | 78        |
| 2.2   | Convergence . . . . .  | 79        |
| 2.3   | Expérimentation numérique . . . . .  | 80        |
| 2.3.1   | Topo-géométrie . . . . .   | 80        |
| 2.3.2   | Exemple . . . . .  | 81        |
| 2.3.3   | Fonctions de forme . . . . .   | 83        |
| 2.3.4   | Convergence . . . . .  | 85        |
| 2.4   | Conclusion . . . . .   | 85        |

|            |   |            |
|------------|---|------------|
| <b>III</b> | <b>Application à la modélisation surfacique</b>                   | <b>87</b>  |
| <b>1</b>   | <b>Courbure d'une surface triangulée</b>                          | <b>89</b>  |
| 1.1        | Courbure d'une surface 3d . . . . .                               | 91         |
| 1.2        | Calcul de la courbure . . . . .                                   | 92         |
| 1.2.1      | Méthodes géométriques . . . . .                                   | 92         |
| 1.2.2      | Méthodes par ajustement . . . . .                                 | 93         |
| 1.2.3      | Différences finies . . . . .                                      | 94         |
| 1.2.4      | Facettes . . . . .  | 94         |
| 1.2.5      | Cas des surfaces triangulées . . . . .                            | 94         |
| 1.3        | Calcul diffus de la courbure . . . . .                            | 95         |
| 1.3.1      | Voisinage . . . . .   | 95         |
| 1.3.2      | Pondérateurs . . . . .  | 96         |
| 1.3.3      | Premiers résultats . . . . .                                      | 97         |
| 1.3.4      | Prise en compte des normales aux faces . . . . .                  | 100        |
| 1.3.5      | Calcul du paramètre de réglage $\lambda$ . . . . .                | 102        |
| 1.3.6      | Robustesse de l'algorithme avec normales . . . . .                | 103        |
| 1.3.7      | Calcul « quasi-dual » . . . . .                                   | 107        |
| 1.4        | Comparaison avec la méthode de Samson . . . . .                   | 109        |
| 1.4.1      | Méthode de Samson . . . . .                                       | 109        |
| 1.4.2      | Comparaison expérimentale . . . . .                               | 110        |
| 1.5        | Conclusion . . . . .  | 111        |
| <b>2</b>   | <b>Reconnaissance des surfaces</b>                                | <b>113</b> |
| 2.1        | Courbure et reconnaissance des surfaces . . . . .                 | 113        |
| 2.1.1      | Le Graphe Global des Courbures . . . . .                          | 113        |
| 2.1.2      | Bibliographie . . . . .   | 114        |
| 2.2        | Présentation de l'algorithme proposé . . . . .                    | 115        |
| 2.2.1      | Hypothèse de travail/ Présentation générale . . . . .             | 115        |
| 2.2.2      | Classification initiale . . . . .                                 | 116        |
| 2.2.3      | Propagation . . . . .   | 119        |
| 2.2.4      | Post-traitements . . . . .  | 121        |
| 2.2.5      | Démonstration de l'algorithme . . . . .                           | 121        |
| 2.3        | Robustesse, généralisation . . . . .                              | 124        |
| 2.3.1      | Classement initial (discussion de H3) . . . . .                   | 124        |
| 2.3.2      | Propagation (discussion de H2) . . . . .                          | 125        |
| 2.3.3      | Discrétisation (discussion de H1) . . . . .                       | 125        |
| 2.3.4      | Cas réels traités . . . . .                                       | 126        |
| 2.4        | Conclusion . . . . .  | 126        |
| <b>IV</b>  | <b>Éléments Diffus</b>  | <b>133</b> |
| <b>1</b>   | <b>Méthodes sans maillage</b>                                     | <b>135</b> |
| 1.1        | Introduction . . . . .  | 136        |
| 1.1.1      | Résidus pondérés . . . . .  | 136        |
| 1.1.2      | Formulation variationnelle de l'équation de la chaleur . . . . .  | 141        |
| 1.1.3      | Crimes variationnels . . . . .                                    | 142        |
| 1.1.4      | Le patch test . . . . .   | 142        |
| 1.2        | Choix d'une méthode d'approximation . . . . .                     | 143        |
| 1.2.1      | Smooth particle hydrodynamics (SPH) . . . . .                     | 143        |
| 1.2.2      | Moindres carrés glissants . . . . .                               | 144        |
| 1.2.3      | hp-coulds et <i>éléments finis généralisés</i> ou PUFEM . . . . . | 144        |
| 1.2.4      | Autres techniques . . . . .                                       | 145        |

|          |   |            |
|----------|---|------------|
| 1.3      | Raffinement de la méthode d'interpolation . . . . .                 | 146        |
| 1.3.1    | Moindres carrés glissants . . . . .                                 | 146        |
| 1.3.2    | Consistance . . . . .   | 148        |
| 1.3.3    | Analyse multi-échelle . . . . .                                     | 149        |
| 1.3.4    | Base hiérarchique, <i>hierarchical partition of unity</i> . . . . . | 151        |
| 1.4      | Formulation . . . . .   | 153        |
| 1.4.1    | Formulations pour méthodes sans maillage . . . . .                  | 153        |
| 1.4.2    | Conditions aux limites . . . . .                                    | 153        |
| 1.5      | Intégration . . . . .   | 154        |
| 1.5.1    | Pavage d'intégration . . . . .                                      | 155        |
| 1.5.2    | Intégration nodale . . . . .  | 157        |
| 1.5.3    | Le patch test . . . . .   | 160        |
| 1.6      | Conclusion . . . . .  | 162        |
| <b>2</b> | <b>Intégration numérique</b> . . . . .                              | <b>165</b> |
| 2.1      | Problèmes types . . . . .   | 165        |
| 2.1.1    | Problème de Dirichlet . . . . .                                     | 165        |
| 2.1.2    | Modes propres . . . . .   | 166        |
| 2.2      | Construction d'une intégration numérique . . . . .                  | 167        |
| 2.2.1    | Construction . . . . .  | 167        |
| 2.2.2    | Propriétés du système résultant . . . . .                           | 168        |
| 2.2.3    | Résolution du système d'optimalité . . . . .                        | 169        |
| 2.3      | Méthodes testées . . . . .  | 170        |
| 2.3.1    | Construction des fonctions d'influence . . . . .                    | 170        |
| 2.3.2    | Liste des méthodes testées . . . . .                                | 173        |
| 2.4      | Analyse des résultats . . . . .                                     | 175        |
| 2.4.1    | Critères de comparaison . . . . .                                   | 175        |
| 2.4.2    | Intégration, première analyse . . . . .                             | 176        |
| 2.4.3    | Sensibilité aux différents paramètres . . . . .                     | 176        |
| 2.4.4    | Convergence pour le problème de Dirichlet . . . . .                 | 183        |
| 2.4.5    | Convergence pour le calcul des modes propres . . . . .              | 185        |
| 2.5      | Conclusion . . . . .  | 186        |
| <b>3</b> | <b>Poutre diffuse</b> . . . . .                                     | <b>187</b> |
| 3.1      | Description du problème . . . . .                                   | 188        |
| 3.1.1    | Rappel de géométrie différentielle des courbes . . . . .            | 188        |
| 3.1.2    | Équilibre d'une poutre . . . . .                                    | 189        |
| 3.1.3    | Formulation variationnelle . . . . .                                | 190        |
| 3.1.4    | Hypothèse de Kirchhoff . . . . .                                    | 190        |
| 3.2      | Patch test, approximation . . . . .                                 | 191        |
| 3.2.1    | Le patch test . . . . .   | 191        |
| 3.2.2    | Approximation utilisée . . . . .                                    | 192        |
| 3.2.3    | Modèle de poutre discret . . . . .                                  | 192        |
| 3.3      | Intégration numérique . . . . .                                     | 194        |
| 3.3.1    | Patch test pour une équation d'ordre $2m$ . . . . .                 | 194        |
| 3.3.2    | Analyse des contraintes pour un schéma de Gauss modifié . . . . .   | 195        |
| 3.3.3    | Test pour la flexion . . . . .                                      | 196        |
| 3.3.4    | Mise en œuvre pour les poutres . . . . .                            | 196        |
| 3.4      | Tests numériques . . . . .  | 197        |
| 3.4.1    | Intégration (flexion) . . . . .                                     | 197        |
| 3.4.2    | Cas tests (flexion) . . . . .                                       | 198        |
| 3.4.3    | Poutres courbes . . . . .   | 199        |
| 3.4.4    | Discussion des résultats . . . . .                                  | 199        |
| 3.5      | Conclusion . . . . .  | 201        |

|           |   |            |
|-----------|---|------------|
| <b>4</b>  | <b>Application à un problème de forage</b>                            | <b>203</b> |
| 4.1       | Problème posé . . . . .   | 203        |
| 4.1.1     | Description du problème physique . . . . .                            | 203        |
| 4.1.2     | Modélisation, géométrie . . . . .                                     | 204        |
| 4.2       | Contact 2d . . . . .  | 205        |
| 4.2.1     | Description de l'algorithme . . . . .                                 | 205        |
| 4.2.2     | Exemple . . . . .   | 206        |
| 4.2.3     | Discussion . . . . .  | 207        |
| 4.3       | Contact 3d . . . . .  | 210        |
| 4.3.1     | Description . . . . .   | 210        |
| 4.3.2     | Calcul élémentaire des directions . . . . .                           | 210        |
| 4.3.3     | Mise en œuvre . . . . .   | 212        |
| 4.4       | Conclusion . . . . .  | 212        |
| <b>V</b>  | <b>Conclusion</b>   | <b>215</b> |
| <b>VI</b> | <b>Annexes</b>  | <b>221</b> |
| <b>A</b>  | <b>Calcul de l'approximation diffuse</b>                              | <b>223</b> |
| A.1       | Cas 1d, dérivée réelle . . . . .                                      | 223        |
| A.2       | Algorithme de Breïtkopf . . . . .                                     | 227        |
| A.3       | Complexités . . . . .   | 228        |
| <b>B</b>  | <b>Construction de la Topo-géométrie</b>                              | <b>229</b> |
| B.1       | Aspects généraux . . . . .  | 229        |
| B.1.1     | Construction par croissance . . . . .                                 | 229        |
| B.1.2     | Voisinage asymétrique . . . . .                                       | 230        |
| B.2       | Cas 1d: voisinage de type Voronoï . . . . .                           | 231        |
| B.2.1     | Consistance linéaire . . . . .  | 231        |
| B.2.2     | Cas général . . . . .   | 232        |
| <b>C</b>  | <b>Surfaces utilisées dans les tests</b>                              | <b>233</b> |
| C.1       | Modèle de vache . . . . .   | 233        |
| C.2       | Cylindres . . . . .   | 234        |
| C.3       | Surface PSA . . . . .   | 235        |
| C.4       | Surface Sony . . . . .  | 237        |
| C.5       | Surface French . . . . .  | 239        |
| C.6       | Surface Benz . . . . .  | 240        |
| <b>D</b>  | <b>Convergence de l'approximation Hermite</b>                         | <b>243</b> |
| <b>E</b>  | <b>Équation de la chaleur</b>   | <b>253</b> |
| E.1       | Mesures de la qualité de répartition d'un semis de nœuds 1d . . . . . | 253        |
| E.2       | Résultats expérimentaux . . . . .                                     | 254        |
| E.2.1     | Influence du semis de nœuds . . . . .                                 | 254        |
| E.2.2     | Dirichlet – Problème 1 . . . . .                                      | 258        |
| E.2.3     | Dirichlet – Problème 3 . . . . .                                      | 262        |
| E.2.4     | Dirichlet – Problème 2 . . . . .                                      | 270        |
| E.2.5     | Dirichlet – Problème 4 . . . . .                                      | 274        |
| E.2.6     | Dirichlet – Problème 5 . . . . .                                      | 278        |
| E.2.7     | Dirichlet – Problème 6 . . . . .                                      | 282        |
| E.2.8     | Valeurs Propres . . . . .   | 286        |



|   |            |
|---|------------|
| <b>F Patch-test numérique</b>                           | <b>289</b> |
| F.1 Introduction . . . . .                              | 289        |
| F.1.1 Les positions . . . . .                           | 290        |
| F.1.2 La matrice de raideur . . . . .                   | 290        |
| F.2 Tests pour comprendre le phénomène . . . . .        | 290        |
| F.2.1 Cas homogène . . . . .                            | 290        |
| F.2.2 Cas hétérogène . . . . .                          | 291        |
| F.2.3 Test du solveur de système linéaire . . . . .     | 291        |
| F.3 Analyse . . . . .                                   | 291        |
| F.3.1 Norme énergie . . . . .                           | 291        |
| F.3.2 Influence des paramètres . . . . .                | 293        |
| F.3.3 Système linéaire . . . . .                        | 293        |
| F.4 Conclusion . . . . .                                | 295        |
| <b>G Poutre circulaire et hélicoïdale</b>               | <b>297</b> |
| G.1 Cas d'une poutre en arc de cercle . . . . .         | 297        |
| G.1.1 Problème posé . . . . .                           | 297        |
| G.1.2 Solution analytique . . . . .                     | 298        |
| G.1.3 Conclusion . . . . .                              | 300        |
| G.2 Cas d'une poutre en hélice . . . . .                | 300        |
| G.2.1 Géométrie . . . . .                               | 300        |
| G.2.2 Résultat . . . . .                                | 300        |
| G.2.3 Conclusion . . . . .                              | 301        |
| <b>H Approximation diffuse trigonométrique</b>          | <b>303</b> |
| H.1 Approximation diffuse – cas général . . . . .       | 303        |
| H.2 Approximation diffuse « trigonométrique » . . . . . | 304        |
| <b>I Convergence des éléments diffus Hermite</b>        | <b>307</b> |

Première partie

**Approximation sans maillage**



# Chapitre 1

## Introduction

Au moment de commencer mon travail de thèse, une question me semblait essentielle: « Quel scientifique n'a pas utilisé dans son travail les mots « interpolation », « approximation » ou « lissage »? » Je pense aujourd'hui que tous les scientifiques utilisent ces termes.

Ce constat laisse apparaître combien il est difficile de donner un sens précis à ces mots d'un emploi si courant et qui regroupent des réalités différentes selon la discipline, et même parfois selon les personnes! Il me semble donc nécessaire avant tout de faire le point sur quelques mots de vocabulaire. Les définitions qui suivent ont pour seul but de lever toute ambiguïté quant à l'emploi de ces termes dans le manuscrit.

**Globale :** Une propriété est globale si elle dépend de tout le domaine d'étude, c'est le cas de l'équilibre d'une structure par exemple.

**Locale :** Une propriété est locale si elle est vérifiée sur le voisinage d'un point.

**Ponctuelle :** Une propriété est ponctuelle si elle ne dépend que d'un point de données particulier. C'est le cas des critères de plasticité usuels par exemple.

**Ajustement :** L'ajustement consiste à calculer les paramètres d'un modèle pour qu'il décrive au mieux des mesures.

**Nœud :** C'est un point géographique pour lequel on connaît le champ de données.

**Point d'évaluation :** C'est un point où l'on évalue une approximation du champ. Un point d'évaluation peut être confondu avec un nœud.

**Approximation :** L'approximation d'un champ est une fonction qui « ressemble » à ce champ. C'est-à-dire qu'elle vérifie un certain critère d'erreur aux nœuds.

**Interpolant :** Une approximation est interpolante lorsqu'elle prend exactement les mêmes valeurs que le champ aux nœuds.

**Lissage :** Au sens strict, le lissage est la construction d'une approximation ou d'une interpolation. Le choix de ce terme peut s'expliquer de différentes façons,

- le lissage, dans le sens esthétique, est la construction d'une fonction dont la courbe est agréable à l'œil (splines plaques minces).
- le lissage, dans le sens mathématique, est la construction d'une fonction de régularité donnée (même  $C^0$  par exemple).

L'importante question qui subsiste est: « pourquoi rechercher une fonction lisse? ». Des éléments de réponse seront proposés au long de mon exposé.

Le terme lissage prend donc une place centrale. Cependant, j'ai choisi de limiter son emploi au minimum pour choisir des termes plus précis. De la même façon, il me semble impossible de donner une liste exhaustive des utilisations d'approximation ou d'interpolation, la figure 1.1 montre quelques exemples. Mais avant d'aller plus loin, rappelons le postulat suivant [Mal74] :

« il n'existe aucun procédé générateur d'information »  
« en dehors des procédés de mesures. »

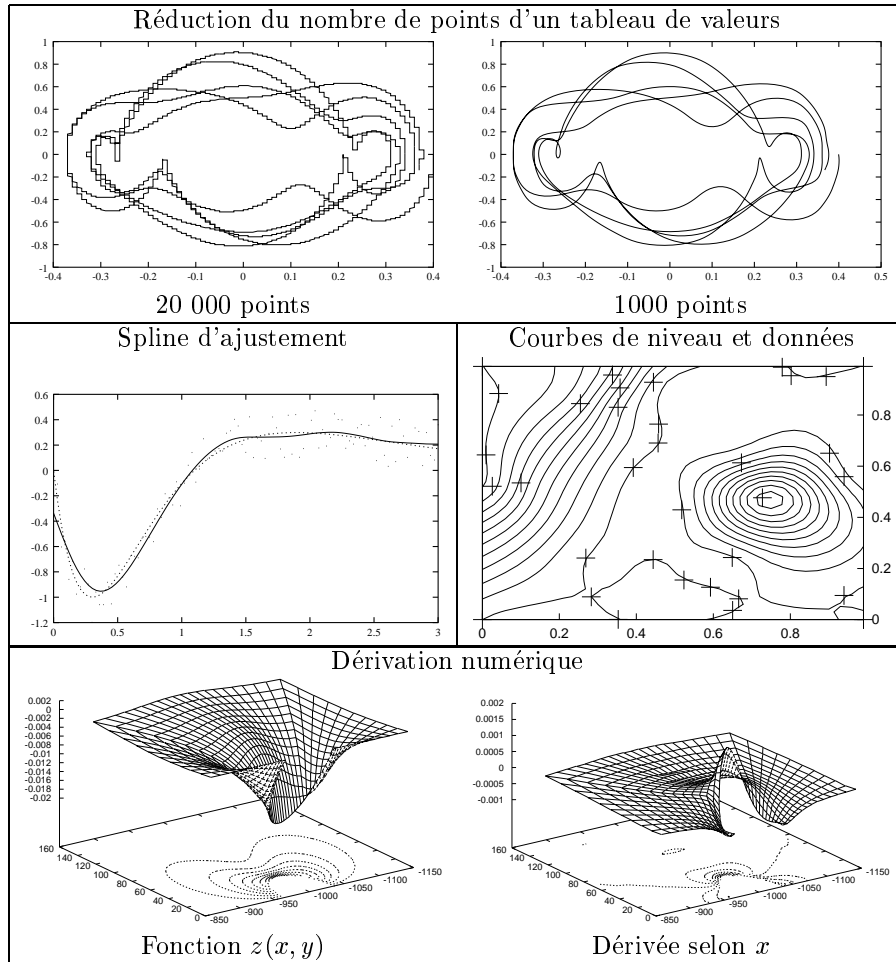


FIG. 1.1: Exemple d'utilisation d'approximation ou d'interpolation

Faisons un rapide tour d'horizon des méthodes d'approximation usuelles, afin de fixer notre cadre de travail. La première étape est de séparer deux groupes de méthodes aux objectifs différents :

1. La construction d'ajustement.
2. La construction d'interpolation.

### Ajustements

Lorsqu'on dispose de données abondantes, mais entachées d'erreur, on construit une courbe moyenne dans un certain sens : c'est l'ajustement. La méthode

des moindres carrés est certainement la plus célèbre des méthodes d'ajustement. Le principe de base en a été proposé par Gauss. Choisissons a priori une forme pour la fonction ajustée

$$f(x) = \sum_{l=1}^m a_l f^l(x)$$

où les fonctions de base  $f^l$  peuvent être issues d'une analyse de la physique du phénomène ou résulter d'un choix arbitraire. Les coefficients  $a_l$  sont calculés de façon à minimiser l'erreur d'estimation (**une** mesure de l'erreur d'estimation) :

$$E_d = \sum_{i=1}^n w_i (f(x_i) - u_i)^2$$

où les poids  $w_i$  sont des constantes choisies a priori. Le vecteur des  $a_l$  qui rend stationnaire  $E_d$  est la solution, on le calcule par le système :

$$[F^T W F] \{a\} = [W F] U$$

où  $W$  est la matrice diagonale des  $w_i$  et  $F_{il} = f^l(x_i)$ . Dans la méthode générale les paramètres inconnus  $a_l$  peuvent intervenir de manière arbitraire dans la fonction  $f$ . C'est donc délibérément que j'ai restreint cette présentation au cas de l'ajustement linéaire.

On peut aussi développer une théorie continue<sup>1</sup> en remplaçant la sommation par une intégration dans le critère d'erreur

$$E_c = \int_a^b w(t) [f(t) - u(t)]^2 dt$$

Sur la base des moindres carrés peuvent se construire plusieurs variantes. La première est l'utilisation de base de fonctions particulières qui donnent à la matrice  $A = F^T W F$  une forme simple (diagonale par exemple). Les polynômes de Legendre  $L_m$  définis par

$$\begin{cases} \gamma_m(x) = \frac{(x^2-1)^m}{2^m m!} \\ L_m(x) = \frac{d^m}{dx^m} \gamma_m(x) \end{cases}$$

comme fonctions de base sur le segment  $] - 1, 1[$  nous permettent de construire un système diagonal car ils vérifient la propriété

$$\int_{-1}^1 L_n(x) L_m(x) dx = \begin{cases} 0 & \text{si } m \neq n \\ \frac{2}{2n+1} & \text{si } m = n \end{cases}$$

Ce sont des polynômes orthogonaux (par construction).

Une seconde variante est l'algorithme d'ajustement *Locally Weighted Scatter plot Smoothing* (LOWESS). Les constantes  $w_i$  du critère d'erreur discret sont remplacés par des fonctions  $w_i(x)$  et on évalue un nouveau jeu de coefficients  $a_l(x)$  en chaque point  $x$ . Certains arguments statistiques permettent un calcul itératif (convergent) des fonctions  $w_i(\cdot)$  pour aboutir à un « meilleur ajustement ». C'est exactement cette idée que nous utiliserons pour la construction de l'approximation diffuse dans le chapitre 2, mais les fonctions de pondérations  $w_i$  seront fixées à l'avance pour assurer certaines propriétés de l'interpolant.

Nous rencontrerons aussi une seconde méthode d'ajustement dans le chapitre 3 basée sur les splines (ou le krigeage).

1. Dont l'intérêt est de fournir un cadre sous-jacent au cas discret.

### Interpolation (cas linéaire)

La construction d'interpolation poursuit un but différent. Pour commencer, il s'applique à des types de données différentes. L'ajustement permettra de tirer une loi de données trop nombreuses, et bruitées, alors que l'approximation permettra de construire des valeurs là où le champ n'est pas connu. Dans ce cadre, on construit aussi des interpolations, ce qu'on évite dans le cadre d'ajustement. Enfin la construction d'approximation fait apparaître un espace de travail particulier qui est sous-jacent à de nombreuses méthodes d'approximation.

Le problème de l'approximation est de construire une fonction sur un domaine ouvert  $\Omega$ , qui s'approche d'une fonction connue sur un ensemble fini de points  $(x_i)_{i \in I} \subset \Omega$ , où elle prend les valeurs  $(u_i)_{i \in I}$ . Cette recherche s'effectue dans  $\mathcal{F}(\Omega)$  l'ensemble des fonctions sur  $\Omega$ .

Pour simplifier notre analyse, fixons les positions des points de données, et penchons nous sur le problème de l'interpolation. Dans  $\mathcal{F}$ , les fonctions susceptibles de répondre à notre question forment l'ensemble

$$F_U = \{f \in \mathcal{F}, \forall i \in I, f(x_i) = u_i\}$$

Ces ensembles sont évidemment non vides pour tout vecteur de valeurs  $U$ , et sont les classes d'équivalence associées à la relation

$$f \mathcal{R} g \Leftrightarrow \forall i \in I, f(x_i) = g(x_i)$$

L'ensemble naturel de travail dans le cadre de l'interpolation est donc  $\mathcal{F}/\mathcal{R}$ , ensemble de ces classes d'équivalence. Le lissage est ici le choix d'un représentant de la classe  $F_U$ , c'est-à-dire d'une fonction particulière interpolant les points de données. Ceci implique aussi que la fonction  $u$  dont sont issues les données est dans  $F_U$ .

Pour caractériser une méthode d'interpolation, nous allons utiliser la structure de  $\mathcal{F}/\mathcal{R}$ .  $\mathcal{F}/\mathcal{R}$  est un espace vectoriel isomorphe à l'ensemble  $\mathbb{R}^n$  (où  $n$  est le cardinal de  $I$ ) des vecteurs  $U$  par la bijection (que l'on suppose linéaire)

$$U \longrightarrow F_U$$

Ainsi la base canonique de  $\mathbb{R}^n$  donne comme image par cette bijection une base de  $\mathcal{F}/\mathcal{R}$ . Chaque méthode d'interpolation se caractérise par le représentant de ces vecteurs, ce sont les **fonctions de forme**  $N_i$  de l'interpolation. Ces fonctions particulières sont la réponse à un dirac en  $x_i$  de l'interpolation, et chaque interpolant  $\tilde{u}$  d'une fonction  $u$  prenant les valeurs  $(u_i)_{i \in I}$  prend la forme

$$\tilde{u}(x) = \sum_{i \in I} N_i(x) u_i = \langle N \rangle \{U\}$$

Notons que cette analyse est valable lorsque la construction des fonctions de forme ne fait pas intervenir les données. Dans le cas du krigeage par exemple, ces fonctions de forme sont une fonction de la covariance généralisée et des positions des points de données. Ainsi, l'ajustement de la fonction de covariance sur les données donnera des fonctions de forme différentes pour chaque covariance. Ceci n'est pas incohérent avec le développement précédent si l'on considère que chaque ajustement de covariance génère un nouvel interpolateur.

### Interpolation polynomiale

Le cas de l'interpolation polynomiale en 1d illustre bien notre propos. Pour chaque  $U$  fixé, il existe un unique polynôme de degré  $n - 1$  dans  $F_U$ . Ce choix

permet de construire une interpolation. La technique de construction est simple, ce polynôme  $P_U$  a la forme

$$P_U(x) = \sum_{i=0}^{n-1} a_i x^i$$

et les conditions d'interpolation forment le système

$$[A] \{a\} = U$$

où

$$A = \begin{bmatrix} 1 & x_1 & \dots & x_1^{n-1} \\ 1 & x_2 & \dots & x_2^{n-1} \\ \vdots & & & \vdots \\ 1 & x_n & \dots & x_n^{n-1} \end{bmatrix}$$

est la matrice de Van der Monde associée aux points  $x_1, \dots, x_n$  inversible dès que ces points sont distincts. On peut donc calculer  $P_U$  pour tout  $U$ .

La construction des fonctions de forme associées à l'interpolation polynomiale nous amène naturellement aux polynômes de Lagrange

$$L_i(x) = \frac{\prod_{j \neq i} (x - x_j)}{\prod_{j \neq i} (x_i - x_j)}$$

qui prennent la valeur 1 en  $x_i$  et 0 pour tous les autres points de données. Le polynôme  $P_U$  s'écrit immédiatement sous la forme

$$P_U(x) = \sum_{i \in I} L_i(x) u_i$$

On retrouve ici l'idée des fonctions orthogonales, où l'on résout une fois pour toutes le système de construction de l'interpolation.

Le problème principal de la méthode que nous venons d'exposer apparaît lorsqu'on doit traiter un grand volume de données. De nombreux calculs numériques sont nécessaires, qui peuvent devenir instables lorsque deux points de données sont proches. On peut aussi remarquer que les fonctions de forme associées présentent de très grandes oscillations. Cette interpolation est aussi une interpolation globale, et on préfère des interpolations locales pour résoudre des équations différentielles (quatrième partie).

### Interpolation par sous-domaines

L'interpolation polynomiale souffre de deux problèmes principaux : elle est globale, elle est instable dès que le nombre de points de données devient important. L'interpolation linéaire par morceaux qui est continue et définie naturellement (on relie les points de données par une droite), donne une piste. Les splines cubiques donnent aussi une interpolation de ce type : ils sont composés d'arcs de polynômes de degrés trois sur les segments reliant deux points de données successifs, et se raccordent de façon  $\mathcal{C}^2$  sur les points de données. Mais l'interpolation spline est globale. Une méthode pour rendre l'interpolation polynomiale locale est basée sur un découpage de  $\Omega$  en parties élémentaires  $\Omega_e$  sur lesquelles on calcule une interpolation respectant des conditions de raccordement à la frontière. Cette transformation de l'interpolation polynomiale introduit de façon naturelle l'interpolation par sous-domaines.



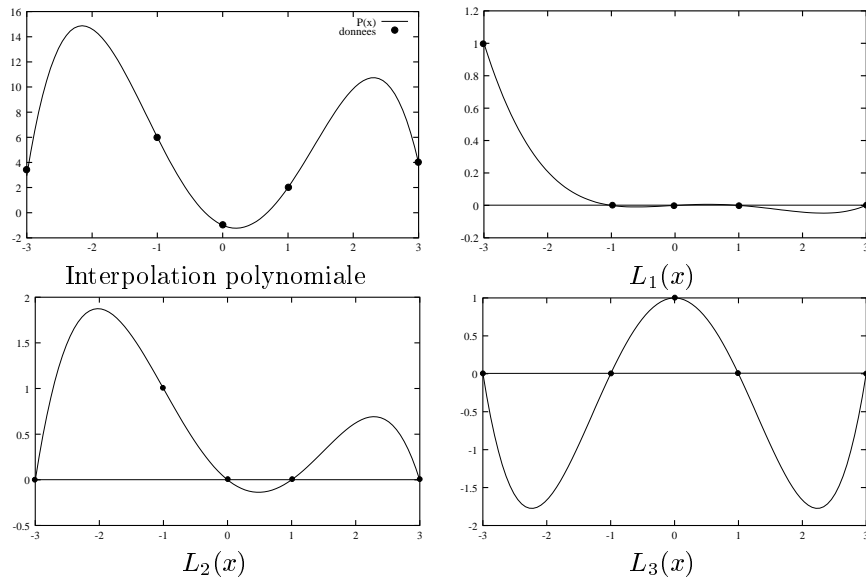


FIG. 1.2: *Interpolant polynomial pour cinq points de données. Polynômes de Lagrange associés ( $L_4$  et  $L_5$  sont symétriques de  $L_2$  et  $L_1$  respectivement).*

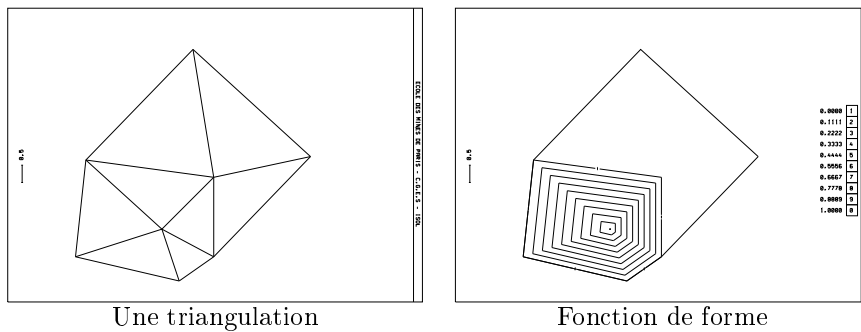


FIG. 1.3: *Une triangulation appuyée sur des points de données. Courbe de niveau d'une fonction de forme; le passage d'un élément à son voisin fait apparaître une discontinuité de la pente.*

En 2d, on peut par exemple générer une triangulation appuyée sur les points de données. Sur chaque élément  $T_{ijk}$  de sommet  $x_i, x_j, x_k$ , on construit une interpolation, par exemple linéaire (voir la figure 1.3). Cette construction assure une interpolation continue, mais dont les dérivées sont constantes par morceaux.

Il existe d'autres types d'éléments qui sont plus ou moins modulables et qui permettent de prendre en compte la géométrie de  $\Omega$  avec des techniques différentes. Le principal problème de ce type d'interpolation est la construction d'interpolant à haut degré de régularité. Par exemple, la construction d'une surface 3d  $C^2$  sur une triangulation demande l'emploi de 27 points supplémentaires (points de Bézier) et de fonctions rationnelles de degré 9 sur 4 (voir [CS97])!

Ce problème de régularité des interpolations par sous-domaine trouve une solution plus simple lorsqu'on travaille sur des domaines rectangulaires. Il est alors possible d'écrire que l'interpolant est le produit tensoriel des interpolations 1d sur les côtés du rectangle. Ce cas est cependant beaucoup trop particulier pour être satisfaisant. Un premier pas vers des interpolants plus réguliers peut être franchi avec l'interpolant de Sibson [Sib81].

L'interpolant de Sibson s'appuie sur la notion de *voisins naturels*. Deux points sont des voisins naturels si leurs cellules de Voronoï d'ordre 1 sont adjacentes. L'interpolation en  $x$  s'écrit comme une combinaison linéaire des valeurs aux voisins naturels  $x_i$  du point  $x$ , (nous noterons  $V_n(x)$  l'ensemble des voisins naturels de  $x$ ). Les pondérateurs sont les « coordonnées naturelles » du point  $x$ .

Pour définir les coordonnées naturelles, introduisons quelques notations. La cellule de Voronoï d'ordre 1 associée au nœud  $i$  est notée  $V_i$  lorsqu'on ne prend pas en compte le point  $x$ , et  $V_i^x$  lorsqu'on le prend en compte. La coordonnée naturelle de  $x$  par rapport à  $i$  est le rapport des surfaces (volumes) de  $V_x^x \cap V_i = V_i - V_i^x$  et  $V_x^x$ , nous la noterons  $\phi_i(x)$ . Nous avons donc finalement l'approximation

$$\tilde{u}(x) = \sum_{i \in V_n(x)} \phi_i(x) u_i$$

La figure 1.4 illustre cette construction.

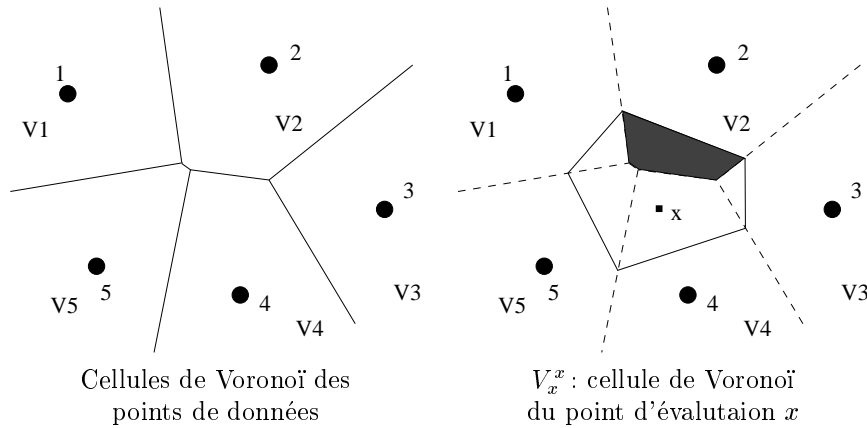


FIG. 1.4: La fonction de forme  $\phi_2$  est égale au rapport de la surface grisée  $V_x^x \cap V_2$  et de la surface de la cellule de Voronoï  $V_x^x$  associée à  $x$ .

L'approximation de Sibson vérifie les propriétés suivantes :

- $\forall i, j \in I, \phi_i(x_j) = \delta_i^j$ , l'approximation de est interpolante.
- L'interpolant est consistant d'ordre 1, c'est-à-dire

$$\forall x \in \Omega \left\{ \begin{array}{l} \sum_{i \in V_n(x)} \phi_i(x) = 1 \\ \sum_{i \in V_n(x)} \phi_i(x) x_i = x \end{array} \right.$$

- L'interpolant est local.
- L'interpolant est  $\mathcal{C}^\infty$  partout sauf aux points de données où il n'est que continu (d'après [Suk98]).

Notons qu'il existe des techniques adaptées basées sur la triangulation de Delaunay qui permettent un calcul rapide des fonctions de forme  $\phi_i$ . Ces techniques, comme celles de maillage restent complexes. Enfin, une forme  $C^1$  de l'interpolation de Sibson a été proposée dans [Far90].

Ce premier pas vers des interpolants de grande régularité est très encourageant. On retiendra surtout l'idée de base qui permet de construire l'interpolant de Sibson : on s'appuie sur une relation de voisinage entre les nœuds. Cependant les fonctions de

forme de Sibson ont des supports très complexes : c'est la réunion des disques définis par le nœud et deux de ses voisins naturels (c'est-à-dire la réunion des disques circonscrits aux triangles de Delaunay dont le nœud est un sommet) comme le montre la figure 1.5. Or nous verrons dans le chapitre 1 de la quatrième partie que l'on cherche aussi des supports de fonctions de forme très simples (pour l'intégration). L'étape suivante est donc la construction d'interpolation qui conserve les points forts de l'interpolant de Sibson (relation de voisinage, régularité) et qui permet de choisir les supports des fonctions de forme.

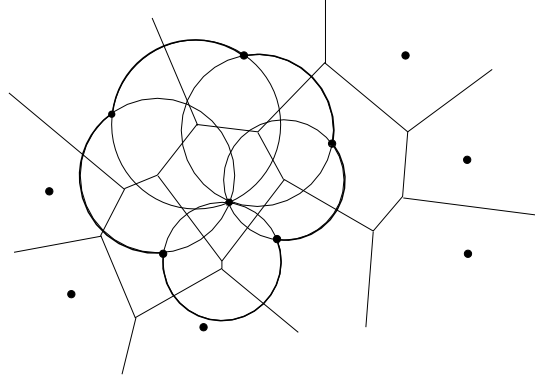


FIG. 1.5: *Diagramme de voronoï et support de la fonction de forme associée au nœud central. Le support de la fonction de forme est l'intérieur de la frontière tracée en gras.*

### Interpolation de type Hermite

Nous n'avons abordé jusqu'ici que le problème d'approximation de valeurs d'une fonction  $u$  échantillonnée aux nœuds  $x_i$ . Il arrive parfois (en géologie par exemple) que l'on dispose aussi de données différentielles (les pendages). En 1d, si les valeurs  $u_i$  et les dérivées  $v_i$  sont connues en tous les nœuds, l'interpolation de Hermite permet de construire l'interpolation  $Q_U$ .

Comme dans le cas de l'interpolation Lagrangienne, on préfère travailler par sous-domaines pour limiter le degré des polynômes. Ainsi, sur chaque segment  $[x_i, x_{i+1}]$  de demi-longueur  $l$  ramené sur  $[-1, 1]$  linéairement, on a

$$\begin{aligned} Q_U(\xi) &= u_i N_1(\xi) + \frac{v_i}{l} M_1(\xi) + u_{i+1} N_2(\xi) + \frac{v_{i+1}}{l} M_2(\xi) \\ Q'_U(\xi) &= l u_i N'_1(\xi) + v_i M'_1(\xi) + l u_{i+1} N'_2(\xi) + v_{i+1} M'_2(\xi) \end{aligned}$$

et les fonctions de forme de l'interpolation Hermite 1d sont définies par :

$$\begin{aligned} N_1(\xi) &= \frac{1}{4}(\xi^3 - 3\xi + 2) \\ M_1(\xi) &= \frac{1}{4}(\xi^3 - \xi^2 - \xi + 1) \\ N_2(\xi) &= \frac{1}{4}(-\xi^3 + 3\xi + 2) \\ M_2(\xi) &= \frac{1}{4}(\xi^3 + \xi^2 - \xi - 1) \end{aligned}$$

Ce type d'interpolation est utile pour traiter des problèmes faisant intervenir des dérivées élevées (calcul de poutre sous les hypothèses de Kirchhoff par exemple).

### L'interpolateur « idéal »

Nous sommes maintenant en mesure de décrire une technique d'approximation « idéale » qui serait :

- facile d'emploi (peu de calcul),

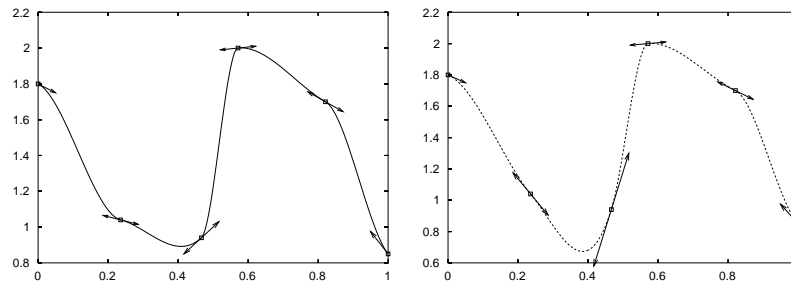


FIG. 1.6: *Interpolation Hermite: même valeurs et deux dérivées différentes aux nœuds 2 et 3.*

- adaptable (calcul d’interpolation, d’approximation et d’ajustement),
- flexible (prise en compte de données différentielles).
- locale et dont les supports des fonctions de forme sont simples,
- aussi régulière que l’on veut, et avec des irrégularités localisées (ligne de discontinuité),
- indépendante de la dimension de l’espace géométrique,

Nous allons montrer que l’approximation diffuse se rapproche d’une telle méthode d’approximation idéale.



## Chapitre 2

# Approximation Diffuse

La méthode d'approximation que nous allons décrire dans ce chapitre semble avoir été revisitée plusieurs fois. Le premier article est celui de Lancaster et Salkauskas [LS80], puis Villon dans [Vil91] propose le formalisme continu et décline le formalisme discret proposé par [LS80].

Après avoir fixé le vocabulaire et quelques notations, nous décrivons le lisseur continu en suivant la présentation de Villon puis le lisseur discret en suivant le même cheminement que précédemment et en nous appuyant sur les travaux de Guyon [Guy95a] et [Guy95b].

Nous rappelons le théorème de convergence dans les deux cas. Ceci permet de montrer que la correspondance entre la fonction approximée  $u$  et l'approximation  $\tilde{u}$  est continue. Comme l'approximation diffuse est consistante, on peut alors appliquer les résultats généraux d'approximation dans les espaces de Sobolev (voir par exemple [RT98, DO96]).

### 2.1 Définitions et notations

Cette première partie a pour objectif de fixer le vocabulaire et les notations. Nous introduirons aussi la notion de fonction d'atténuation, qui est centrale, ainsi que la forme générale des opérateurs de lissage qui seront étudiés.

Dans la suite on travaillera dans  $\mathbb{R}$ , mais la plupart des résultats sont facilement transposables à  $\mathbb{R}^d$ . Il en sera fait mention explicitement dans le cas contraire. L'espace des polynômes de degré inférieur ou égal à  $m$  sur  $\mathbb{R}$  est  $\mathcal{P}_m$ .

L'ensemble  $C^r(\mathbb{R})$  est l'ensemble des fonctions continues d'ordre  $r$  sur  $\mathbb{R}$ , et  $\mathcal{L}^q(\mathbb{R})$  l'ensemble des fonctions de puissance  $q$  sommable.

#### 2.1.1 Opérateurs de Lissage

Les deux opérateurs que nous allons rencontrer dans cette présentation font partie de la classe d'applications suivante :

$$\begin{aligned} \mathcal{L}^2(\mathbb{R}) &\longrightarrow C^r(\mathbb{R}) \\ u &\longrightarrow \tilde{u} \end{aligned}$$

avec  $\forall x \in \mathbb{R}, \tilde{u}(x) = \langle p(x) \rangle \{a(x)\}$  où  $\langle p(x) \rangle$  est une base de l'espace  $\mathcal{P}_m$  et  $\{a(x)\}$  minimise un certain critère  $\mathcal{E}_x(\{a\})$ .

La différence entre les deux opérateurs tient dans le critère  $\mathcal{E}$ , qui peut avoir une forme continue ou discrète. L'opérateur continu, dont la présentation est inspirée de [Vil91] offre un cadre théorique sous-jacent à l'opérateur discret utilisé en pratique. Le lien entre ces deux opérateurs peut être vu de deux façons différentes, d'une

part, le lisseur discret est considéré comme la forme discretisée du lisseur continu [LLB96], d'autre part, le lisseur discret est la transcription discrète de la forme continue [Vil91]. C'est dans la deuxième optique, que nous nous placerons.

### 2.1.2 Fonction d'atténuation ou d'influence

La fonction d'atténuation fait partie des paramètres essentiels de l'approximation diffuse. Elle va en particulier gouverner la régularité de la fonction lissée. Son rôle est d'atténuer la contribution de chaque point (dans le calcul de l'approximation) lorsque l'on s'en éloigne.

**Définition 1** *w est une fonction d'atténuation de régularité r et d'ordre m, si et seulement si*

- (i)  $w \in C^r(\mathbb{R}) \cap \mathcal{L}^2(\mathbb{R})$
- (ii)  $\forall x \in \mathbb{R}, w(x) \geq 0$
- (iii)  $\int_{\mathbb{R}} w(x) dx = 1$
- (iv)  $\forall P \in \mathcal{P}_m, w^{1/2}P \in \mathcal{L}^2(\mathbb{R})$

La propriété (iii) n'est pas essentielle dans cette définition, car c'est une simple normalisation. La propriété (iv) en revanche est très importante, car elle nous permettra de fabriquer notre opérateur de lissage (elle est nécessaire pour l'existence de  $\mathcal{E}(\{a\})$ ).

Enfin, la propriété (ii) assure que la forme bilinéaire :

$$\langle f|w|g \rangle = \int_{\mathbb{R}} f(x)w(x)g(x)dx$$

est bien un produit scalaire sur l'espace des polynômes, et ainsi l'unicité du vecteur  $\{a\}$  minimisant  $\mathcal{E}$  (voir équation (2.2)). On peut la remplacer par une hypothèse moins forte (ii') qui donne les mêmes propriétés :

$$(ii') \forall P \in \mathcal{P}_{2m+2}, \begin{cases} \int_{\mathbb{R}} P^2(x)w(x)dx = 0 \Rightarrow P = 0 \\ \int_{\mathbb{R}} P^2(x)w(x)dx \geq 0 \end{cases} .$$

On introduit aussi un paramètre de réglage  $\rho$  qui aura une grande importance pour la convergence de l'approximation.

**Définition 2** *Soit  $\rho$  le rayon d'influence et la fonction  $w_\rho$  définie par*

$$w_\rho : x \longrightarrow \frac{1}{\rho} w\left(\frac{x}{\rho}\right)$$

Comme son nom l'indique,  $\rho$  mesure la distance caractéristique de l'influence du point  $x$ .  $\rho$  peut être une fonction de  $x$  si l'on veut scruter plus particulièrement une zone limitée.  $\rho$  est l'analogue de la dimension caractéristique des mailles dans l'interpolation par sous-domaines.

**Définition 3** *Une fenêtre de référence  $w_{ref}$  est une fonction vérifiant les propriétés (i), (ii) et (iv) de la définition 1.*

En pratique, les fonctions d'atténuation sont construites par normalisation (propriété (iii)) de fenêtre de référence. La plus naturelle est la fonction de Gauss  $w_{ref}(x) = e^{-3x^2}$ , mais on lui préfère des fonctions de support borné, comme :

$$- \text{exponentielle tronquée } w_{ref}(x) = \begin{cases} e^{-x^2} - \frac{1}{e}, & |x| < 1 \\ 0 & \text{sinon} \end{cases}$$

$$\begin{aligned}
- \text{ spline cubique } w_{ref}(x) &= \begin{cases} \frac{2}{3} - 4x^2 + 4x^3, & |x| < \frac{1}{2} \\ \frac{4}{3} - 4x + 4x^2 - \frac{4}{3}x^3, & \frac{1}{2} < |x| < 1 \\ 0 & \text{sinon} \end{cases} \\
- \text{ spline quartique } w_{ref}(x) &= \begin{cases} 1 - 6x^2 + 8x^3 - 3x^4, & |x| < 1 \\ 0 & \text{sinon} \end{cases} \\
- \text{ fenêtre de Schwartz } w_{ref}(x) &= \begin{cases} \exp\left(\frac{x^2}{x^2-1}\right), & |x| < 1 \\ 0 & \text{sinon} \end{cases}
\end{aligned}$$

Le choix de  $w_{ref}$  n'a été abordé que dans le cadre RKPM et il est basé sur une analyse fréquentielle (transformée de Fourier, voir chapitre 1, partie IV).

## 2.2 Lissage par moindres carrés – Cas continu

Cette partie propose la description du lisseur continu en cinq étapes

1. Définition du critère d'erreur  $\mathcal{E}$ ,
2. Construction d'une forme équivalente,
3. Enoncé de propriétés,
4. Définition de la pseudo-dérivation,
5. Enoncé d'un théorème de convergence.

### 2.2.1 Opérateur $\Pi$

Le critère que nous allons utiliser pour l'opérateur  $\Pi$  est le suivant :

$$\begin{aligned}
\mathcal{E}_x^\rho(\{a\}) &= \frac{1}{2} \int_{\mathbb{R}} w_\rho(x-y) (u(y) - \langle p(y) \rangle \{a\})^2 dy \\
&= \frac{1}{2} \langle a \rangle A(\rho, x) \{a\} - \langle b(\rho, x) \rangle \{a\} + R(\rho, x)
\end{aligned}$$

avec

$$\begin{cases} A(\rho, x) &= \int_{\mathbb{R}} w_\rho(x-y) \{p(y)\} \langle p(y) \rangle dy & \text{matrice } m+1 \times m+1 \\ b(\rho, x) &= \int_{\mathbb{R}} w_\rho(x-y) \{p(y)\} u(y) dy & \text{vecteur } m+1 \text{ lignes} \\ R(\rho, x) &= \frac{1}{2} \int_{\mathbb{R}} w_\rho(x-y) u^2(y) dy \end{cases} \quad (2.1)$$

Il mesure une certaine norme de l'erreur commise en utilisant  $\tilde{u}$  comme approximation de  $u$ . L'idée de base d'un tel critère est de choisir les points qui seront considérés comme importants pour le calcul de l'erreur au point  $x$  via la fonction  $w_\rho$ .

On peut montrer que ce critère définit un unique vecteur  $\mathbf{a}^*$  à chaque  $x$  fixé sous les hypothèses de définition de la fonction d'atténuation (voir [Vil91]). La propriété (iv) assure l'existence de  $\mathcal{E}$ , et (ii) l'existence et l'unicité d'un minimum.

**Propriété 1**  $\mathbf{a}^*$  est solution du système d'optimalité

$$A(\rho, x) \{a\} = b(\rho, x). \quad (2.2)$$

Ceci définit l'approximation diffuse dans le cas continu.

**Définition 4**  $L'$  approximation diffuse (cas continu) de  $u$  est définie par

$$\tilde{u}(x) = \langle p(x) \rangle \{\mathbf{a}^*(x)\} \quad (2.3)$$



**Propriété 2**

$$\forall u \in \mathcal{L}^2(\mathbb{R}), \tilde{u} \in C^r(\mathbb{R}) \quad (2.4)$$

C'est en effet la régularité de la fonction  $w$  (et non celle de la fonction approximée) qui donne celle de la fonction approximante.  $\Pi$  est donc bien un lisseur.

La fonction approximante que nous cherchons,  $\tilde{u}(x) = \langle p(x) \rangle \{a(x)\}$  est un développement de  $u$  selon la base de polynômes que nous nous sommes fixée. En 0, ce développement nous rappelle celui de Taylor–Lagrange. C'est pourquoi nous allons nous pencher sur l'expression centrée de  $\Pi$ .

**2.2.2 Expression centrée**

On peut donner une formulation équivalente à la définition de  $\Pi$ , en utilisant la nouvelle fonction de coût

$$\mathcal{E}_x^{c\rho}(\{\alpha\}) = \int_{\mathbb{R}} w_\rho(x-y)(u(y) - \langle p(x-y) \rangle \{\alpha\})^2 dy$$

[Vil91] montre que, si  $\{\alpha^*\} = \operatorname{argmin}(\mathcal{E}_x^{c\rho})$  et  $\{a^*\} = \operatorname{argmin}(\mathcal{E}_x^\rho)$  alors

$$\begin{aligned} \mathcal{E}_x^{c\rho}(\{\alpha^*\}) &= \mathcal{E}_x^\rho(\{a^*\}) \quad \text{et} \\ \tilde{u}(x) &= \alpha_0^*(x) \end{aligned} \quad (2.5)$$

avec  $\alpha_0^* = \langle 1, 0, \dots, 0 \rangle \{\alpha\}$  et ce lorsque  $\mathcal{P}_m$  est muni de la base canonique  $\langle 1, \dots, \frac{x^m}{m} \rangle$ . Si  $u \in \mathcal{L}^2(\mathbb{R})$ , la minimisation de  $\mathcal{E}_x^{c\rho}$  amène au système

$$A(\rho)\{\alpha\} = \{b(\rho, x)\} \text{ où}$$

$$\begin{aligned} A(\rho) &= \int_{\mathbb{R}} w_\rho(y) \{p(y)\} \langle p(y) \rangle dy \\ b(\rho, x) &= \int_{\mathbb{R}} w_\rho(y-x) \{p(y-x)\} u(y) dy \end{aligned}$$

qui nous donne  $\{\alpha\}$ ,

$$\begin{aligned} \{\alpha(x)\} &= \int_{\mathbb{R}} w_\rho(y-x) A^{-1}(\rho) \{p(y-x)\} u(y) dy \\ &= \int_{\mathbb{R}} \mathcal{K}(\rho, x-y) u(y) dy \end{aligned} \quad (2.6)$$

et

$$\tilde{u}(x) = \int_{\mathbb{R}} K(\rho, x-y) u(y) dy \quad (2.7)$$

avec

$$\mathcal{K}(\rho, x) = w_\rho(x) A^{-1}(\rho) \{p(x)\} \quad (2.8)$$

$$K(\rho, x) = \langle 1, 0, \dots, 0 \rangle \mathcal{K}(\rho, x) \quad (2.9)$$

Ceci prouve que  $\Pi$  est un opérateur de convolution (et donc linéaire), et ouvre une porte vers les méthodes spectrales. En particulier, les paramètres  $x$  et  $\rho$  nous rappellent les translations et les dilatations des ondelettes. Nous retiendrons la propriété suivante.

**Propriété 3** *Les critères  $\mathcal{E}_x^\rho$  et  $\mathcal{E}_x^{c\rho}$  construisent le même opérateur.*

### 2.2.3 Propriétés

**Propriété 4** Les éléments de  $\mathcal{P}_m$  sont invariants par  $\Pi$ .

si  $P \in \mathcal{P}_m$  alors  $\forall x \in \mathbb{R}, P(x) = \langle p(x) \rangle \{a_P\}$  et il suffit de prendre  $\{a\} = \{a_P\}$  qui annule  $\mathcal{E}_x^p$  pour tout  $x$  dans  $\mathbb{R}$ .

**Définition 5** On dit que l'approximation diffuse continue est **consistante** d'ordre  $m$ .

L'invariance des polynômes par  $\Pi$  nous invite à nous pencher de plus près sur les autres fonctions reproduites par  $K$ , c'est-à-dire l'ensemble

$$\mathcal{H}_K = \{f \in \mathcal{L}^2(\mathbb{R}), f = \int_{\mathbb{R}} K(\rho, x - y)f(x)dx\}$$

Nous pouvons montrer que  $\mathcal{H}_K$  admet les propriétés suivantes

- $\mathcal{H}_K \subset C^r(\mathbb{R})$ , car il est composé d'images de  $\Pi$ ,
- $\mathcal{P}_m \subset \mathcal{H}_K$ , car les polynômes sont stables par  $\Pi$ .

En revanche, le noyau  $K$  n'appartient pas toujours à  $\mathcal{H}_K$ . Prenons l'exemple de la fonction d'atténuation

$$w(x) = \frac{e^{-x^2}}{\sqrt{\pi}}$$

et les paramètres  $\langle p(x) \rangle = \bar{1}$ ,  $\rho = 1$ . On montre facilement que  $K = w$  et  $\Pi w = v$  avec  $v(x) = e^{-x^2}/\sqrt{2}$ . Donc  $\Pi K \neq K$ . Ceci implique que l'on n'a pas toujours  $K \in \mathcal{H}_K$ .

**Propriété 5** L'opérateur  $\Pi$  est continu sur l'espace de fonctions

$$\mathcal{U} = \{u \in C^{m+1}(\mathbb{R}), u^{(k)} \in \mathcal{L}^2(\mathbb{R}), \forall k \in \llbracket 0, m+1 \rrbracket\}$$

Voir [Vil91].

### 2.2.4 Pseudo-dérivation

Si  $u \in C^{m+1}(\mathbb{R})$ , on peut écrire son développement de Taylor-Young au voisinage du point  $x$ ,

$$u(y) = \sum_{i=0}^m \frac{(y-x)^i}{i!} u^{(i)}(x) + \mathcal{O}((x-y)^{(m+1)})$$

Or la minimisation de  $\mathcal{E}_x$  revient à chercher la courbe polynomiale la plus proche de  $y = u(x)$ , c'est-à-dire qu'au voisinage de  $x$  on écrit

$$\tilde{u}(y) = \sum_{i=0}^m \frac{(y-x)^i}{i!} \alpha_i^*(x)$$

Ces deux formules nous invitent à identifier terme à terme pour définir la pseudo-dérivation.

**Définition 6** La pseudo-dérivée d'ordre  $i$  de  $\tilde{u}$  est définie par

$$\frac{\delta^i \tilde{u}}{\delta x^i}(x) = \alpha_i^*(x) \quad (2.10)$$

Cette définition appelle deux remarques,

1. La pseudo-dérivation ne vérifie pas la propriété  $D(uv) = uD(v) + D(u)v$ , ce qui justifie le terme « pseudo » (voir figure 2.3 dans le cas discret).
2. On peut se demander si les pseudo-dérivées représentent bien les dérivées de  $u$ ; ce qui nous amène à la convergence du lisseur  $\Pi$ .

### 2.2.5 Convergence

Je me contenterai ici d'enoncer le théorème de convergence de l'opérateur  $\Pi$  et de donner un rapide commentaire.

#### Théorème 1

$$\forall u \in \mathcal{U}, \forall k \in \llbracket 0, m \rrbracket, \exists M_k > 0, \forall x \in \mathbb{R}, \quad \left| \left( \frac{\delta^k \tilde{u}}{\delta x^k} - \frac{d^k u}{dx^k} \right) (x) \right| \leq \frac{\rho^{m+1-k}}{(m+1)!} M_k \quad (2.11)$$

La norme de l'erreur fait bien intervenir les deux aspects principaux du lisseur que nous venons de présenter par le terme  $\rho^{m+1-k}$ .  $\rho$  représente la portée des fonctions d'atténuation (comportement de type  $h$ ) et  $m + 1$  l'ordre de la base polynômiale (comportement de type  $p$ ).

Si  $\rho$  n'est pas constant, alors sous l'hypothèse supplémentaire  $\rho(x) \in C^0(\mathbb{R}) \cap \mathcal{L}^\infty(\mathbb{R})$  et  $\rho(x) > 0$ , le théorème précédent reste valide, à condition de remplacer  $\rho$  par  $\rho(x)$ .

### 2.2.6 Reproducing Kernel Particle

Il existe d'autre forme de lissage par convolution. L'une d'entre elle est la méthode dite *reproducing kernel particle* qui aboutit au même genre de noyau que l'approximation diffuse. La construction est cependant différente et basée sur la consistance de l'approximation [LJZ95] (on utilise ici la formulation proposée dans [LHCG97]). On cherche à construire une approximation de type convolution

$$\tilde{u}(x) = \int_{\mathbb{R}} u(y) \bar{\phi}_\rho(x, x-y) dy$$

sous conditions de consistance de degré  $m$ , ce qui peut s'écrire

$$\forall k \in \llbracket 0, m \rrbracket, \int_{\mathbb{R}} y^k \bar{\phi}_\rho(x, x-y) dy = x^k$$

et de façon équivalente en formulation centrée ( $\delta_i^j$  symbol de kronecker).

$$\forall k \in \llbracket 0, m \rrbracket, \int_{\mathbb{R}} (y-x)^k \bar{\phi}_\rho(x, x-y) dy = \delta_0^k$$

On pose a priori la forme du noyau comme le produit d'une fonction d'atténuation et d'un polynôme à coefficients variables, soit

$$\bar{\phi}_\rho(x, x-y) = \left( \sum_{k=0}^m b_k^\rho(x) \left( \frac{x-y}{\rho} \right)^k \right) w_\rho(x-y)$$

et les coefficients  $b_k^\rho(x)$  sont calculés de façon à satisfaire les conditions de consistance. Ceci conduit à un système linéaire  $A(x)\{b^\rho(x)\} = e_1$  où  $A(x)$  a le même sens que précédemment. On aboutit finalement à des propriétés semblables à celle de l'approximation diffuse avec une fonction de correction différente<sup>1</sup>. L'expression *reproducing kernel* est utilisée car on a corrigé le noyau pour reproduire exactement les polynômes. Ceci ne doit pas être confondu avec les espaces de Hilbert à noyau reproduisant que nous verrons plus loin.

1. Le lien RKPM-approximation diffuse est éclairci page 144 et 224

## 2.3 Lissage par moindres carrés – Cas discret

L'opérateur que nous venons de présenter n'est pas adapté à des données nodales. Villon, dans [Vil91], montre qu'il est possible de le construire dans l'espace  $\mathcal{D}'$  des distributions, mais préfère s'intéresser au cas discret. La partie précédente garde pourtant son intérêt, pour deux raisons :

1. Elle donne le cadre continu sous-jacent au cas discret.
2. De nombreuses similitudes rapprochent les deux opérateurs.

Trois présentations du lisseur discret existent dans la littérature, les pionniers sont Lancaster et Salkauskas [LS80], puis Villon dans [Vil91], et enfin l'équipe Belytschko qui présente pour la première fois le lisseur discret dans [TBB94]. Ces différentes présentations semblent être indépendantes; l'objectif poursuivi et la présentation sont différents :

- Lancaster et Salkauskas utilisent le lisseur pour générer des surfaces.
- Villon développe le lisseur continu, puis utilise le lisseur discret dans le formalisme de Galerkin pour résoudre des équations aux dérivées partielles.
- Belytschko propose uniquement le lisseur discret dans le formalisme de Galerkin [BLG94]. Puis dans [LLB96], donne une présentation du cas continu. Ces travaux recourent largement ceux de Villon.

Cette partie reprend le plan de la précédente auquel on a dû ajouter une légère introduction et un développement concernant le caractère interpolant de l'approximation avant d'en décrire les propriétés. Le cheminement est donc le suivant <sup>2</sup>

1. *Introduction*
2. Définition de  $\mathcal{E}$  discret
3. *Forme interpolante*
4. Forme centrée
5. Propriétés
6. Pseudo-dérivation
7. Convergence

### 2.3.1 Introduction

Nous définissons de nouveau un lisseur, mais l'ensemble de départ est maintenant différent puisqu'on s'intéresse à une fonction connue en des nœuds seulement. Nous noterons  $(x_i)_{i \in I}$  un ensemble fini de points d'un ouvert  $\Omega$  de  $\mathbb{R}$ , et une fonction  $w$  de  $\Omega \times \Omega$  dans  $\mathbb{R}^+$ . En pratique, la fonction  $w$  à la forme  $w(x, y) = w_{ref}(\frac{\|x-y\|}{\rho(x)})$ , où  $x$  et  $y$  peuvent être interchangeables. La fonction  $u$  que nous cherchons à approximer prend les valeurs  $u_i$  aux points  $x_i$ . Nous retrouvons donc le cadre de l'introduction.

**Définition 7** *Les notations suivantes seront utilisées dans la suite*

1.  $I(x) = \{\text{indice des nœuds voisins de } x\} = \{i \in I, w(x, x_i) \neq 0\}$  est la **liste d'influence** du point  $x$  et chaque nœuds  $x_i, i \in I(x)$  est un **nœud d'influence** de  $x$ .

---

<sup>2</sup>. les parties ajoutées sont en italiques

2.  $V_i$  est la **zone d'influence** de  $x_i$ , où  $V_i = \text{supp}(w(\cdot, x_i))$ , et
3.  $P_K = \{x, I(x) = K\}$  l'élément diffus associé aux nœuds  $\{x_i\}_{i \in K}$ .

L'ensemble des  $P_K$  non vides forme un pavage de  $\cup_{i \in I} V_i$ , car pour  $K$  fixé,  $P_K = \cap_{i \in K} V_i$ . On a aussi la propriété

$$i \in I(x) \Leftrightarrow x \in V_i \quad (2.12)$$

Les définitions 1. et 2. utilisent pleinement la double interprétation de la fonction  $w(\cdot, \cdot)$ ; d'une part (1.) on fixe  $x$  et on considère la fonction  $w^x(x_i)$ , d'autre part (2.), on fixe  $i$  et on considère la fonction  $w_i(x)$ . Enfin, la propriété 2.12 lie ces deux interprétations de  $w(\cdot, \cdot)$ .

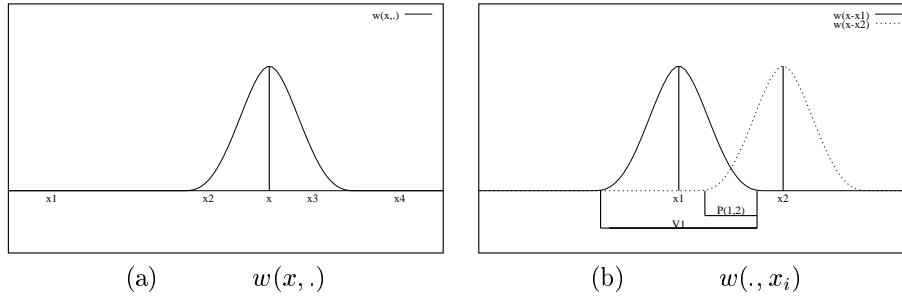


FIG. 2.1: *Dualité de la fonction  $w(\cdot, \cdot)$ : (a) On considère la fonction  $i \rightarrow w^x(x_i)$  pour définir  $I(x) = \{2, 3\}$  (b) La fonction de pondération associée à  $x_i$  définit son support  $V_i$  lorsque le point d'évaluation  $x$  bouge.*

**Définition 8** On dit qu'un ensemble de points est  $\mathcal{P}_k$ -**unisolvant** s'il admet un sous-ensemble de points sur lequel l'interpolation polynomiale (au sens des moindres carrés) d'ordre  $k$  existe et est unique.

Cette notion est utile pour assurer l'existence de l'approximation; [ST81] étudie quelques cas particuliers.

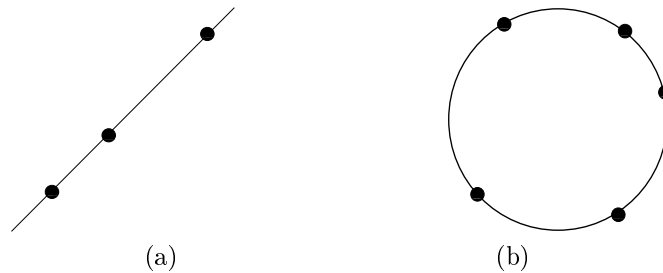


FIG. 2.2: *Cas de nuage (a) non  $\mathcal{P}_1$ -unisolvant (b) non  $\mathcal{P}_2$ -unisolvant pour  $\mathbb{R}^2$*

**Hypothèse 1** Dans la suite on considère un semis de points qui vérifie la propriété :

$$\forall x, \{x_i, i \in I(x)\} \text{ est } \mathcal{P}_m\text{-unisolvant} \quad (2.13)$$

### 2.3.2 Lisseur discret

Le critère  $\mathcal{E}$  discret peut être défini à partir des éléments introduits dans la partie précédente comme

**Définition 9**

$$\mathcal{E}_x(\{a\}) = \sum_{i \in I(x)} w(x, x_i) (u_i - \langle p(x_i) \rangle \{a\})^2$$

Notons  $U$  le vecteur des valeurs nodales de  $u$ ,

$$P = \begin{bmatrix} p_1(x_1) & \dots & p_m(x_1) \\ \vdots & \ddots & \vdots \\ p_1(x_n) & \dots & p_m(x_n) \end{bmatrix}$$

et

$$W(x) = \begin{bmatrix} w(x, x_1) & & 0 \\ & \ddots & \\ 0 & & w(x, x_n) \end{bmatrix}$$

Le vecteur  $\{a\}$  est solution de

$$\begin{aligned} A(x)\{a\} &= \{b(x)\} \text{ avec} \\ A(x) &= P^\top W(x)P \\ b(x) &= P^\top W(x)U \end{aligned} \tag{2.14}$$

**Définition 10**  $L$ 'approximation diffuse  $\tilde{u}$  de  $u$  s'écrit

$$\tilde{u}(x) = \langle p(x) \rangle [A(x)]^{-1} P^\top W(x)U \tag{2.15}$$

**Propriété 6** Sous l'hypothèse d'unisolvance (2.13), la matrice  $A(x)$  est bien inversible, et l'approximation  $\tilde{u}$  de  $u$  existe.

L'approximation diffuse est construite par ajustements locaux par moindres carrés, elle n'interpole donc pas dans le cas général les données  $u_i$ . Cette propriété demande de modifier la fonction  $w$ .

### 2.3.3 Interpolation

Revenons au système d'optimalité au point  $x$ ,

$$\left[ \sum_{i \in I(x)} w(x, x_i) p(x_i) p^\top(x_i) \right] \{a\} = \left\{ \sum_{i \in I(x)} w(x, x_i) p(x_i) u(x_i) \right\}$$

En désignant par  $x_j$  le plus proche voisin de  $x$ , nous écrivons le système sous la forme équivalente

$$\begin{aligned} & \left[ \sum_{i \in I(x), i \neq j} w(x, x_i) p(x_i) p^\top(x_i) \right] \{a\} + [w(x, x_j) p(x_j) p^\top(x_j)] \{a\} \\ &= \left\{ \sum_{i \in I(x), i \neq j} w(x, x_i) p(x_i) u(x_i) \right\} + w(x, x_j) p(x_j) u(x_j) \end{aligned}$$

Notons  $A_j$  et  $b_j$  les matrices suivantes,

$$\begin{aligned} A_j(x) &= \left[ \sum_{i \in I(x), i \neq j} w(x, x_i) p(x_i) p^\top(x_i) \right] \\ b_j(x) &= \left[ \sum_{i \in I(x), i \neq j} w(x, x_i) p(x_i) u_i \right] \end{aligned}$$

et  $\lambda(x)$  défini par

$$\lambda(x) = w(x, x_j)(p^\top(x_j)\{a\} - u_j)$$

le système d'optimalité se transforme en

$$\begin{bmatrix} A_j(x) & p(x_j) \\ p^\top(x_j) & -\frac{1}{w(x, x_j)} \end{bmatrix} \begin{Bmatrix} a \\ \lambda \end{Bmatrix} = \begin{Bmatrix} b_j(x) \\ u_j \end{Bmatrix} \quad (2.16)$$

Si l'on modifie la fonction poids  $w(x, x_j)$  pour que  $\lim_{x \rightarrow x_j} w(x, x_j) = +\infty$ , le système en  $x_j$  s'écrit

$$\begin{bmatrix} A_j(x) & p(x_j) \\ p^\top(x_j) & 0 \end{bmatrix} \begin{Bmatrix} a \\ \lambda \end{Bmatrix} = \begin{Bmatrix} b_j(x) \\ u_j \end{Bmatrix} \quad (2.17)$$

qui est le système d'optimalité auquel on a ajouté la contrainte d'interpolation au noeud  $x_j$ .

**Propriété 7** *Si la fonction  $\bar{w}(\cdot, \cdot)$  vérifiant*

$$(i) \quad \lim_{x \rightarrow x_i} \bar{w}(x, x_i) = +\infty$$

$$(ii) \quad \text{supp}(\bar{w}(\cdot, x_i)) = \text{supp}(w(\cdot, x_i))$$

$$(iii) \quad \bar{w}(\cdot, x_j) \in C^m(\Omega - \{x_j\})$$

*est utilisée comme fonction d'atténuation, l'opérateur que nous construisons calcule une interpolation.*

Le choix  $\bar{w} = \frac{w}{1-w}$  assure les propriétés précédentes. Les propriétés de l'approximation se transposent automatiquement à l'interpolation. Dans la suite, nous utiliserons le terme *approximation* de façon générique, et le terme *interpolation* pour le cas interpolant.

On peut aussi remarquer que cette façon de rendre interpolante l'approximation diffuse par une contrainte progressive peut être généralisée à d'autres cas. C'est le cas par exemple pour le respect de frontières par l'approximation diffuse (voir [Guy95a]).

### 2.3.4 Forme centrée, propriétés

En suivant la présentation de l'opérateur continu, on montre que l'expression centrée (avec la base  $\langle 1, x, \frac{x^2}{2}, \dots \rangle$ ):

$$\begin{aligned} \tilde{u}(x) &= \langle p(x - x_i) \rangle \{ \alpha(x) \} \\ \mathcal{E}_x^c(\{ \alpha \}) &= \sum_{i \in I(x)} w(x, x_i) (u_i - \langle p(x - x_i) \rangle \{ \alpha(x) \})^2 \end{aligned}$$

est équivalente à la précédente. Cette propriété est importante de plusieurs points de vue,

- c'est sous la forme centrée que s'appréhende le mieux la convergence de l'approximation,
- une forme centrée est plus stable numériquement,

Nous noterons  $P(x)$  la matrice des polynômes centrés

$$\begin{bmatrix} p_1(x - x_1) & \dots & p_k(x - x_1) \\ \vdots & \ddots & \vdots \\ p_1(x - x_n) & \dots & p_k(x - x_n) \end{bmatrix}$$

pour écrire l'approximation diffuse

$$\tilde{u}(x) = \langle 1, 0 \dots 0 \rangle [A(x)]^{-1} P(x)^\top W(x) U = \alpha_0^*(x)$$

**Propriété 8** *L'approximation diffuse admet les propriétés suivantes :*

1. *L'approximation diffuse est consistante d'ordre  $m$ .*
2. *L'approximation diffuse (au sens de l'application  $u \rightarrow \tilde{u}$ ) est linéaire et continue.*
3. *La fonction  $\tilde{u}$  a la même régularité que la fonction  $w$ .*

Ces propriétés se démontrent de la même façon que dans le cas continu.

### 2.3.5 Pseudo-dérivation

En nous basant sur la même remarque que dans le cas continu, nous définissons la pseudo-dérivation de la façon suivante

**Définition 11**

$$\frac{\delta^k \tilde{u}}{\delta x^k}(x) = \left\langle \frac{d^k p}{dx^k}(x) \right\rangle \{a(x)\} = \alpha_k^*(x) \quad (2.18)$$

La figure 2.3 illustre l'appellation *pseudo-dérivation*.

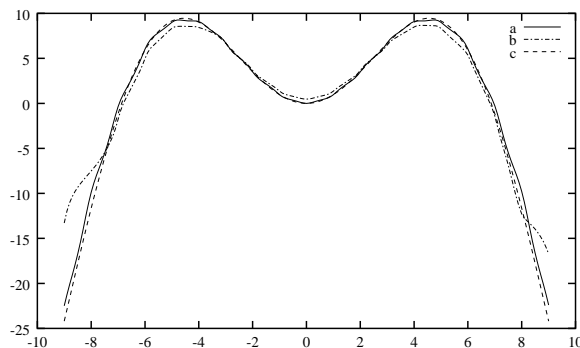


FIG. 2.3: *Calcul de la pseudo-dérivée de  $f(x) = u(x)v(x)$  où  $u(x) = x^2$  et  $v(x) = \sin(x/3)$ . (a) Calcul à partir des valeurs de  $f$  aux points d'abscisses entières, (b) On calcule les pseudo-dérivées de  $u$  et  $v$  séparément, puis on applique la formule de dérivation des produits  $f' = u'v + uv'$ , (c) dérivée réelle.*

Dans le cas de l'interpolation, on conserve les propriétés de continuité et de consistance de l'approximation diffuse. Cependant, une propriété supplémentaire, relative aux dérivées diffuses apparaît.



**Propriété 9** Si les fonctions définies par  $w_j(x) = w(x, x_j)$ , vérifient la propriété

$$\begin{aligned} l &\in \mathbb{N}, l \leq k \\ \forall i \in \llbracket 1, l \rrbracket, w_j^{(i)}(x_j) &= 0 \\ \bar{w} &= \frac{w}{1-w} \end{aligned}$$

alors on a l'égalité

$$\forall i \in \llbracket 0, l \rrbracket, \langle p(x_j) \rangle \{a^{(i)}(x_j)\} = 0$$

Commençons par démontrer le lemme de calcul suivant,

$$\begin{aligned} \text{Si } \bar{w} &= \frac{w}{1-w} \\ \text{alors } \left(\frac{-1}{\bar{w}}\right)^{(i)} &= \frac{\sum_j a_j^i \prod_{p=1}^i w^{(\lambda_{jp}^i)}}{w^{i+1}} \\ &\text{où } \sum_{p=1}^i \lambda_{jp}^i = i, \forall j \end{aligned}$$

Le résultat s'obtient par récurrence sur  $i$ . En effet, la propriété est vrai à l'ordre 0, et si elle l'est à l'ordre  $i$ , alors

$$\begin{aligned} \left(\frac{-1}{\bar{w}}\right)^{(i+1)} &= \frac{\left(\sum_j a_j^i \prod_{p=1}^i w^{(\lambda_{jp}^i)}\right)' w^{i+1} - i w^i w' \left(\sum_j a_j^i \prod_{p=1}^i w^{(\lambda_{jp}^i)}\right)}{w^{2i+2}} \\ &= \frac{w^i \left(\left(\sum_j a_j^i \prod_{p=1}^i w^{(\lambda_{jp}^i)}\right)' w - i w' \left(\sum_j a_j^i \prod_{p=1}^i w^{(\lambda_{jp}^i)}\right)\right)}{w^{2i+2}} \\ &= \frac{\left(\sum_j a_j^i \prod_{p=1}^i w^{(\lambda_{jp}^i)}\right)' w - i w' \left(\sum_j a_j^i \prod_{p=1}^i w^{(\lambda_{jp}^i)}\right)}{w^{i+2}} \end{aligned}$$

Or, les deux termes de la différence peuvent bien s'écrire sous la forme

$$\sum_j a_j^i \prod_{p=1}^{i+1} w^{(\lambda_{jp}^i)}$$

où  $\sum_{p=1}^{i+1} \lambda_{jp}^i = i + 1$ , ce qui établit la propriété<sup>3</sup>.

Les hypothèses de régularité sur  $w$  nous permettent par ailleurs de dériver le système d'optimalité  $l$  fois ( $l < m$ ), ce qui nous donne

$$\begin{aligned} \sum_{i=1}^l C_l^i \begin{bmatrix} A_j^{(i)}(x) & 0 \\ 0 & \left(\frac{-1}{\bar{w}_j(x)}\right)^{(i)} \end{bmatrix} \begin{Bmatrix} a^{(l-i)}(x) \\ \lambda^{(l-i)}(x) \end{Bmatrix} + \\ + \begin{bmatrix} A_j(x) & p(x_j) \\ p^\top(x_j) & \frac{-1}{\bar{w}_j(x)} \end{bmatrix} \begin{Bmatrix} a^{(l)}(x) \\ \lambda^{(l)}(x) \end{Bmatrix} = \begin{Bmatrix} b^{(l)}(x) \\ 0 \end{Bmatrix} \end{aligned}$$

système dont la dernière ligne s'écrit

$$\langle p(x_j) \rangle \{a^{(l)}(x)\} - \underbrace{\frac{\lambda^{(l)}(x)}{\bar{w}_j(x)} + \sum_{i=1}^l C_l^i \left(\frac{-1}{\bar{w}_j(x)}\right)^{(i)} \lambda^{(l-i)}(x)}_0 = 0$$

L'hypothèse  $\forall i \in \llbracket 1, l \rrbracket w_j^{(i)}(x_j) = 0$  et le lemme de calcul assurent que  $\left(\frac{-1}{w_j(x_j)}\right)^{(i)} = 0$ , ce qui établit la propriété; c'est-à-dire que la dérivée diffuse est exactement égale à la dérivée de  $\tilde{u}$  aux nœuds à l'ordre 1.

$$\frac{\delta \tilde{u}}{\delta x} = \frac{d \tilde{u}}{dx}$$

3. Ainsi « des dérivées nulles de  $w$  impliquent des dérivées nulles de  $\bar{w}$  »

À l'ordre 2, on montre facilement que

$$\frac{\delta^2 \tilde{u}}{\delta x^2}(x_j) = \frac{d^2 \tilde{u}}{dx^2}(x_j) + \langle p'(x_j) \rangle \{a'(x_j)\}$$

Nous allons maintenant nous pencher sur la convergence de l'approximation diffuse.

### 2.3.6 Convergence

La démonstration de la convergence de l'approximation diffuse suit quatre étapes

1. Montrer que la formulation habituelle et la formulation centrée de l'interpolation diffuse sont équivalentes (voir 2.3.4).
2. Écrire le critère quadratique à l'aide du développement de Taylor autour de  $x$  ;
3. Passer en coordonnées adimensionnelles ;
4. Enfin, majorer l'erreur commise.

Cette démonstration est constructive, elle permet de mettre en évidence un indicateur d'erreur que l'on pourra utiliser dans des processus adaptatifs. Nous utilisons la base polynomiale  $\langle 1, x, \frac{1}{2}x^2, \dots, \frac{1}{m!}x^m \rangle$ . La démonstration est généralisable dans  $\mathbb{R}^d$  au prix de l'utilisation de multi-index.

Si  $u \in C^m(\Omega)$ , et que sa dérivée d'ordre  $m + 1$  est intégrable, alors au voisinage de  $\bar{x}$

$$u(x_i) = \langle p(x_i - \bar{x}) \rangle U + \mathcal{O}((x_i - \bar{x})^{m+1})$$

avec  $U = \langle u(\bar{x}), u'(\bar{x}), \dots, u^{(m)}(\bar{x}) \rangle^\top$

le critère  $\mathcal{E}_{\bar{x}}^c(\{\alpha\})$  se réécrit alors

$$\mathcal{E}_{\bar{x}}^c(\{\alpha\}) = \frac{1}{2} \{U - \alpha\}^\top A(\bar{x}) \{U - \alpha\} + b(\bar{x}) \{U - \alpha\} + \text{reste}$$

où

$$A(\bar{x}) = P(x)^\top W(\bar{x})P(x)$$

$$b(\bar{x}) = P(x)^\top W(\bar{x})$$

avec  $P$  et  $W$  définis comme précédemment et si  $R(\bar{x})$  est le vecteur colonne des restes de Taylor, alors l'erreur  $\{U - \alpha\}$  s'écrit

$$\begin{cases} \{U - \alpha\} = [N(\bar{x})] R(\bar{x}) \\ [N(\bar{x})] = A(\bar{x})^{-1} b(\bar{x}) \end{cases} \quad (2.19)$$

Soit  $\rho$ , tel que tous les nœuds  $\{x_i\}_{i \in I(x)}$  soient dans la boule  $B(\bar{x}, \rho)$  centrée en  $\bar{x}$  et de rayon  $\rho$ . Le changement de variables  $\xi = \frac{x - \bar{x}}{\rho}$  se traduit alors par les formules suivantes

$$\langle p(x_i - x) \rangle = D \langle p(\xi_i) \rangle$$

$$D = \begin{bmatrix} 1 & & 0 \\ & \ddots & \\ 0 & & \rho^m \end{bmatrix}$$

$$A(\bar{x}) = DA(\xi)D$$

$$b(\bar{x}) = Db(\xi)$$

$$R(\bar{x}) = \rho^{m+1}R(\xi)$$

L'erreur d'approximation  $\{U - \alpha\}$  est donnée par l'expression

$$\{U - \alpha\} = \begin{bmatrix} \rho^{m+1} & & 0 \\ & \ddots & \\ 0 & & \rho \end{bmatrix} [n(\xi)] \{r(\xi)\} \quad (2.20)$$

où

$$[n(\xi)] = A^{-1}(\xi)b(\xi) = \begin{bmatrix} n_1(\xi) & \dots & n_r(\xi) \\ \vdots & & \vdots \\ \frac{\delta^m n_1}{\delta x^m}(\xi) & \dots & \frac{\delta^m n_r}{\delta x^m}(\xi) \end{bmatrix}$$

Si la fonction  $u^{(m+1)}$  est bornée, alors les coordonnées du vecteur  $\{r(\xi)\}$  le sont aussi, soit  $M^{(m+1)}$  cette borne, il vient alors

$$\forall j \leq m \quad \left| \frac{d^j u}{dx^j}(\bar{x}) - \frac{\delta^j \bar{u}}{\delta x^j}(\bar{x}) \right| \leq \rho^{m+1-j} \left( \sum_{i \in I(\bar{x})} \left| \frac{\delta^j n_i}{\delta x^j}(\xi) \right| \right) M^{(m+1)} \quad (2.21)$$

Ceci établit la convergence en  $\bar{x}$  de l'interpolation diffuse dès que le terme  $\sum_i \left| \frac{\delta^j n_i}{\delta x^j}(\xi) \right|$  reste borné lorsque  $\rho$  varie (tend vers 0). Villon dans [Vil91] montre que l'hypothèse:

### Hypothèse 2

$$\begin{aligned} \forall j \quad M_j(\bar{x}, I(\bar{x}), \rho) &= \sum_{i \in I(\bar{x})} w(\xi_i) \xi_i^j \\ \bar{M}_j(\bar{x}, I(\bar{x}), \rho) &= \sum_{i \in I(\bar{x})} w(\xi_i) |\xi_i|^{j+m+1} \\ &\text{restent bornés lorsque } (I(\bar{x}), \rho) \text{ varie} \end{aligned}$$

est suffisante. Ceci achève la démonstration.

**Théorème 2** *Sous les hypothèses 2,*

$$\forall x \in \Omega, \forall k \in \llbracket 0, m \rrbracket, \exists M_k^x, \left| \left( \frac{\delta^k \tilde{u}}{\delta x^k} - \frac{d^k u}{dx^k} \right) (x) \right| \leq \rho^{m+1-k} M_k^x$$

Notons que la convergence uniforme (que nous n'étudierons pas ici) impose l'introduction d'une hypothèse supplémentaire reliant  $I(\bar{x})$  et les fonctions  $w_i(\cdot)$ .

### 2.3.7 Fonctions de forme de l'approximation diffuse

La définition de l'approximation diffuse 2.15 montre que  $\tilde{u}$  est une fonction linéaire du vecteur  $U$ . Ceci nous permet d'écrire

$$\forall x \in \Omega, \tilde{u}(x) = \sum_{i \in I(x)} N_i(x) u_i = \langle N(x) \rangle \{U(x)\}$$

**Définition 12** *Les fonctions  $N_i$  sont les fonctions de forme de l'approximation diffuse. Elle se calcule par :*

$$\langle N(x) \rangle = \langle p(x) \rangle [A(x)]^{-1} P^\top W(x)$$

Bien entendu, la formule de calcul des fonctions de forme est aussi valable en formulation centrée. Le théorème suivant caractérise les fonctions de forme de l'approximation diffuse. Il permet une nouvelle définition de l'approximation diffuse, « *fondée sur ses propriétés essentielles : approximation locale, représentation exacte des polynômes* » [Guy95a]. Ce théorème est aussi à la base de méthodes de calcul efficaces des fonctions de forme décrites en annexe A. Ces techniques sont importantes, car les fonctions de formes sont au cœur des applications de l'approximation diffuse comme la résolution d'équations différentielles par exemple.

**Théorème 3** *Les fonctions de forme de l'approximation diffuse sont solutions du problème de minimisation sous contraintes*

$$\begin{cases} \min_N \sum_{i \in I(x)} \frac{N_i^2}{w_i} \\ P^\top N = p(x) \end{cases} \quad (2.22)$$

Introduisons les multiplicateurs de Lagrange  $\mu$ , et les fonctions de forme sont solutions du système

$$\begin{bmatrix} W^{-1}(x) & P \\ P^\top & 0 \end{bmatrix} \begin{Bmatrix} N(x) \\ \mu(x) \end{Bmatrix} = \begin{Bmatrix} 0 \\ p(x) \end{Bmatrix}$$

qui se résout en

$$\begin{cases} N(x) = W(x)P(P^\top W(x)P)^{-1}p(x) = W(x)P[A(x)]^{-1}p(x) \\ \mu(x) = (P^\top W(x)P)^{-1}p(x) \end{cases}$$

$N$  est bien le vecteur de fonctions de forme diffuses car les matrices  $W(x)$  et  $A(x)$  sont symétriques. Il existe un théorème équivalent pour les pseudo-dérivées des fonctions de forme [Guy95a].

**Propriété 10** *Le support de la fonction de forme  $N_i$  est inclu dans  $V_i$ . Les fonctions de forme sont aussi régulières que  $w$ .*

Cette propriété est liée à la construction de l'approximation diffuse. De plus, les fonctions de formes sont des approximations particulières, ce qui démontre la seconde propriété.

### 2.3.8 Fonction d'atténuation

L'approximation diffuse discrète repose sur la définition de la fonction d'atténuation  $w$ . Elle gouverne deux de ses propriétés principales : approximation locale, et continuité. Elle est aussi au cœur du traitement de certaines discontinuités des dérivées des solutions d'équations différentielles (voir [Kro96] et la partie IV). La construction de  $w$  doit donc être menée avec soin, elle est discutée en annexe B. Dans la suite, nous utiliserons principalement des fonctions d'atténuation  $w_i$  attachées à chaque nœud  $x_i$ .

## 2.4 Conclusion

En conclusion, il existe deux formes d'approximation diffuse : l'une continue, l'autre discrète. Comme pour les ajustements par moindres carrés, la formulation continue présente surtout l'intérêt d'être un arrière-plan théorique à la forme discrète. Nous utiliserons désormais le terme approximation diffuse pour la version discrète.

La construction de l'approximation diffuse repose sur la définition des fonctions d'atténuation  $w_i$  et le choix d'un degré d'approximation. Son utilisation est donc aisée. On peut calculer des approximations, des interpolations et même des ajustements en choisissant judicieusement les fonctions d'atténuation. Enfin, les fonctions de forme sont locales.

Les propriétés de l'approximation diffuse montrent que cette technique se rapproche de la méthode d'interpolation idéale que nous avons décrite dans le chapitre précédent. Cependant, la version que nous avons décrite ici n'est qu'une approximation de type Lagrangienne. Nous étudierons l'approximation diffuse de type Hermite dans la seconde partie.

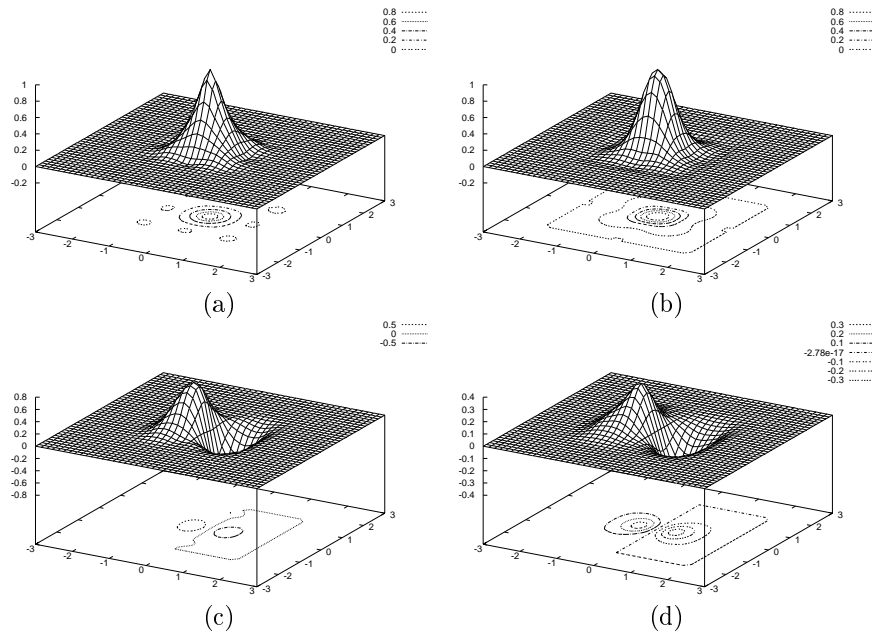


FIG. 2.4: Fonctions de forme et leurs dérivées diffuses (en  $x$ ) pour des consistances linéaire (a) (c) et quadratique (b) (d).



## Chapitre 3

# Interpolateurs Radiaux

L'approximation diffuse est une méthode d'interpolation qui ne nécessite pas de maillage, aussi avons-nous décidé d'étudier d'autres méthodes d'interpolation de ce type. Dans le monde du sous-sol, le krigeage est un interpolateur sans maillage particulièrement puissant et célèbre. Cependant, Myers écrit dans [Mye92] « *There are at least three developments for interpolators to lead to the same functional form for the interpolator; the thin plate spline, radial basis functions and the regression method known as kriging* ». Cette phrase nous invite donc à présenter ces trois méthodes dans un formalisme unifié, bien que la base bibliographique que nous avons employée soit principalement celle du krigeage. Myers propose aussi une présentation de ce type des interpolateurs radiaux, mais n'insiste pas sur les mêmes points. Par ailleurs, la forme de l'interpolateur fait apparaître des fonctions de la distance aux points de données, ce qui justifie le terme *interpolateurs radiaux*.

Ce chapitre diffère fondamentalement du précédent qui donne une présentation détaillée de l'approximation diffuse. Il s'agit plus ici de comprendre les idées sous-jacentes communes aux trois méthodes d'interpolation, ainsi que leurs différences. Aucun résultat n'est démontré, mais on pourra se reporter à

- Micchelli [Mic86] pour les *radial basis functions*,
- Chauvet [Cha94] pour le krigeage,
- Wahba [Wah90] pour les splines.

Les trois premières parties sont principalement une synthèse bibliographique, alors que la dernière aborde les fonctions de forme qui ne sont pas traitées dans la littérature.

### 3.1 Introduction

Dans ce chapitre, nous conservons les habitudes de notations utilisées dans la présentation de l'approximation diffuse.

#### 3.1.1 Radial Basis Functions

Micchelli dans [Mic86] propose une définition des *radial basis functions* comme un interpolateur de la forme suivante dans laquelle les fonctions  $K$  et  $f_i$  ainsi que l'entier  $m$  ( $m \leq n$ ) sont donnés *a priori* :

$$\tilde{u}(x) = \sum_{i=1}^n c_i K(\|x - x_i\|) + \sum_{j=0}^m d_j f_j(x) \quad (3.1)$$



Les inconnues  $c_i$  et  $d_j$  sont telles que

$$\forall j \in \llbracket 0, m \rrbracket, \sum_{i=1}^n c_i f_j(x_i) = 0 \quad (3.2)$$

et  $\tilde{u}(x)$  est une interpolation des valeurs  $u_i$  aux points de données  $x_i$  :

$$\forall k \in \llbracket 1, n \rrbracket, \tilde{u}(x_k) = \sum_{i=1}^n c_i K(\|x_k - x_i\|) + \sum_{j=0}^m d_j f_j(x_k) = u_k \quad (3.3)$$

Les équations (3.2) et (3.3) forment un système linéaire qui permet de calculer les coefficients  $(c, d)$  lorsque la fonction  $K$  est de type positif conditionnel (voir 3.1.4), et que les fonctions de base  $f_j$  sont linéairement indépendantes sur les points de donnée  $x_i$ .

### 3.1.2 Le krigeage ponctuel

Nous présentons ici le krigeage dans le formalisme du krigeage universel. Cette approche admet des limites théoriques importantes, qui peuvent être levées grâce à la théorie des fonctions aléatoires intrinsèques (FAI- $k$ ). La forme de l'interpolateur est la même dans les deux cas, mais le cadre des FAI- $k$  la rend plus générale.

L'hypothèse de base du cadre universel est de considérer que la fonction  $u$  que nous cherchons à interpoler est la réalisation d'une fonction aléatoire  $U$  de la forme

$$U(x) = V(x) + m(x)$$

où

- $V$  admet une covariance  $K(h) = \text{Cov}(V(x), V(x+h))$ , qui ne dépend pas de  $x$ ,
- $m(x)$  est déterministe de la forme  $\sum_l a_l f^l$ .

On forme un estimateur linéaire de  $U$

$$U^*(x) = \sum_i \lambda_i(x) U(x_i)$$

et des arguments statistiques nous amènent à un estimateur de la forme (3.1), où le noyau est la fonction de covariance  $K$ . L'interpolateur existe car la fonction de covariance est de type positif strict (conditionnel en FAI- $k$ ).

Suivant le cadre théorique dans lequel on se place, les contraintes de filtrage (3.2) ont une interprétation différente. Dans le cadre des FAI- $k$ , elles sont une condition d'existence des objets mathématiques que nous manipulons, alors que dans le cadre du krigeage universel, il s'agit de conditions de non-biais.

### 3.1.3 Splines

L'angle de vue des splines est fonctionnel. On cherche la fonction  $\tilde{u}$  qui passe par les points de données, et qui minimise un certain critère de rugosité.

L'exemple le plus célèbre est celui des splines plaque mince, qui (en 1d) cherche  $\tilde{u}$  dérivable deux fois telle que

- $\tilde{u}(x_i) = u_i$
- $\int_{\mathbb{R}} \left(\frac{d^2 u}{dx^2}\right)^2 dx$  soit minimale.

La forme de l'interpolateur est encore celui de la formule (3.1), et le noyau s'écrit dans ce cas particulier  $K(x) = \frac{1}{6}|x|^3$ .

Les conditions de filtrage (3.2) s'appliquent sur les monômes 1 et  $x$  qui sont une base du noyau du critère de rugosité. Cette propriété est générale, et elle joue exactement le même rôle que dans le krigeage (voir [Mye92]).

Wahba dans [Wah90] propose un cadre algébrique très général reposant sur la théorie des espaces hilbertiens à noyaux auto-reproduisants pour construire des fonctions splines générales. Nous utiliserons cette théorie pour construire l'approximation myope qui est une méthode intermédiaire entre interpolateurs radiaux et approximation diffuse.

Matheron démontre dans [Mat80] que les splines et le krigeage sont des interpolateurs algébriquement équivalents. La démonstration est justement basée sur la forme commune des interpolateurs de krigeage, splines et *radial basis functions*.

*Ce résultat nous permet de transposer toutes propriétés d'un interpolateur vers l'autre, et ainsi de travailler avec celui dont le formalisme est le plus avantageux en fonction de l'objectif.*

### 3.1.4 Fonctions de type positif conditionnel

La démonstration d'existence des interpolateurs radiaux reposent sur la notion de fonction de type positif conditionnel. Si les positions des données sont  $\mathcal{P}_k$ -unisolvantes, alors l'interpolateur construit avec un noyau de type positif conditionnel d'ordre  $k$  et des conditions de filtrage du même ordre existe.

Une fonction  $F(\cdot)$  est de type positif conditionnel d'ordre  $k$ , si pour tout ensemble de points,  $x_1, \dots, x_n \in \mathbb{R}^d$ , et toute combinaison linéaire  $(c_1, \dots, c_n)$  non nulle telle que

$$\sum_{i=1}^n c_i p(x_i) = 0, \forall p \in \mathcal{P}_k$$

la forme quadratique

$$\sum_{i=1}^n \sum_{j=1}^n c_i c_j F(\|x_i - x_j\|)$$

est strictement positive.

Le théorème de Bochner donne une caractérisation des fonctions de type conditionnel strict (pour toute combinaison linéaire) basée sur sa transformée de Fourier. Matheron généralise ce résultat pour les fonctions de type positif conditionnel dans [Mat73].

## 3.2 Choix du noyau

Nous avons vu que l'estimateur (3.1) existe dès que la fonction  $K$  est de type positif conditionnel, et que les fonctions  $f_j$  sont linéairement indépendantes sur la famille des points  $(x_i)_{i \in I}$ . Or il existe beaucoup de fonctions de type positif conditionnel. Le choix du noyau est donc un problème essentiel des interpolateurs radiaux.

### 3.2.1 Cas du krigeage

Le krigeage est l'interpolateur pour lequel le choix du noyau est le mieux formalisé. Son interprétation probabiliste lui fournit en effet un ensemble d'outils qui permet d'orienter le choix de la fonction de covariance.

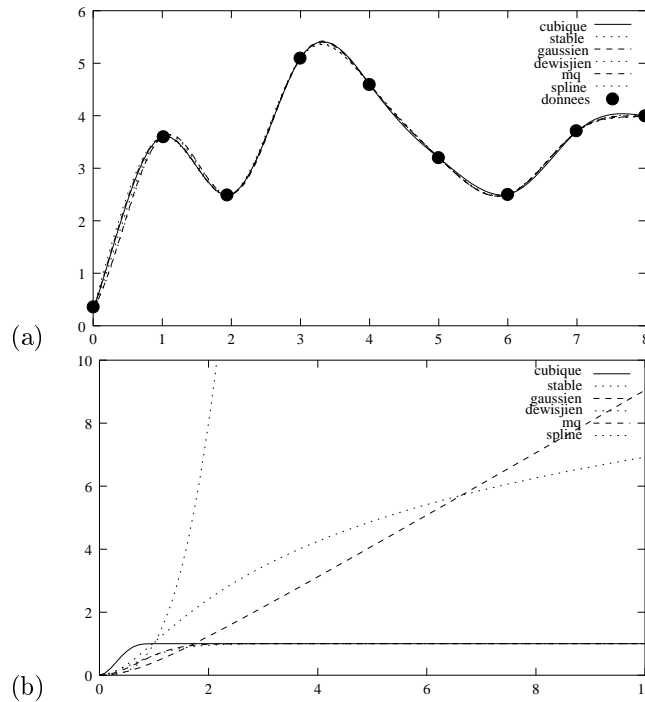


FIG. 3.1: interpolation de données 1d par plusieurs covariances généralisées d'ordre 1 (a), covariances correspondantes (b)

La méthode consiste en une analyse approfondie des données, afin de comprendre la structure spatiale des phénomènes sous-jacente à la grandeur étudiée, c'est l'**analyse structurale**, ou **analyse variographique**. L'outil de base est le variogramme expérimental.

### Le variogramme

Le variogramme se définit comme la covariance d'accroissements d'ordre 1, c'est une covariance généralisée d'ordre 0 (elle filtre les constantes):

$$\gamma(x, h) = \frac{1}{2} \text{Var}[U(x) - U(x + h)] = E[(U(x) - U(x + h))^2] \quad (3.4)$$

L'analyse des données à travers le variogramme permet de mieux comprendre la structure spatiale du phénomène, et guide vers le choix d'une forme de covariance. Une fois ce choix effectué, on ajuste la covariance aux données expérimentales. Dès que l'analyse variographique est terminée, on est en mesure de dérouler l'interpolation par krigeage des données.

### Commentaires

Cette méthode de travail s'inscrit dans un très grand respect des données car on ne leur impose pas un noyau arbitraire. Cependant, le nombre des modèles effectivement proposé dans les logiciels, et utilisé en pratique est extrêmement faible. La figure 3.1 montre que le choix de la covariance généralisée a parfois peu d'impact sur l'interpolation obtenue. Le cas contraire est aussi rapporté dans l'industrie minière!

Ceci illustre aussi le fait que l'interpolateur ne dépend finalement que des positions géographiques des points de données une fois que la fonction de covariance est choisie.

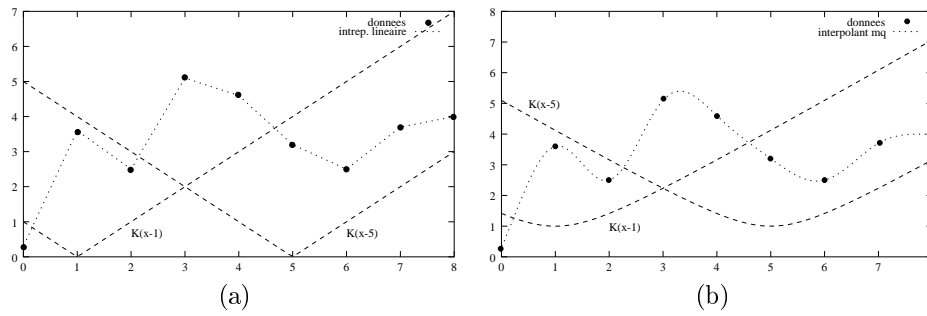


FIG. 3.2: *Interpolation et fonctions de base associées à 1 et 5 (a) noyau linéaire (b) noyau multiquadratique*

L'une des spécificités du krigeage est de proposer une estimation naturelle de l'erreur d'estimation par le biais de la variance d'estimation, mais, dès que la covariance généralisée est choisie et ajustée aux données, « *la variance d'erreur est plus caractéristique de la position géographique des données qu'une réelle quantification de l'erreur* » [Mye92].

### 3.2.2 Splines

Les splines sont particulièrement appréciées des utilisateurs pour deux raisons

1. La plus courante est la spline plaque mince, qui comme son nom l'indique répond à une analogie mécanique (la flexion des plaques minces),
2. Les surfaces splines ont en général un aspect visuel très agréable.

Le noyau est une conséquence de la fonctionnelle minimisée, et le cas  $h^2 \log(h)$  par exemple (spline plaque mince en 2d) ne permet aucun ajustement aux données. Cependant le choix du critère de rugosité permet, lorsqu'on dispose d'une analogie, de faire un choix judicieux.

### 3.2.3 Radial Basis Functions

Contrairement au krigeage, où la littérature traite largement le choix de la covariance, je n'ai trouvé aucune référence sur le sujet dans le cadre des *radial basis functions*.

Prenons cependant l'exemple de J. Allison dans [All93] pour comprendre les points qui motivent son choix. Il commence par développer le schéma pour  $K(x) = |x|$ , qui se confond en 1d avec un interpolation linéaire par morceaux. Cependant il désire obtenir une interpolation dérivable, il passe donc à  $K(x) = \sqrt{x^2 + \Delta^2}$  qui assure une interpolation  $C^1$ . Le paramètre  $\Delta$  est choisi en fonction de critère géométrique (distance au plus proche voisin).

D'après R. Schaback [Sch99], le choix du noyau pour les *radial basis functions* s'appuie principalement sur l'expérience de l'utilisateur sans utiliser sur une analyse des données comme dans le cas du krigeage. Il existe cependant un catalogue de fonctions candidates basé sur les travaux de Micchelli [Mic86]. Wendland [Wen98] propose des noyaux à supports compacts.

### 3.2.4 Validation croisée

Les trois méthodes d'interpolation utilisent la validation croisée comme mesure de qualité de l'interpolant. On trouve ces idées développées dans [Wah90] (dans le

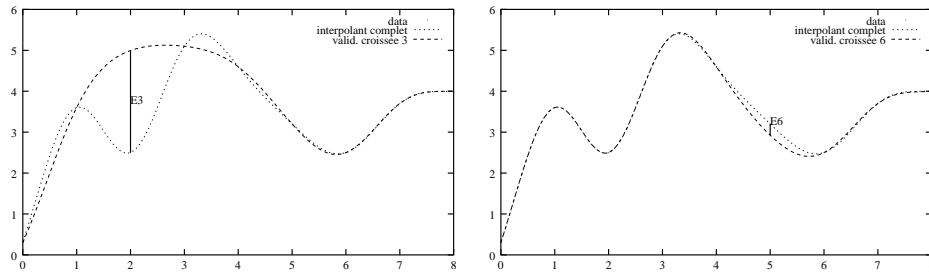


FIG. 3.3: Exemple de courbe obtenue par validation croisée, erreur mesurée.

cadre des splines d'ajustement), ou dans [GCK96]. Cette étape finit toute étude géostatistique.

Le principe de la validation croisée est de supprimer une donnée et de recalculer l'interpolation en ce point à partir des autres données. Notons  $u^{[k]}$  l'interpolant basé sur les données privées de  $u_k$ , alors on peut définir un critère de qualité de l'interpolation comme

$$Q^2 = \sum_{k=1}^n w_k (u_k - u^{[k]}(x_k))^2 \quad (3.5)$$

Ce critère est en réalité une fonction des paramètres du noyau, et on peut caler a posteriori ces paramètres en minimisant  $Q$ . C'est cette idée qui est développée dans [GCK96]; l'interpolation est construite avec le noyau  $K(x) = \sqrt{x^2 + \Delta^2}$ , mais où  $\Delta$  peut être différent en chaque point de donnée (dans ce cas, il n'existe pas de modèle probabiliste équivalent) : les  $\Delta_i$  sont calculés par minimisation de  $Q((\Delta_i)_{i \in \llbracket 1, n \rrbracket})$ .

### 3.3 Données bruitées

Il est rare de connaître des données sans erreur, et les valeurs aux points prennent alors la forme

$$y_k = u_k + \epsilon_k, \forall k \in \llbracket 1, n \rrbracket$$

où

- $y_k$  est la valeur mesurée,
- $u_k$  est la valeur « vraie »<sup>1</sup>,
- $\epsilon_k$  est l'erreur de mesure.

Dans ce cas, il peut sembler judicieux de lâcher la contrainte d'interpolation sur  $\hat{u}$ , pour proposer un modèle qui prenne cette erreur en compte. Le krigeage qui est probabiliste est bien adapté à ce genre de problématique, et les splines ont été adaptées pour y faire face. Je n'ai pas rencontré d'application des radial basis functions dans ce cadre.

---

1. si elle existe!

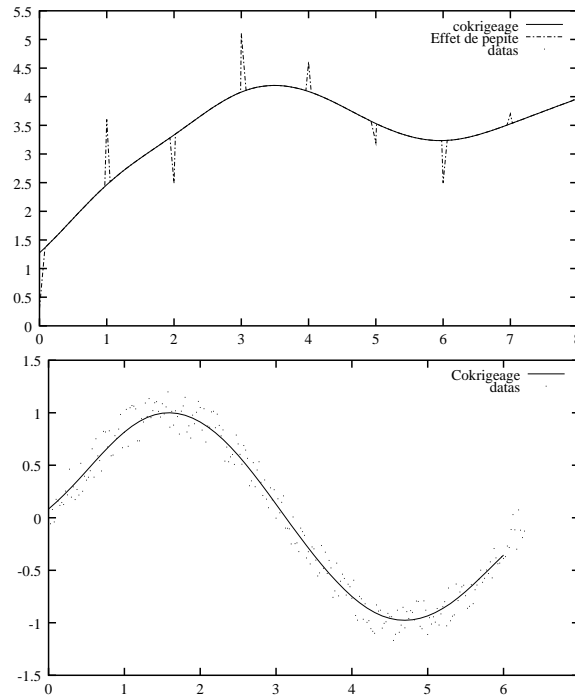


FIG. 3.4: Cokrigage d'erreur et effet de pépite, cokrigage d'erreur de données bruitées

### 3.3.1 Cokrigage d'erreur

#### Cadre théorique

Le cokrigage d'erreur propose le modèle suivant :  $\epsilon_k$  est une variable aléatoire et la famille  $(\epsilon_k)_k$  vérifie

- $\forall k, E[\epsilon_k] = 0$ , l'erreur est nulle en moyenne,
- $\forall k, l \in \llbracket 1, n \rrbracket^2$ ,  $\text{Cov}[\epsilon_k, U_l] = 0$ , l'erreur est indépendante de la mesure,
- $\forall k, l \in \llbracket 1, n \rrbracket^2$ ,  $\text{Cov}[\epsilon_k, \epsilon_l] = S_{kl}$ , les erreurs sont corrélées d'un point à un autre.

Ce modèle va nous permettre de développer une estimation de  $u$  à partir des données  $y_k$ .

Les coefficients de l'approximation sont alors calculés en remplaçant l'équation (3.3) par

$$\sum_{i=1}^n c_i (K(\|x_k - x_i\|) + S_{ki}) + \sum_{j=0}^m d_j f_j(x_k) = y_k, \forall k$$

#### Effet de pépite

On obtient un interpolateur presque partout identique au précédent dans le cas  $[S] = \alpha I$  en utilisant la notion d'effet de pépite. Cette notion signifie que les valeurs ne sont pas corrélées à très courte distance. La covariance admet alors la forme  $K(h) = \alpha \delta(h) + K'(h)$ , où  $\delta$  est un dirac en 0, et  $K'$  est une fonction de

covariance généralisée. On vérifie aisément que le système de krigeage est bien celui du cokrigeage, mais l'interpolateur fait un saut pour honorer les données comme le montre la figure 3.4.

Bien que les interpolateurs soient presque partout identiques dans ce cas particulier, le formalisme du cokrigeage d'erreur est plus général, et surtout mieux adapté au cas de données bruitées (on peut associer une covariance d'erreur par point de donnée). Cependant le cas  $[S] = \alpha I$  est exactement celui traité par les splines d'ajustement. Nous avons donc une analogie entre effet de pépite et coefficients d'ajustement des splines. Ceci est cohérent avec la pratique où l'on choisit un fort effet de pépite pour obtenir une carte lisse.

### 3.3.2 Splines d'ajustement

La construction des splines d'ajustement est basée sur un nouveau critère de rugosité, qui s'écrit dans le cas des splines plaques minces

$$\frac{1}{n} \sum_{k=1}^n (y_k - f(x_k))^2 + \lambda \int_{\mathbb{E}} \left( \frac{d^m f}{dx^m} \right)^2 dx \quad (3.6)$$

Il est construit en ajoutant au critère habituel le terme de gauche, qui prend en compte le bruit des données. Là encore, la solution est de la forme (3.1) et l'équation (3.3) est remplacée comme pour le cokrigeage par

$$y_k = \sum_{i=1}^n c_i (K(|x_k - x_i|) + n\lambda\delta_i^k) + \sum_{j=0}^m d_j f_j(x_k), \forall k \in \llbracket 1, n \rrbracket$$

On reconnaît bien un cokrigeage d'erreur avec  $[S] = n\lambda I$ .

La figure 3.5 recalculée à partir d'une illustration de [Wah90] montre que le choix de  $\lambda$  est très important.

Wahba propose d'utiliser la validation croisée pour évaluer la valeur optimale de  $\lambda$ . Pour éviter des temps de calcul prohibitifs, on remarque que le calcul des fonctions  $\tilde{u}^{[k]}$  est un sous-problème du calcul de  $\tilde{u}$ .

Par ailleurs, le choix  $\sum_{k=1}^n (y_k - \tilde{u}(x_k))^2$  est restrictif, et une première généralisation est

$$\sum_{k=1}^n w_k (y_k - \tilde{u}(x_k))^2$$

pour se rapprocher du cokrigeage d'erreur. Nous verrons dans la suite que Salkauskas propose même d'écrire  $w_k(x)$  dans [Sal92] pour tirer parti des avantages conjugués des splines et de l'approximation diffuse.

## 3.4 Fonctions de forme

Nous allons maintenant nous pencher plus particulièrement sur les fonctions de forme associées aux interpolateurs radiaux. Ceci est motivé par l'application de ces interpolateurs à la résolution d'équations aux dérivées partielles.

Dans une première partie, nous allons justifier l'existence des fonctions de forme des interpolateurs radiaux. Nous en profiterons pour énoncer quelques propriétés souhaitables de ces fonctions de formes.

La seconde partie regroupe quelques éléments connus sous le nom de « comportements des poids de krigeage » (voir [Riv84]), mais présentés dans une optique différente. Enfin la troisième partie montre quelques exemples issus de notre propre expérimentation.

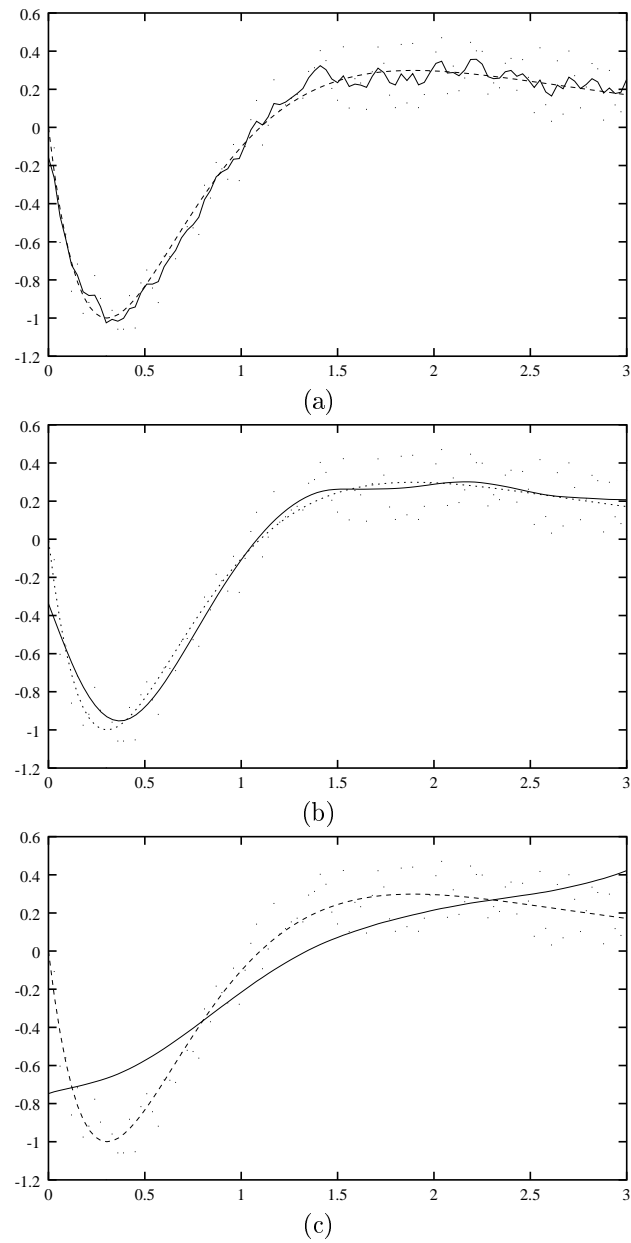


FIG. 3.5: De haut en bas, trois splines cubiques d'ajustement avec des valeurs de  $\lambda$  respectivement (a) trop petite, (b) correcte, (c) trop grande. Le calcul automatique de  $\lambda$  en s'appuyant sur la validation croisée permet de calculer sa valeur optimale (b).



### 3.4.1 Existence des fonctions de forme

Il existe au moins deux façons de justifier l'existence de fonctions de forme pour les interpolateurs radiaux (c'est à dire que ce sont des estimateurs linéaires).

1. Dans le cadre du krigeage, l'approximation  $\tilde{u}$  de  $u$  s'écrit

$$\tilde{u}(x) = \sum_{k=1}^n \lambda_k(x) u_k$$

et la propriété est contenue dans la définition.

2. Dans le cadre des *radial basis functions*, le système (3.3), (3.2)

$$\begin{cases} \tilde{u}(x_k) = \sum_{i=1}^n c_i K(\|x_k - x_i\|) + \sum_{j=0}^m d_j f_j(x_k) = u_k, \forall k \in \llbracket 1, n \rrbracket \\ \sum_{i=1}^n c_i f_j(x_i) = 0, \forall j \in \llbracket 0, m \rrbracket \end{cases}$$

montre que les coefficients  $(c, d)$  sont des combinaisons linéaires des données  $(u_k)_k$ , et ainsi l'existence d'une forme

$$\tilde{u}(x) = \sum_{k=1}^n N_k(x) u_k \quad (3.7)$$

de l'interpolant.

Quatre propriétés sont très appréciées lorsqu'on cherche à résoudre des équations aux dérivées partielles :

1. l'interpolation,  $N_i(x_j) = \delta_{ij}, \forall i, j \in \llbracket 1, n \rrbracket^2$ ,
2. le caractère local des supports des fonctions de forme,
3. la régularité des fonctions  $N_i$
4. la consistance, c'est à dire la représentation des polynômes jusqu'à un certain ordre.

Les interpolateurs radiaux vérifient la première propriété par construction. Il est facile de voir que les fonctions  $N_i(\cdot)$  sont des combinaisons linéaires des  $K(\|x_i - \cdot\|)$  et  $f_j(\cdot)$ , leur régularité est donc celle de ces fonctions. La quatrième propriété est elle aussi assurée par construction ; c'est le rôle des conditions de filtrage (3.2).

La seconde propriété fait l'objet des deux parties suivantes. Elle est essentielle d'un point de vue pratique.

### 3.4.2 Comportement des poids de krigeage

Nous allons nous intéresser plus particulièrement à deux phénomènes qui sont l'effet d'écran et le phénomène de relais. Un point essentiel de ces phénomènes est qu'ils interviennent pour des types de krigeage particuliers. Afin d'alléger le discours les termes suivants couramment utilisés en géostatistique linéaire seront employés

**Krigeage Simple (KS)** , krigeage sans condition de filtrage en supposant la fonction aléatoire de moyenne nulle.

**Krigeage Ordinaire (KO)** , krigeage avec condition de filtrage sur les constantes (ou à moyenne inconnue, voir [Cha94])

**Krigeage Intrinsèque d'ordre  $k$**  , krigeage avec conditions de filtrage sur  $\mathcal{P}_k$  ; le terme krigeage intrinsèque (KI) correspond au cas  $k = 0$ .

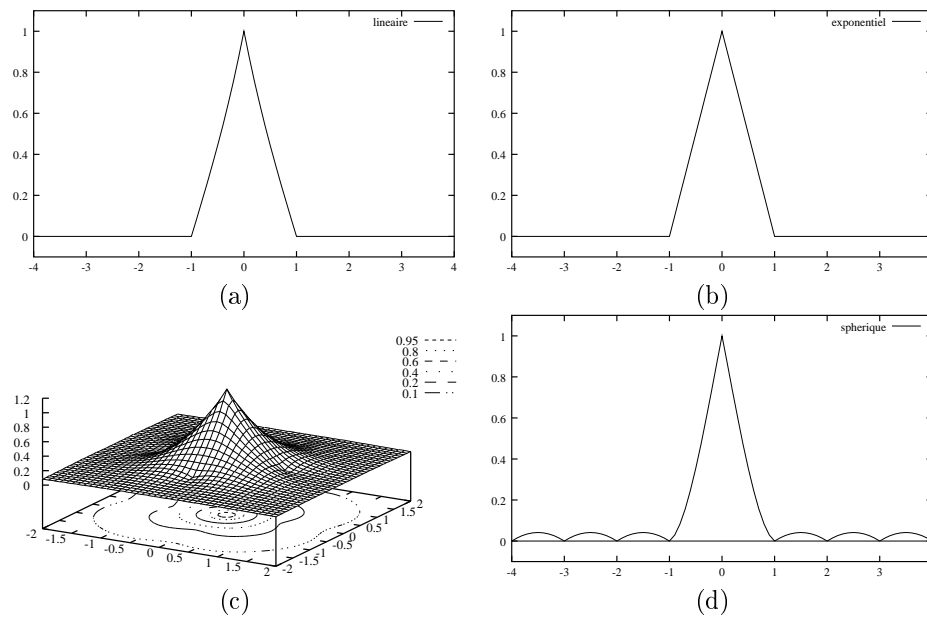


FIG. 3.6: Illustration des effets d'écran par les fonctions de forme, (a) variogramme linéaire, (b) covariance exponentielle, (c) variogramme de De Wisj, (d) variogramme sphérique.

### Effet d'écran

On parle d'effet d'écran total lorsque les fonctions de forme du krigeage ont un support borné. La terminologie exprime qu'une donnée fait écran aux suivantes, plus lointaines. Trois effets d'écran sont bien connus (voir par exemple [Mat70])

1. Krigeage Ordinaire avec covariance linéaire en 1d, les fonctions de forme sont alors celles de l'approximation linéaire par sous-domaine.
2. Krigeage Simple avec covariance exponentielle en 1d.
3. Krigeage Ordinaire de variogramme de De Wisj en 2d. Ce troisième cas est l'analogie de la cage de Faraday, car  $\log(r)$  est le potentiel harmonique de  $\mathbb{R}^2$ , comme  $\frac{1}{r}$  est celui de  $\mathbb{R}^3$ . Ainsi si l'on connaît un contour fermé informé, il limite les fonctions de forme de tout nœud intérieur à ce contour.

À ces trois cas s'ajoutent aussi les effets d'écran dits partiels, pour lesquels la fonction de forme prend une valeur faible, c'est le cas du sphérique en 1d (Krigeage Ordinaire).

### Effet relais

L'effet relais est l'inverse de l'effet d'écran. Les fonctions de forme gardent une valeur élevée sur la totalité de l'espace géographique d'intérêt. Bien que le cas de la figure 3.7 soit pathologique, c'est exactement l'opposé de ce que nous recherchons.

### Commentaires

De ces propriétés, l'effet d'écran peut sembler particulièrement intéressant, car il limite le support des fonctions de forme. Cependant, il apparaît dans des cas très particuliers, qui ne vérifient pas toujours les autres contraintes que de « bonnes »

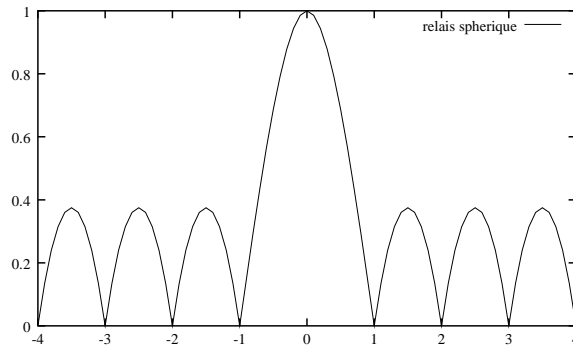


FIG. 3.7: *Krigeage simple (moyenne 1) avec covariance sphérique ( $r=1, a=0.5$ ).*

fonctions de forme devraient vérifier. En particulier, les fonctions de forme générées ne sont que continues (non dérivable aux points de données), et elles ne sont pas toujours consistantes.

Le meilleur moyen d'assurer la consistance est d'utiliser des conditions de filtrage; or ces conditions lient tous les points de données et étalent ainsi la fonction de forme sur tout l'espace. Ainsi ces propriétés ne nous sont pas utiles.

### 3.4.3 Quelques fonctions de forme en 2d

Pour aller plus loin, nous avons tracé les fonctions de forme associées à de nombreuses covariance  $C^1$  en 0. Les figures suivantes en montrent quelques-unes.

Le krigeage utilisé est intrinsèque d'ordre 1, car on demande une consistance d'ordre 1. Pour la plupart des fonctions de forme, on observe bien un effet d'écran, mais il n'est pas total. On s'aperçoit même que la fonction de forme se stabilise autour du terme constant de sa partie  $\sum d_j f_j$ . De plus cette stabilisation intervient dans la partie extrapolation de la fonction de forme, comme le montrent les figures où les points de données sont les points de coordonnées entières du carré  $[-3, 3] \times [-3, 3]$ .

Ces expérimentations confirment le constat de la partie précédente: les fonctions de forme du krigeage ne sont pas locales. Les logiciels de géostatistique proposent cependant de travailler en voisinage glissant, c'est-à-dire en basant l'approximation sur des points de données situés à l'intérieur d'un disque centré au point d'évaluation. Cette pratique génère des discontinuités lorsqu'un point sort ou entre dans le voisinage de krigeage, mais elles sont d'autant plus faibles que la valeur résiduelle des fonctions de forme le sont. Ce problème nécessite un traitement adapté lorsqu'on utilise directement  $\tilde{u}$  pour le tracé de courbes de niveaux<sup>2</sup>.

## 3.5 Conclusion

En conclusion, les splines, le krigeage et les *radial basis functions* forment une famille d'interpolants du même type: ils sont consistants, globaux (dans le cas général), et l'influence de chaque point diminue fortement avec la distance (il existe des théorèmes dans des cas particuliers). Les interpolants radiaux n'utilisent aucun maillage et leur régularité est liée à celle du noyau  $K$  qui génère les fonctions  $K_i(\cdot) = K(\|\cdot - x_i\|)$ . Ces deux points nous rappellent des propriétés et la définition de l'approximation diffuse, on se demande donc si on est capable de définir une approximation diffuse correspondant à un krigeage donné et inversement.

<sup>2</sup>. La stratégie de tracé est en général différente, voir [Dub81] par exemple

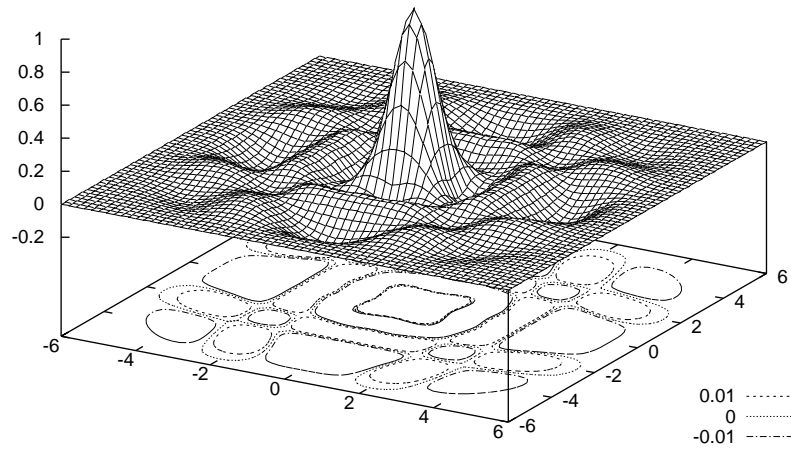


FIG. 3.8: *Fonction de forme associée à une covariance cubique. Cette covariance est à support borné.*

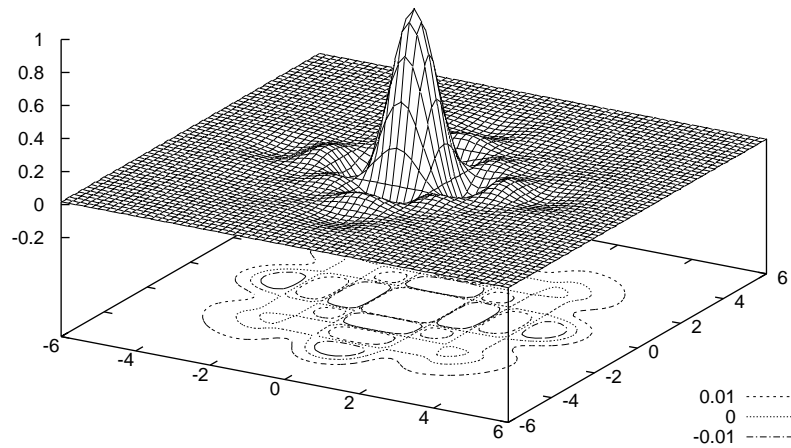


FIG. 3.9: *Fonction de forme associée à une covariance gaussienne.*

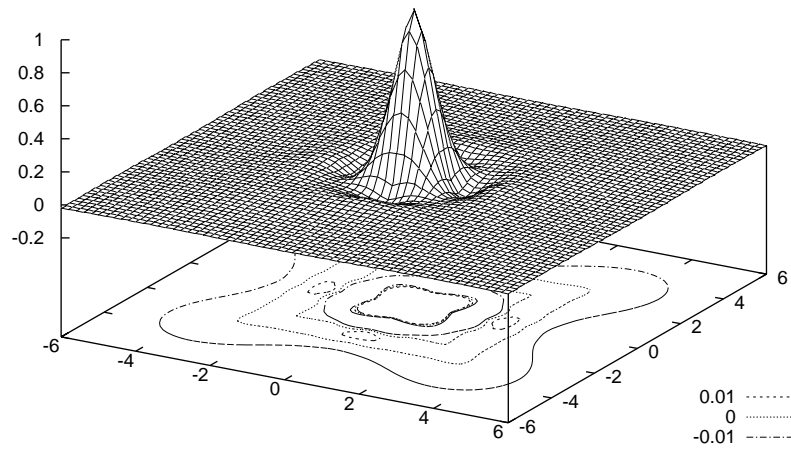


FIG. 3.10: *Fonction de forme associée à une covariance généralisée spline 2d.*

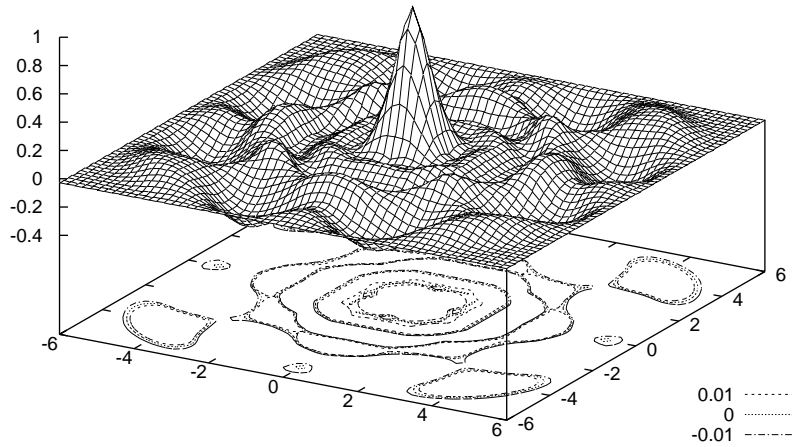


FIG. 3.11: *Fonction de forme associée à une covariance généralisée spline 2d tronquée (voir [Tro93]).*

Nous avons montré sur un exemple qu'il n'existe pas de théorèmes généraux [Sav98] et la définition de l'approximation myope dans la seconde partie montre qu'approximation diffuse et krigeage sont de natures différentes.

Au delà des aspects techniques, ce chapitre montre que le développement et le choix d'une méthode d'interpolation ou d'approximation sont étroitement liés à la nature du problème à traiter ou à la nature des données :

- Le krigeage est particulièrement bien adapté à des données éparses, structurées (au sens topo–probabiliste) et qui demandent un modèle représentant cette structure.
- Les splines plaques minces interpolent de façon lisse (agréable à l'oeil) grâce au modèle sous–jacent (en 1d, poutre en flexion sous l'hypothèse de Kirchhoff).
- L'interpolation par sous–domaines est utilisée pour résoudre des équations différentielles (pas de modèle dans la méthode d'interpolation).

Ceci montre finalement qu'il existe deux façons d'interpoler des données :

1. Définir un modèle (interpolateurs radiaux, modèles physiques, . . . )
2. Utiliser une méthode d'approximation/interpolation (interpolation par sous–domaines, approximation diffuse, interpolant de Sibson, . . . )

Là encore, c'est le type de problème à résoudre qui décide du choix de la famille de méthode: on préfère utiliser un modèle pour « remplir des trous », comme une méthode d'interpolation pour estimer une dérivée.

Deuxième partie

Extensions de  
l'approximation diffuse



# Chapitre 1

## Approximation Myope

L'approximation ou l'interpolation de données sans maillage dans le cas multidimensionnel s'appuie principalement sur deux types de méthode :

- Les méthodes radiales, que sont les splines, le krigeage ou les radial basis functions
- Les méthodes glissantes, approximation diffuse ou moving least square interpolation.

Salkauskas propose dans [Sal92] une méthode hybride en s'appuyant sur les travaux de Duchon [Duc76]. Nous reprenons la même idée dans le cadre monodimensionnel par souci de simplicité et en utilisant le cadre théorique décrit par Wahba dans [Wah90]. La méthode d'interpolation obtenue se généralise alors très facilement en utilisant le théorème d'équivalence entre splines et krigeage, et nous permet de passer au cas multidimensionnel. Nous définissons aussi une estimation de la dérivée par un opérateur de pseudo-dérivation.

Ce chapitre propose pour commencer une description du cadre fonctionnel dans lequel nous nous plaçons : ce sont les espaces à noyau reproduisant qui sont bien adaptés à la construction des splines. La modification du critère de construction des splines pour obtenir un schéma glissant demande de centrer leur construction au point d'évaluation. Nous construisons ensuite l'approximation myope, puis nous l'étudions d'un point de vue théorique. Nous terminons par quelques exemples issus de tests numériques.

### 1.1 Cadre Fonctionnel

Le cadre de travail que nous utilisons ici est celui décrit dans Wahba [Wah90]. Il s'agit d'espaces de fonctions définies sur un espace géographique  $\mathcal{T}$  que l'on munit d'une structure d'espace de Hilbert à noyau reproduisant.

Commençons par définir la notion d'**espace de Hilbert à noyau reproduisant** qui est sous-jacente à tous nos développements.

**Définition 13** *Soit  $\mathcal{H}$  un espace de Hilbert muni du produit scalaire  $(\cdot, \cdot)_{\mathcal{H}}$ . C'est un espace à noyau reproduisant si et seulement si pour tout  $t \in \mathcal{T}$ , la forme linéaire*

$$L_t \begin{cases} \mathcal{H} \mapsto \mathbb{R} \\ f \mapsto f(t) \end{cases}$$

*est continue et bornée.*



Les formes linéaire  $L_t$  sont continues et bornées si elles vérifient la propriété suivante

$$\begin{aligned} \forall t \in \mathcal{T}, \exists M_t \\ \forall f \in \mathcal{H}, |L_t f| = |f(t)| \leq M_t \|f\|_{\mathcal{H}^m} \end{aligned}$$

Si  $\mathcal{H}$  est un espace de Hilbert à noyau reproduisant (r.k.h.s. dans la suite<sup>1</sup>), alors les formes linéaires  $L_t$  sont reproduites par une fonction  $R_t$  de  $\mathcal{H}$  comme le montre le théorème suivant (voir [Aro50]) :

**Théorème 4** *Si  $\mathcal{H}$  est un r.k.h.s., alors*

$$\forall t \in \mathcal{T}, \exists R_t \in \mathcal{H}, \forall f, L_t f = (R_t, f)_{\mathcal{H}} = f(t)$$

Ce théorème nous conduit naturellement à la définition du noyau reproduisant associé au r.k.h.s.

**Définition 14** *Le noyau reproduisant associé à  $\mathcal{H}$  un r.k.h.s. est la fonction définie par*

$$\forall (t, s) \in \mathcal{T}^2, R(s, t) = (R_s, R_t)_{\mathcal{H}} = R_t(s) = R_s(t)$$

Le noyau reproduisant d'un r.k.h.s. est une fonction symétrique définie positive. On peut démontrer qu'à chaque r.k.h.s. est associé un unique noyau reproduisant  $R$  et réciproquement, qu'à chaque fonction définie positive  $R$  on peut associer un espace de Hilbert à noyau reproduisant.

Rappelons qu'une fonction  $R$  de deux variables sur  $\mathcal{T}$  est définie positive si pour tout  $n \in \mathbb{N}$ ,  $(a_1, \dots, a_n) \in \mathbb{R}^n$  non nul, et  $t_1, \dots, t_n \in \mathcal{T}$ ,

$$\sum_{i=1}^n \sum_{j=1}^n a_i a_j R(t_i, t_j) > 0$$

Ces quelques définitions et propriétés nous suffiront dans la suite, mais on pourra se reporter à [Aro50] pour plus de détails sur la théorie des r.k.h.s.

## 1.2 Structure de Hilbert des espaces de Sobolev $\mathcal{H}^m$

Nous allons définir sur  $\mathcal{H}^m$  un produit scalaire pour lequel il est un r.k.h.s. Cette structure est exposée dans [Wah90], mais certaines de nos démonstrations diffèrent.

### 1.2.1 Définitions

On s'intéresse dans toute la suite à des sous-espaces des fonctions définies de  $[a, b]$  dans  $\mathbb{R}$  vers  $\mathbb{R}$ .  $a$  et  $b$  sont bien entendu supposés distincts.

**Définition 15** *L'espace de Sobolev  $\mathcal{H}^m$  est défini par*

$$\mathcal{H}^m = \mathcal{H}^m(a, b) = \left\{ f, f', \dots, f^{(m-1)} \text{ absolument continues, } f^m \in \mathcal{L}^2(a, b) \right\}$$

---

1. Pour « Reproducing kernel Hilbert space »

Les fonctions de  $\mathcal{H}^m$  admettent un développement de Taylor à reste intégral en  $a$  (le développement de Taylor existe dans  $C^m$ , donc dans  $\mathcal{H}^m$  par densité)

$$\begin{aligned} \forall t \in \mathcal{T}, \quad f(t) &= \sum_{\nu=0}^{m-1} \frac{(t-a)^\nu}{\nu!} f^{(\nu)}(a) + \int_a^t \frac{(t-u)^{m-1}}{(m-1)!} f^{(m)}(u) du \\ &= \sum_{\nu=0}^{m-1} \frac{(t-a)^\nu}{\nu!} f^{(\nu)}(a) + \int_a^b \frac{Y(t-u)(t-u)^{m-1}}{(m-1)!} f^{(m)}(u) du \\ &= f_1(t) + f_2(t) \end{aligned}$$

où  $f_1$  et  $f_2$  sont définies par identification entre les deuxième et troisième lignes.

Ceci nous invite donc à définir les espaces  $\mathcal{H}_1^m$  et  $\mathcal{H}_2^m$  des fonctions  $f_1$  et  $f_2$ .

**Définition 16** Les espaces  $\mathcal{H}_1^m$  et  $\mathcal{H}_2^m$  sont définis par

$$\begin{aligned} \mathcal{H}_1^m &= \mathcal{P}_{m-1} = \text{span} \left\{ \frac{(t-a)^\nu}{\nu!}, \nu \in \llbracket 0, m-1 \rrbracket \right\} \\ \mathcal{H}_2^m &= \{ f \in \mathcal{H}^m, \forall \nu \in \llbracket 0, m-1 \rrbracket, f^{(\nu)}(a) = 0 \} \end{aligned}$$

Pour plus de clarté, nous noterons dans la suite  $\phi_\nu(t) = \frac{(t-a)^{\nu-1}}{(\nu-1)!}$  pour  $\nu \in \llbracket 1, m \rrbracket$ ; la famille des  $\phi_\nu$  est une base de  $\mathcal{H}_1^m$ .

### 1.2.2 Produits scalaires de $\mathcal{H}_1^m$ et $\mathcal{H}_2^m$

Commençons par démontrer la propriété suivante

**Propriété 11** La forme bilinéaire symétrique définie sur  $\mathcal{H}_1^m$  par

$$f, g \in \mathcal{H}_1^m, (f, g)_1 = \sum_{\nu=0}^{m-1} f^{(\nu)}(a) g^{(\nu)}(a)$$

est un produit scalaire sur  $\mathcal{H}_1^m$

En effet,  $f$  et  $g$  s'écrivent de façon unique sous la forme  $\sum_{\nu=0}^{m-1} a_\nu \phi_\nu$  et  $\sum_{\nu=0}^{m-1} b_\nu \phi_\nu$ .

Le produit  $(f, g)_1$  prend alors la forme  $\sum_{\nu=0}^{m-1} a_\nu b_\nu$ , car  $\phi_\nu^{(i)}(a) = \delta_\nu^i, i, \nu \in \llbracket 0, m-1 \rrbracket$ .

Il en découle que  $(\cdot, \cdot)_1$  est un produit scalaire et que la famille des  $\phi_\nu$  est une base orthonormée de  $\mathcal{H}_1^m$  pour  $(\cdot, \cdot)_1$ .

**Propriété 12** La forme bilinéaire symétrique  $(\cdot, \cdot)_2$  définie sur  $\mathcal{H}_2^m$  par

$$(f, g)_2 = \int_a^b f^{(m)}(u) g^{(m)}(u) du$$

est un produit scalaire pour  $\mathcal{H}_2^m$ .

Le caractère bilinéaire symétrique positif découle de la linéarité de la dérivation et du fait que  $(f, g)_2$  est le produit scalaire dans  $\mathcal{L}^2$  de  $f^{(m)}$  et  $g^{(m)}$ . Il reste juste à démontrer qu'une fonction de norme nulle est nulle. Pour cela, on remarque que le développement de Taylor de  $f \in \mathcal{H}_2^m$  se limite à son reste intégral, et que la condition  $\|f\|_2 = 0$  implique que  $f^{(m)}$  est nulle presque partout. Ainsi on a bien

$$\forall t \in \mathcal{T}, f(t) = \int_a^b \frac{Y(t-u)(t-u)^{m-1}}{(m-1)!} f^{(m)}(u) du = 0$$

ce qui assure la propriété.

### 1.2.3 Somme directe de $\mathcal{H}_1^m$ et $\mathcal{H}_2^m$

**Propriété 13**  $\mathcal{H}^m$  peut être muni d'un produit scalaire pour lequel  $\mathcal{H}_1^m$  et  $\mathcal{H}_2^m$  sont en somme directe :

1. La forme bilinéaire

$$(f, g) = (f, g)_1 + (f, g)_2 = (f_1, g_1)_1 + (f_2, g_2)_2$$

où  $f = f_1 + f_2$  et  $g = g_1 + g_2$  sont décomposées selon  $\mathcal{H}_1^m$  et  $\mathcal{H}_2^m$  est un produit scalaire sur  $\mathcal{H}^m$ .

2.  $\mathcal{H}_1^m$  et  $\mathcal{H}_2^m$  sont en somme directe pour ce produit scalaire :

$$\mathcal{H}^m = \mathcal{H}_1^m \oplus \mathcal{H}_2^m.$$

3.  $\int_a^b (f^{(m)})^2 = \|P_2 f\|_{\mathcal{H}_2^m}^2$  où  $P_2$  est la projection orthogonale sur  $\mathcal{H}_2^m$ .

La démonstration de ces propriétés découle immédiatement des définitions.

### 1.2.4 $\mathcal{H}^m$ est un espace de Hilbert

L'espace  $\mathcal{H}^m$  admet une structure pré-Hilbertienne. Nous devons maintenant démontrer que  $\mathcal{H}^m$  est complet pour la norme associée au produit scalaire  $(\cdot, \cdot)$ . Comme  $\mathcal{H}^m$  est somme directe de  $\mathcal{H}_1^m$  et  $\mathcal{H}_2^m$ , il nous suffit de démontrer que ces deux sous-espaces sont complets. Le cas de  $\mathcal{H}_1^m$  est évident, puisqu'il est de dimension finie.

Démontrons maintenant que  $\mathcal{H}_2^m$  est complet.

**Propriété 14** La norme  $\|\cdot\|_2$  est équivalente à la norme  $\|\cdot\|_{\mathcal{H}^m}$  dans  $\mathcal{H}_2^m$ .

Nous allons démontrer la propriété dans le cas  $m = 2$  en appliquant l'inégalité de Poincaré deux fois en cascade (inspiré de [Bre83] et [RT98]). Démontrons pour commencer que

$$\int_a^b (f')^2 \geq C \int_a^b f^2$$

Pour cela, on remarque que pour tout  $x \in ]0, 1]$ ,

$$\int_a^b (f')^2 \geq \int_a^x (f')^2$$

D'autre part, l'inégalité de Schwartz

$$\left( \int_a^x (f')^2 \right) \left( \int_a^x 1^2 \right) \geq \left( \int_a^x f'(t) 1(t) dt \right)^2$$

implique que

$$\int_a^x (f')^2 \geq \frac{1}{x-a} (f(x) - f(a))^2$$

Or sous l'hypothèse  $f \in \mathcal{H}_2^2$ ,  $f(a) = 0$ ; nous avons donc

$$\int_a^x (f')^2 \geq \frac{1}{x-a} f(x)^2$$

Cette inégalité et la remarque initiale assurent alors que

$$\int_a^b (f')^2 \geq \frac{1}{b-a} (f(x))^2, \forall x \in [a, b] \quad (1.1)$$

En intégrant cette dernière inégalité entre  $a$  et  $b$ , on obtient l'inégalité de Poincaré

$$\exists C > 0, \forall f \in \mathcal{H}_2^2, \|f'\|_{\mathcal{L}^2}^2 \geq C \|f\|_{\mathcal{L}^2}^2$$

L'application de cette inégalité à  $f''$  nous donne aussi (on utilise l'hypothèse  $f'(a) = 0$ )

$$\forall f \in \mathcal{H}_2^2, \|f''\|_{\mathcal{L}^2}^2 \geq C^2 \|f\|_{\mathcal{L}^2}^2$$

Comme  $\|f\|_{\mathcal{H}^2}^2 = \|f\|_{\mathcal{L}^2}^2 + \|f'\|_{\mathcal{L}^2}^2 + \|f''\|_{\mathcal{L}^2}^2$ , les inégalités de l'équivalence de norme s'écrivent

$$\begin{aligned} \exists K > 0, \forall f \in \mathcal{H}_2^2, \|f\|_{\mathcal{H}^2} &\leq K \|f\|_{\mathcal{H}^2} \\ \|f\|_{\mathcal{H}^2} &\leq \|f\|_{\mathcal{H}^2} \end{aligned}$$

Ceci assure donc le résultat pour  $m = 2$ . La démonstration suit le même raisonnement pour d'autre valeur de  $m$  en appliquant  $m$  fois l'inégalité de Poincaré. Or l'espace  $\mathcal{H}^m$  muni de la norme  $\|f\|_{\mathcal{H}^m}^2 = \sum_{\nu=0}^m \|f^{(\nu)}\|_{\mathcal{L}^2}^2$  est complet et démontre le théorème suivant.

**Théorème 5** *L'espace  $\mathcal{H}^m$  muni du produit scalaire  $(.,.)$  est complet : c'est un espace de Hilbert.*

## 1.3 Structure de r.k.h.s. des espaces de Sobolev $\mathcal{H}^m$

### 1.3.1 $\mathcal{H}^m$ est un r.k.h.s.

Comme on sait déjà que  $\mathcal{H}^m$  est un espace de Hilbert, il ne nous reste plus qu'à démontrer que les formes linéaires  $L_t$  sont continues. Reprenons l'inégalité (1.1) appliquée à  $f^{(m)}$  (elle est vraie car  $f \in \mathcal{H}_2^m$ )

$$\int_a^b (f^{(m)})^2 \geq \frac{1}{b-a} (f^{(m-1)}(t))^2, \forall t \in [a, b]$$

qui s'écrit aussi (en remarquant que la valeur absolue d'un réel lui est supérieure ou égale)

$$\|f\|_{\mathcal{H}_2^m} \geq \frac{1}{\sqrt{b-a}} f^{(m-1)}(t)$$

On intègre cette inégalité  $m-1$  fois de  $a$  à  $t$ , et comme précédemment en remarquant que pour tout  $\nu \in [0, m-1]$ ,  $f^{(\nu)}(a) = 0$  on a l'inégalité

$$\|f\|_{\mathcal{H}_2^m} \geq \frac{1}{(t-a)^{m-1}} \frac{1}{\sqrt{b-a}} f(t)$$

et comme  $t \leq b$

$$\|f\|_{\mathcal{H}_2^m} \geq \frac{1}{(b-a)^{m-\frac{1}{2}}} f(t) \quad (1.2)$$

Si  $f$  est positive, ceci assure la propriété. Dans le cas contraire, on décompose  $f$  en  $f = f^+ - f^-$  où  $f^+ = \sup(f, 0)$  et  $f^- = \sup(-f, 0)$ . Si  $f$  appartient à  $\mathcal{H}_2^m$ , alors  $f^+$  et  $f^-$  sont aussi dans  $\mathcal{H}_2^m$  et l'inégalité (1.2) peut leur être appliquée soit

$$\begin{aligned}\|f^+\|_{\mathcal{H}_2^m} &\geq \frac{1}{(b-a)^{m-\frac{1}{2}}} f^+(t) \\ \|f^-\|_{\mathcal{H}_2^m} &\geq \frac{1}{(b-a)^{m-\frac{1}{2}}} f^-(t)\end{aligned}$$

En élevant au carré et en ajoutant, on trouve

$$\|f^+\|_{\mathcal{H}_2^m}^2 + \|f^-\|_{\mathcal{H}_2^m}^2 \geq \frac{1}{(b-a)^{2m-1}} (f^+(t))^2 + (f^-(t))^2$$

Et par définition même de  $f^+$  et  $f^-$ , ces deux fonctions sont orthogonales dans  $\mathcal{H}_2^m$  (car leur supports sont disjoints), ce qui montre finalement que (par application du théorème de Pythagore)

$$\|f\|_{\mathcal{H}_2^m}^2 \geq \frac{1}{(b-a)^{2m-1}} (f(t))^2$$

et enfin

$$\|f\|_{\mathcal{H}_2^m} \geq \frac{1}{(b-a)^{m-\frac{1}{2}}} |f(t)|$$

Ainsi donc pour une famille de points  $x_i$  de l'intervalle  $]a, b[$ , on introduit des formes linéaires  $L_i$  (précédemment notées  $L_{x_i}$ ).

**Propriété 15** *Les formes linéaires  $L_i$*

$$L_i \begin{cases} \mathcal{H} \mapsto \mathbb{R} \\ f \mapsto f(x_i) \end{cases}$$

*sont continues et bornées.*

Et par définition, l'espace  $\mathcal{H}^m$  muni du produit scalaire  $(.,.)$  est un r.k.h.s.

### 1.3.2 Représentant des formes linéaires $L_i$

Nous verrons dans la suite que la forme même des fonctions  $\xi_i$  qui sont les représentants des formes linéaires  $L_i$  est très importante. Le calcul que nous proposons s'appuie sur trois points

- (i) On cherche  $\xi_i$  dans  $\mathcal{H}_2^m$
- (ii) Toute fonction  $f$  de  $\mathcal{H}_2^m$  admet un développement de Taylor à reste intégral
- (iii)  $\xi_i$  est représentant de  $L_i$  dans  $\mathcal{H}_2^m$

Chaque fonction  $f$  de  $\mathcal{H}_2^m$  admet un développement de Taylor à reste intégral dont la partie polynomiale est nulle (car  $\forall \nu \in \llbracket 0, m-1 \rrbracket, f^{(\nu)}(a) = 0$ )

$$f(x_i) = \int_a^b \frac{Y(x_i-x)(x_i-x)^{m-1}}{(m-1)!} f^{(m)}(x) dx$$

Ceci signifie donc que  $\xi_i^{(m)}(x) = \frac{Y(x_i-x)(x_i-x)^{m-1}}{(m-1)!}$  d'où

$$\xi_i(x) = \frac{(-1)^m}{(2m-1)!} Y(x_i-x)(x_i-x)^{2m-1} + p_i(x)$$

avec  $p_i \in \mathcal{P}_{m-1}$ . On doit alors ajuster le polynôme  $p_i$  pour assurer  $\xi_i \in \mathcal{H}_2^m$ , c'est à dire

$$\xi_i(a) = \xi_i'(a) = \dots = \xi_i^{(m-1)}(a) = 0 \quad (1.3)$$

Le polynôme  $p_i$  peut s'écrire sous la forme

$$p_i(x) = \alpha_0 + \alpha_1(x-a) + \dots + \alpha_{m-1}(x-a)^{m-1}$$

et les conditions (1.3) donnent le système qui nous permet de calculer les coefficients  $\alpha_i$

$$\left\{ \begin{array}{l} \frac{(-1)^{m+1}(x_i-a)^{2m-1}}{(2m-1)!} = \alpha_0 \\ \frac{(-1)^m(x_i-a)^{2m-2}}{(2m-2)!} = \alpha_1 \\ \vdots \\ \frac{(-1)^2(x_i-a)^m}{m!} = (m-1)! \alpha_{m-1} \end{array} \right.$$

et se résout en  $\alpha_i = \frac{1}{i!} \frac{(-1)^{m+1-i}(x_i-a)^{2m-1-i}}{(2m-1-i)!}$ ,  $i \in \llbracket 0, m-1 \rrbracket$ .

**Propriété 16** Les fonctions  $\xi_i$  représentant les formes linéaires  $L_i$  dans  $\mathcal{H}_2^m$  ont la forme

$$\begin{aligned} \xi_i(x) &= \frac{(-1)^m Y(x_i-x)(x_i-x)^{2m-1}}{(2m-1)!} + p_i(x) \\ \text{avec} \quad & \left\{ \begin{array}{l} p_i(x) = \sum_{\nu=0}^{m-1} \alpha_i(x-a)^\nu \\ \alpha_i = \frac{1}{i!} \frac{(-1)^{m+1-i}(x_i-a)^{2m-1-i}}{(2m-1-i)!} \end{array} \right. \end{aligned} \quad (1.4)$$

**Remarque 1** On a bien (voir [Wah90])

$$\begin{aligned} \xi_i &\in \mathcal{P}_{2m-1} \text{ si } x \in [a, x_i] \\ \xi_i &\in \mathcal{P}_{m-1} \text{ si } x \in [x_i, b] \end{aligned}$$

On peut vérifier que la formule  $\xi_i(x_j) = \xi_j(x_i)$  issue de la symétrie du produit scalaire et de la propriété de reproduction des  $\xi_i$  est bien vraie avec cette expression. Nous allons maintenant démontrer la propriété suivante

**Propriété 17** La famille des  $(\xi_i)_{i \in [1, n]}$  est libre si et seulement si les points  $x_i$  sont distincts.

Pour cela, on utilise la forme des fonctions  $\xi_i = \gamma_i + p_i$  démontrée plus haut, où  $\gamma_i(x) = \frac{(-1)^m Y(x_i-x)(x_i-x)^{2m-1}}{(2m-1)!}$  et  $p_i$  est un polynôme de degré inférieur ou égal à  $m-1$ . On commence par démontrer que les fonctions  $\gamma_i$  sont indépendantes linéairement dès que les points  $x_i$  sont distincts, et différents de  $a$  et de  $b$  (cette dernière hypothèse n'est pas restrictive, car on peut toujours prendre un interval de travail plus grand, elle interviendra aussi dans la deuxième phase de la démonstration). Cherchons des réels  $\mu_i$  tels que la combinaison linéaire  $\sum_{i=1}^n \mu_i \gamma_i$  soit nulle. Si on suppose que les  $x_i$  sont distincts et ordonnés par ordre croissant, alors pour  $y_n = \frac{x_{n-1} + x_n}{2}$ , l'équation s'écrit

$$\mu_n \gamma_n(y_n) = \mu_n \frac{(x_n - y_n)^{2m-1}}{(2m-1)!} = 0$$

d'où on tire  $\mu_n = 0$  car  $x_n \neq x_{n-1}$ . On passe alors à  $y_{n-1} = \frac{x_{n-2} + x_{n-1}}{2}$  qui avec la condition  $\mu_n = 0$  s'écrit

$$\mu_{n-1} \gamma_{n-1}(y_{n-1}) = \mu_{n-1} \frac{(x_{n-1} - y_{n-1})^{2m-1}}{(2m-1)!} = 0$$

qui implique de la même façon  $\mu_{n-1} = 0$ . On continue ainsi de suite jusqu'à  $y_1 = (x_1 + a)/2$ , et démontre la propriété.

On cherche maintenant des réels  $\mu_i$  qui annulent  $\sum \mu_i \xi_i = \sum \mu_i \gamma_i + \sum \mu_i p_i$ . La somme de gauche est nulle sur  $[x_n, b]$  et non nulle sur  $[a, x_1]$  qui sont des segments de mesure non nulle sous nos hypothèses. Ce ne peut pas être un polynôme. Cependant, la somme de droite est un polynôme de degré inférieur ou égal à  $m - 1$ . La condition  $\sum \mu_i \xi_i = 0$  s'écrit donc  $\sum \mu_i \gamma_i = 0$  et  $\sum \mu_i p_i = 0$ ; or la démonstration précédente montre que les  $\mu_i$  doivent être tous nuls. Les fonctions sont donc libres si les points de donnée  $x_i$  sont distincts. Par ailleurs s'il existe deux points confondus, alors les fonctions  $\xi_{i_1}$  et  $\xi_{i_2}$  associées sont égales, la famille n'est pas libre. Cela achève la démonstration.

Notons pour finir que le même type de structure peut se construire dans le cas multidimensionnel comme le montre le travail de Duchon [Duc76].

La structure de  $\mathcal{H}^m$  que nous venons de construire correspond à la métrique naturelle pour développer les splines plaques minces. Nous commençons par décrire leurs calculs, puis on passe à la construction d'une approximation glissante, quelques propriétés de cette approximation sont ensuite étudiées.

## 1.4 Splines d'ajustement

### 1.4.1 Introduction

Le modèle de données associé au *special spline smoothing problem* est

$$y_i = f_e(x_i) + \epsilon_i, i \in \llbracket 1, n \rrbracket$$

où les données mesurées  $y_i$  en  $n$  points  $x_i$  de  $]a, b[$  sont considérées comme les valeurs d'une fonction  $u_e \in \mathcal{H}^m$  entachées d'erreurs  $\epsilon_i$  que l'on suppose normales indépendantes et de variance  $\frac{1}{w_i}$ . On note dans la suite  $W$  la matrice diagonale des  $w_i$ .

On cherche alors comme approximation de  $f_e$  la fonction  $f$  qui minimise le critère

$$\mathcal{E}_W^\lambda = \sum_{i=1}^n w_i (y_i - f(x_i))^2 + \lambda \int_a^b (f^{(m)})^2$$

dans  $\mathcal{H}^m$ . Ce critère signifie qu'on cherche  $f$  qui passe au plus près des points de données (au sens des moindres carrés) et qui respecte un certain critère de régularité (second terme). Comme on cherche  $f \in \mathcal{H}^m$ , on utilise la structure que nous avons décrite dans la première partie pour transformer  $\mathcal{E}_W^\lambda$  en (rappel:  $L_i = L_{x_i}$ )

$$\mathcal{E}_W^\lambda(f) = \sum_{i=1}^n w_i (y_i - L_i f)^2 + \lambda \|P_2 f\|_{\mathcal{H}^m}^2 \quad (1.5)$$

C'est le *general spline smoothing problem* qui se pose pour un r.k.h.s. général  $\mathcal{H}$  qui se décompose de la même façon que  $\mathcal{H}^m$  en une somme directe  $\mathcal{H}_1^m \oplus \mathcal{H}_2^m$ ,  $\mathcal{H}_1^m$  de dimension finie.

Ce problème est résolu dans [Wah90] dans le cas  $W = Id$ , [Sal92] pour  $W$  diagonale et enfin [Mat80] pour le cas général  $W$  symétrique<sup>2</sup>.

### 1.4.2 Transformation de $\mathcal{E}_W^\lambda$ et calcul de $f_W^\lambda$

Comme  $\mathcal{H}^m$  est un r.k.h.s., on peut écrire  $L_i f = (\eta_i, f)_{\mathcal{H}^m}$ . On remarque aussi que les fonctions  $\eta_i$  forment une famille libre dès que les points  $x_i$  sont distincts

<sup>2</sup>. ce dernier cas sort de notre propos

(démonstration:  $\sum \mu_i \eta_i = 0$  implique  $P_2(\sum \mu_i \eta_i) = 0$ , soit  $\sum \mu_i \xi_i = 0$  et donc  $\mu_i = 0$  pour tout  $i$  car les  $\xi_i$  sont libres).

Dans un second temps, on décompose la fonction  $f$  selon la somme directe  $\mathcal{H}_1^m \oplus \mathcal{H}_2^m$  en remarquant que

- $(\phi_\nu)_{\nu \in [0, m-1]}$  est une base de  $\mathcal{H}_1^m$
- $\mathcal{H}_2^m$  se décompose en une partie générée par la famille libre  $(\xi_i)_{i \in [1, n]}$  (famille des projetés des  $\eta_i, i \in [1, n]$ ) et son orthogonal

soit

**Propriété 18** *Pour toute fonction  $f$  de  $\mathcal{H}^m$  il existe  $d_1, \dots, d_m$  et  $c_1, \dots, c_n$  des réels et  $\rho$  une fonction orthogonale à tous les  $\phi_\nu$  et  $\xi_i$  tels que*

$$f = \sum_{\nu=0}^{m-1} d_\nu \phi_\nu + \sum_{i=1}^n c_i \xi_i + \rho$$

Nous noterons dans la suite  $c$  et  $d$  les vecteurs des  $c_i$  et  $d_\nu$ .

**Remarque 2** *La fonction  $\rho$  est orthogonale aux fonctions  $\eta_i$*

En effet, il est évident que  $\rho$  est dans  $\mathcal{H}_2^m$  (elle est orthogonale aux  $\phi_\nu$ ) et  $\eta_i - \xi_i$  appartient à  $\mathcal{H}_1^m$ , ainsi on a  $(\rho, \eta_i) = (\rho, \xi_i)$ . Comme nous avons choisi  $\rho$  orthogonale aux  $\xi_i$ , il vient finalement  $(\rho, \eta_i) = 0$  pour tout  $i$ . De la même façon, on montre que

$$\forall i, j \in [1, n]^2, (\eta_i, \xi_j) = (\xi_i, \xi_j)$$

Pour finir explicitons le calcul partiel

$$\begin{aligned} (\eta_i, f) &= \sum_{\nu=1}^m d_\nu (\eta_i, \phi_\nu) + \sum_{j=1}^n c_j (\eta_i, \xi_j) + (\eta_i, \rho) \\ &= \sum_{\nu=1}^m d_\nu \phi_\nu(x_i) + \sum_{j=1}^n c_j (\xi_i, \xi_j) \end{aligned}$$

Le critère de construction de la fonction spline s'écrit alors

$$\mathcal{E}_W^\lambda(f) = [y - Gc - Pd]^\top W[y - Gc - Pd] + \lambda c^\top Gc + \lambda \|\rho\|_{\mathcal{H}^m}^2 \quad (1.6)$$

où nous notons  $G = [(\xi_i, \xi_j)]_{i,j}$  le grammien des fonctions  $\xi_i$  qui est une famille libre (donc  $G$  est inversible).  $P = [L_i \phi_\nu]_{i,\nu}$  qui est supposée de rang  $m$ .

Ainsi, la première condition de minimisation est  $\rho = 0$ , et le calcul en  $(c, d)$  mène au système

$$\begin{cases} (G^\top W G + \lambda G)c + G^\top W P d &= G W y \\ P^\top W G c + P^\top W P d &= P^\top W y \end{cases}$$

ce qui est équivalent à ( $G$  est symétrique et inversible si les points de données  $x_i$  sont distincts)

$$\begin{cases} W y &= (W G + \lambda I)c + W P d \\ P^\top W G c + P^\top W P d &= P^\top (W G + \lambda I)c + P^\top W P d \end{cases}$$

Finalement, nous avons le théorème suivant

**Théorème 6** *La fonction  $f_W^\lambda$  d'approximation des données  $y_i$  aux points  $x_i$  minimisant le critère  $\mathcal{E}_W^\lambda$  a la forme*

$$f = \sum_{\nu=1}^m d_\nu \phi_\nu + \sum_{i=1}^n c_i \xi_i$$



et les vecteurs  $c$  et  $d$  sont donnés par le système (si  $\lambda > 0$ )

$$\begin{bmatrix} G + \lambda W^{-1} & P \\ P^\top & 0 \end{bmatrix} \begin{Bmatrix} c \\ d \end{Bmatrix} = \begin{Bmatrix} y \\ 0 \end{Bmatrix} \quad (1.7)$$

On peut remarquer que la forme de la matrice  $W$  n'influe pas sur la démonstration de ce théorème (si  $W$  est l'inverse d'une matrice de covariance, elle est bien symétrique définie positive, ce qui assure la correction des développements).

### 1.4.3 Régularité du système d'optimalité

Pour démontrer que le système (1.7) est régulier, il suffit de montrer que le système homogène admet comme unique solution  $(c, d) = 0$ . Posons  $S = G + \lambda W^{-1}$  qui est inversible. La solution du système homogène s'écrit alors

$$\begin{cases} c & = -S^{-1}Pd \\ (P^\top S^{-1}P)d & = 0 \end{cases}$$

Comme  $S$  est symétrique définie positive,  $S^{-1}$  l'est aussi. Or  $P^\top S^{-1}Pd = 0$  implique  $d^\top P^\top S^{-1}Pd = 0$  soit  $(Pd)^\top S^{-1}(Pd) = 0$  et donc  $Pd = 0$ . Ceci n'implique  $d = 0$  que dans le cas où l'on a  $n \geq m$  points distincts ( $P$  doit être injective).

**Propriété 19** *Si  $n \geq m$ , les points de données  $x_i$  sont distincts, et  $\lambda > 0$  alors le système de construction (1.7) est régulier.*

Calculons la solution formelle du système (1.7) sous ces hypothèses (on pose toujours  $S = G + \lambda W^{-1}$ )

$$\begin{cases} Sc + Pd & = y \\ P^\top c & = 0 \end{cases}$$

équivalent à

$$\begin{cases} c & = S^{-1}(y - Pd) \\ 0 & = P^\top S^{-1}y - P^\top S^{-1}Pd \end{cases}$$

et finalement

$$\begin{cases} c & = S^{-1}(I - (P^\top S^{-1}P)^{-1}P^\top S^{-1})y \\ d & = (P^\top S^{-1}P)^{-1}P^\top S^{-1}y \end{cases} \quad (1.8)$$

**Remarque 3** *La condition  $n \geq m$  est nécessaire.*

Étudions le cas  $m = 2, n = 1$  par souci de simplicité, le système (1.7) s'écrit alors

$$\begin{cases} sd + c_1 + x_1 c_2 & = u_1 \\ d & = 0 \\ x_1 d & = 0 \end{cases}$$

Il admet alors une infinité de solution  $d = 0, (c_1, c_2) = (u_1 - lx_1, l), l \in \mathbb{R}$ .

### 1.4.4 Cas particuliers, consistance

Nous continuons notre analyse du système (1.7) par deux remarques concernant des cas particuliers.

**Remarque 4** *Si  $n = m$  alors l'approximation construite est l'interpolation polynomiale des données.*

En effet, l'interpolation polynomiale des données annule dans ce cas  $\mathcal{E}_W^\lambda$ . Le vecteur des  $w_i$  ne joue plus aucun rôle: nous ferons donc l'hypothèse  $n > m$  dans la suite.

**Propriété 20** *La fonction spline est consistante: les polynômes de degré inférieur ou égal à  $m - 1$  (c'est à dire  $\mathcal{H}_1^m$ ) sont reproduits exactement.*

En considérant  $y_i = p(x_i)$ ,  $p \in \mathcal{P}_{m-1}$ , il suffit de remarquer que  $p$  annule le critère de construction  $\mathcal{E}_W^\lambda$ .

**Propriété 21** *La fonction  $f_W^\lambda$  interpole le point de donnée  $x_j$  lorsque  $w_j \rightarrow \infty$ .*

Remarquons que la ligne  $j$  du système d'optimalité s'écrit

$$\sum_{\nu=1}^m d_\nu \phi_\nu(x_j) + \sum_{i=1}^n c_i \xi_i(x_j) = u_j - \frac{\lambda c_j}{w_j}$$

car  $(\eta_j, \xi_i) = (\xi_j, \xi_i) = \xi_i(x_j)$  pour tout  $i, j \in \llbracket 1, n \rrbracket$ . Et nous avons donc

$$f_W^\lambda(x_j) = u_j - \frac{\lambda c_j}{w_j}$$

Ainsi, lorsque  $w_j$  tend vers l'infini, la ligne  $j$  du système d'optimalité devient une condition d'interpolation au point  $x_j$ .

Finissons enfin par une dernière remarque concernant les conditions de consistance

**Remarque 5** *On peut remplacer les conditions de consistances centrées en  $a$  par des conditions de consistances centrées en un autre point sans changer l'approximation.*

On démontre de nouveau cette propriété dans le cas  $m = 2$  car la preuve est purement calculatoire, la propriété reste vraie dans le cas général. Les conditions de consistance s'écrivent

$$\begin{cases} \sum_{i=1}^n c_i = 0 \\ \sum_{i=1}^n c_i(x_i - a) = 0 \end{cases}$$

la première équation montre que la seconde se réduit à  $\sum_{i=1}^n c_i x_i$  car  $a \sum_{i=1}^n c_i = 0$ , ou

de façon équivalente à  $\sum_{i=1}^n c_i(x_i - y)$  pour n'importe quel point  $y$ .

### 1.4.5 Système équivalent

La fonction spline d'ajustement que nous venons de construire fait intervenir la borne  $a$  de son intervalle de définition. Nous aimerions supprimer cette dépendance car elle donne un rôle prépondérant à  $a$  qui est choisi de façon arbitraire.

Le système d'optimalité (1.7)

$$\begin{bmatrix} G + \lambda W^{-1} & P \\ P^\top & 0 \end{bmatrix} \begin{Bmatrix} c \\ d \end{Bmatrix} = \begin{Bmatrix} y \\ 0 \end{Bmatrix}$$

dont la ligne  $i$  s'écrit

$$\sum_{j=1}^n \xi_j(x_i) c_j + \frac{\lambda}{w_i} c_i + \sum_{\nu=1}^m \phi_\nu(x_i) d_\nu = u_i \quad (1.9)$$

conserve la même forme si l'on centre les polynômes  $\phi_\nu$  en un autre point  $x$  que  $a$  selon la formule :  $\phi_\nu(y) = \frac{(y-x)^{\nu-1}}{(\nu-1)!}$ , car ce n'est qu'un changement de base dans  $\mathcal{H}_1^m$ . Nous cherchons maintenant à éliminer  $a$  de  $G$ .

Afin de simplifier les calculs, nous nous plaçons dans le cas  $m = 2$ . Nous allons transformer la fonction  $\xi_i$  pour faire apparaître une covariance généralisée  $K$  : on remarque que le terme  $\frac{1}{6}Y(x_i - x)(x_i - x)^3$  peut aussi s'écrire  $\frac{1}{12}|x_i - x|^3 + \frac{1}{12}(x_i - x)^3$ . Ainsi, on a après quelques lignes de calculs

$$\begin{aligned} \xi_i(x) &= \frac{1}{12}|x_i - x|^3 \\ &\quad - \frac{1}{12}(x_i - a)^3 + \frac{1}{4}(x_i - a)^2(x - a) + \frac{1}{4}(x_i - a)(x - a)^2 - \frac{1}{12}(x - a)^3 \end{aligned}$$

Injectons cette forme de  $\xi_i$  dans la première somme de (1.9) en posant  $K(h) = \frac{1}{12}|h|^3$  :

$$\begin{aligned} \sum_{j=1}^n c_j \xi_j(x_i) &= \sum_{j=1}^n c_j K(x_j - x_i) \\ &\quad - \frac{1}{12}(x_i - a)^3 \sum_{j=1}^n c_j + \frac{1}{4}(x_i - a)^2 \sum_{j=1}^n c_j (x_j - a) \\ &\quad + \frac{1}{4}(x_i - a) \sum_{j=1}^n (x_j^2 - 2ax_j + a^2)c_j \\ &\quad - \frac{1}{12} \sum_{j=1}^n (x_j^3 - 3x_j^2a + 3x_ja^2 - a^3)c_j \end{aligned}$$

Or les conditions d'orthogonalité du système (1.7) imposent que  $\sum_{j=1}^n d_j = 0$  et  $\sum_{j=1}^n d_j(x_j - a) = 0$ . Ainsi l'expression précédente s'écrit

$$\begin{aligned} \sum_{j=1}^n c_j \xi_j(x_i) &= \sum_{j=1}^n K(x_j - x_i)c_j + \frac{1}{4}(x_i - a) \sum_{j=1}^n c_j x_j^2 \\ &\quad - \frac{1}{12} \sum_{j=1}^n x_j^3 c_j + \frac{3}{12}a \sum_{j=1}^n x_j^2 c_j \\ &= \sum_{j=1}^n K(x_j - x_i)c_j + \frac{1}{4}x_i \sum_{j=1}^n c_j x_j^2 - \frac{1}{12} \sum_{j=1}^n c_j x_j^3 \end{aligned}$$

Nous avons donc éliminé  $a$  de cette somme. Montrons maintenant que la correction  $\frac{1}{4}x_i \sum_{j=1}^n c_j x_j^2 - \frac{1}{12} \sum_{j=1}^n c_j x_j^3$  peut toujours être centrée en un point  $x$  que nous choisissons. Pour cela, on remarque que

$$\sum_{j=1}^n (x_j - x)^2 c_j = \sum_{j=1}^n x_j^2 c_j - 2x \sum_{j=1}^n x_j c_j + x^2 \sum_{j=1}^n c_j = \sum_{j=1}^n x_j^2 c_j$$

grâce aux conditions d'orthogonalité, et de la même façon,

$$\sum_{j=1}^n x_j^3 c_j = \sum_{j=1}^n (x_j - x)^3 c_j + 3x \sum_{j=1}^n c_j (x_j - x)^2$$

Nous injectons ces deux formules dans l'expression précédente pour obtenir

$$\begin{aligned} \sum_{j=1}^n c_j \xi_j(x_i) &= \sum_{j=1}^n K(x_j - x)c_j + \frac{1}{4}(x_i - x) \sum_{j=1}^n (x_j - x)^2 c_j - \frac{1}{12} \sum_{j=1}^n (x_j - x)^3 c_j \end{aligned} \tag{1.10}$$

Pour finir, on utilise (1.10) dans (1.9) pour obtenir une expression « locale » dans le sens où toutes les fonctions sont centrées en  $x$

$$\begin{aligned} \sum_{j=1}^n K(x_j - x)c_j + \left( d_1 - \frac{1}{12} \sum_{j=1}^n c_j(x_j - x) \right) + (x_i - x) \left( d_2 + \frac{1}{4} \sum_{j=1}^n c_j(x_j - x)^2 \right) \\ = \\ u_i - \frac{\lambda}{w_i} c_i \end{aligned}$$

On pose alors

$$\begin{cases} d'_1 = d_1 - \frac{1}{12} \sum_{j=1}^n c_j(x_j - x) \\ d'_2 = d_2 + \frac{1}{4} \sum_{j=1}^n c_j(x_j - x)^2 \end{cases}$$

et le système d'optimalité devient

$$\begin{bmatrix} K + \lambda W^{-1} & P \\ P^\top & 0 \end{bmatrix} \begin{Bmatrix} c \\ d' \end{Bmatrix} = \begin{Bmatrix} y \\ 0 \end{Bmatrix} \quad (1.11)$$

avec  $K = [K(x_i - x_j)]_{i,j}$ . Les formules de transformation  $d$  vers  $d'$  nous permettent par ailleurs de calculer  $d$ .

On peut aussi montrer que la fonction que nous construisons avec ce système est (on applique à  $f_W^\lambda$  les mêmes transformations que précédemment) :

$$g_W^\lambda(x) = \sum_{i=1}^n c_i K_i(x) + \sum_{\nu=1}^m d'_\nu \phi_\nu(x)$$

où  $K_i(x) = K(x_i - x)$ , et  $g_W^\lambda$  est telle que

$$\forall x \in [a, b], g_W^\lambda(x) = f_W^\lambda(x)$$

C'est le système de construction du cokrigeage d'erreur de covariance généralisée  $K(h)$ , et de matrice de variance d'erreur  $W^{-1}$ . Nous venons donc de redémontrer le théorème d'équivalence splines–kriégeage dans un cas particulier (en nous inspirant largement des travaux de [Dub81]).

**Théorème 7** *La spline d'ajustement construite par minimisation de  $\mathcal{E}_W^\lambda$  conduit à la même approximation que le cokrigeage d'erreur avec covariance généralisée  $K(h) = |h|^{2m-1}$  et matrice de covariance d'erreur  $W^{-1}$ .*

Ce résultat important montre que l'on aurait pu travailler dans le cadre du kriégeage et aboutir à la même approximation. Cependant le formalisme probabiliste se prête peu à des variations de  $W$  avec le point d'estimation comme nous le ferons dans la suite: cela signifierait qu'une variable aléatoire a plusieurs variances différentes selon son point d'observation (ce qui justifie au passage l'appellation « Approximation Myope ») !

Notons que ce théorème nous donne le système de construction de  $g_W^\lambda(x)$  en fonctions de forme  $N_i$  utilisé habituellement en kriégeage

$$\begin{bmatrix} K + \lambda W^{-1} & P(x) \\ P(x)^\top & 0 \end{bmatrix} \begin{Bmatrix} N(x) \\ \mu(x) \end{Bmatrix} = \begin{Bmatrix} K(x) \\ e^1 \end{Bmatrix} \quad (1.12)$$

avec  $K(x)$  le vecteur des  $K(x_i - x)$ , et  $P(x)$  la matrice des monômes centrés en  $x$  et évalué aux  $x_i$ .

Ce développement, ainsi que le théorème précédent appellent deux commentaires. En premier lieu, il nous a permis de centrer le système (1.7) en un point  $x$

arbitraire, ce qui aura une grande importance dans la suite. En second lieu, nous avons maintenant un système qui nous permet de calculer une fonction  $g_W^\lambda$  qui ne respecte plus la structure de  $\mathcal{H}^m$  dans le sens où la combinaison linéaire de  $K_i$  n'est pas dans  $\mathcal{H}_2^m$  (c'est pourquoi les coefficients  $d$  sont transformés en  $d'$ ).

Nous allons donc proposer un système de calcul de  $c$  et  $d$  directement en nous inspirant du calcul précédent. Pour cela nous transformons de nouveau (1.10) en utilisant les conditions d'orthogonalité

$$\begin{aligned} \sum_{j=1}^n c_j \xi(x_i) = & \sum_{j=1}^n K(x_j - x) c_j + \frac{1}{4}(x_i - x) \sum_{j=1}^n (x_j - x)^2 c_j - \frac{1}{12} \sum_{j=1}^n (x_j - x)^3 c_j \\ & + \underbrace{\frac{1}{4}(x_i - x)^2 \sum_{j=1}^n c_j (x_j - x)}_0 + \underbrace{\frac{1}{12}(x_i - x)^3 \sum_{j=1}^n c_j}_0 \end{aligned}$$

ce qui s'écrit aussi en factorisant en  $c_j$ ,

$$\begin{aligned} \sum_{j=1}^n c_j \xi(x_i) = & \sum_{j=1}^n c_j \left( \frac{Y(x_j - x_i)}{6} (x_j - x_i)^3 - \frac{1}{6} (x_j - x)^3 + \frac{1}{2} (x_j - x)^2 (x_i - x) \right) \end{aligned}$$

ce qui revient à remplacer  $a$  par  $x$  dans la formule définissant  $\xi_i$ . Ceci n'implique cependant pas que  $\xi_i$  admet la forme

$$\frac{Y(x_i - y)}{6} (x_i - y)^3 - \frac{1}{6} (x_i - x)^3 + \frac{1}{2} (x_i - x)^2 (y - x)$$

car nous avons utilisé les conditions d'orthogonalité  $P^\top d = 0$ . Nous définissons donc les fonctions

$$\zeta_i^x(y) = \frac{Y(x_i - y)}{6} (x_i - y)^3 - \frac{1}{6} (x_i - x)^3 + \frac{1}{2} (x_i - x)^2 (y - x) \quad (1.13)$$

et  $\tilde{G}(x)$  la matrice des  $\zeta_i^x(x_j)$ , et il vient la propriété suivante.

**Propriété 22** *Le système*

$$\begin{bmatrix} \tilde{G}(x) + \lambda W^{-1} & P(x) \\ P^\top(x) & 0 \end{bmatrix} \begin{Bmatrix} c \\ d \end{Bmatrix} = \begin{Bmatrix} y \\ 0 \end{Bmatrix} \quad (1.14)$$

admet alors les mêmes solutions que le système d'optimalité (1.7).

## 1.5 Approximation Myope

Les développements précédents sont des résultats bien connus que nous avons revisités et analysés avec notre sensibilité (la propriété 22 est originale). Nous passons à présent au calcul de l'approximation avec fenêtre glissante qui est notre but.

### 1.5.1 Définitions

Nous reprenons maintenant l'idée qui est à la base de la construction des approximations glissantes : le vecteur  $\{w_i\}_i$  est maintenant une fonction de  $x$ . Nous

étendons ici le travail de Salkauskas [Sal92]. Nous utiliserons comme nous en avons l'habitude dans le cadre diffus la notation suivante

$$I(x) = \{i \in \llbracket 1, n \rrbracket, w_i(x) \neq 0\}$$

On fait bien entendu l'hypothèse  $\text{Card}(I(x)) > m$  pour tout  $x$  de l'espace d'intérêt (voir la discussion relative au système d'optimalité).

Nous sommes donc amenés à minimiser le critère

$$\mathcal{E}_x^\lambda(f) = \sum_{i \in I(x)} w_i(x)(y_i - f(x_i))^2 + \int_a^b (f^{(m)})^2$$

dont le minimum est atteint pour la fonction  $f_x^\lambda$  de la forme

$$f_x^\lambda(y) = \sum_{i \in I(x)} c_i(x)K(y - x_i) + \sum_{\nu=0}^{m-1} d_\nu(x)\phi_\nu^x(y)$$

avec les coefficients  $c_i(x)$  et  $d_\nu$  calculés comme dans la partie précédente; et où les conditions de consistence sont centrés en  $x$ :  $\phi_\nu^x(y) = \frac{(y-x)^\nu}{\nu!}$  pour  $\nu \in \llbracket 0, m-1 \rrbracket$ . Nous noterons dans la suite  $K_i(y) = K(y - x_i)$ .

**Définition 17** *L'approximation myope en  $x$  construite à partir des nœuds  $x_i$  est définie par*

$$\tilde{u}(x) = f_x^\lambda(x) \quad (1.15)$$

Nous construisons maintenant une pseudo-dérivée en suivant la même idée que pour l'approximation diffuse.

**Définition 18** *La pseudo-dérivée de l'approximation myope en  $x$  est définie par*

$$\frac{\delta \tilde{u}}{\delta x}(x) = \left. \frac{d}{dy} f_x^\lambda \right|_{y=x} \quad (1.16)$$

En pratique cela signifie que

$$\frac{\delta \tilde{u}}{\delta x} = \sum_{i \in I(x)} c_i(x)K_i'(x) + \sum_{\nu=0}^{m-1} d_\nu(x)\phi_\nu^{x'}(x)$$

et l'ordre de pseudo dérivation est limité par la régularité de  $K$  (car les polynômes sont indéfiniment dérivables).

### 1.5.2 Propriétés

L'approximation myope possède trois propriétés fondamentales, c'est une approximation locale, consistante et dont la régularité est fixée par la plus faible de celles de  $K$  et des fonctions  $w_i$ .

**Propriété 23** *L'approximation myope est locale, et plus précisément*

$$\text{supp}(N_i) = \text{supp}(w_i)$$

C'est évident, car le point  $x_i$  disparaît du critère  $\mathcal{E}_x^\lambda$  lorsque  $w_i(x) = 0$ , ainsi la valeur nodale  $u_i$  n'intervient plus dans le calcul de  $\tilde{u}$  et donc finalement  $N_i(x) = 0$ . Notons que cette propriété n'est intéressante que si la fonction de forme  $N_i$  est régulière, c'est-à-dire qu'on n'a pas de « saut » lorsque  $N_i$  devient nulle. C'est le sens de la propriété suivante.

**Propriété 24** *L'approximation myope est aussi régulière que la moins régulière des fonctions  $K_i(x)$  et  $w_i(x)$ .*

À liste de nœuds  $I(x)$  fixée la propriété est assurée par le système de construction de l'approximation. On doit cependant analyser avec soin le cas d'un point qui entre ou sort de la liste de nœuds. Pour cela, on écrit le système de construction de l'approximation myope de façon légèrement différente

$$\begin{bmatrix} W(x)K + \lambda Id & W(x)P(x) \\ P^\top(x) & 0 \end{bmatrix} \begin{Bmatrix} N(x) \\ \mu(x) \end{Bmatrix} = \begin{Bmatrix} W(x)K(x) \\ e^1 \end{Bmatrix}$$

et on fait tendre  $w_{i_0}$  vers zéro. La ligne  $i_0$  devient alors continûment  $N_{i_0} = 0$ , ce qui assure la propriété.

Enfin, nous allons analyser la consistance de l'approximation myope.

**Propriété 25** *L'approximation myope, ainsi que sa pseudo-dérivée sont consistantes jusqu'à l'ordre  $m - 1$*

Cela signifie que si les données sont celles d'un polynôme de degré inférieur ou égal à  $m - 1$  alors l'approximation myope le reproduit exactement. Or nous savons que la spline se confond alors avec le polynôme en question, et cela assure la propriété. Le cas de la pseudo-dérivée est assuré par le fait que les coefficients  $c_i$  sont tous nuls, et les  $d_\nu$  sont ceux du polynôme exprimé dans la base  $(\phi_\nu^x)_\nu$  de  $\mathcal{P}_{m-1}$ .

### 1.5.3 Interpolation myope

Nous allons montrer que l'approximation myope est interpolante sous certaines conditions sur les fonctions de pondérations  $w_i$ .

**Propriété 26** *Si la fonction de pondérations  $w_i$  vérifie*

$$\lim_{x \rightarrow x_i} w_i(x) = +\infty$$

*alors l'approximation myope interpole la donnée en  $x_i$ .*

Cette propriété est une conséquence de la propriété 21 concernant les splines d'ajustement.

Reprenons maintenant le critère de construction (1.6) au voisinage du point  $i$  en posant  $\mu = w_i(x)(u_i - g_i^\top c - t_i^\top d)$ , et  $W_i, G_i, P_i$  les matrices pour lesquelles on a supprimé la contribution du nœud  $i$ .

$$\mathcal{E}_x^\lambda(c, d) = [y_i - G_i c - P_i d]^\top W_i [y_i - G_i c - P_i d] + \mu(u_i - g_i^\top c - t_i^\top d) + \lambda c^\top G c$$

où  $g_i$  et  $t_i$  sont les vecteurs colonnes  $i$  des matrices  $G$  et  $P$ . La minimisation conduit alors au système

$$\begin{cases} (G_i^\top W_i G_i + \lambda G) c & + G_i W_i P_i d & + \mu g_i & = G_i^\top W_i y_i \\ P_i^\top W_i G_i c & + P_i^\top W_i P_i d & + \mu t_i & = P_i^\top W_i y_i \\ g_i^\top c & + t_i d & - \frac{1}{w_i} \mu & = u_i \end{cases}$$

qui a la forme

$$\begin{bmatrix} \mathcal{A}(x) & q \\ q^\top & -\frac{1}{w_i} \end{bmatrix} \begin{Bmatrix} \beta \\ \mu \end{Bmatrix} = \begin{Bmatrix} b(x) \\ u_i \end{Bmatrix}$$

Dérivons ce système en  $x$

$$\begin{bmatrix} \mathcal{A}(x) & q \\ q^\top & -\frac{1}{w_i} \end{bmatrix} \begin{Bmatrix} \beta(x)' \\ \mu' \end{Bmatrix} + \begin{bmatrix} \mathcal{A}(x)' & 0 \\ 0 & -(\frac{1}{w_i})' \end{bmatrix} \begin{Bmatrix} \beta \\ \mu \end{Bmatrix} = \begin{Bmatrix} b'(x) \\ 0 \end{Bmatrix}$$

Faisons tendre  $x$  vers  $x_i$  en faisant l'hypothèse supplémentaire

$$\lim_{x \rightarrow x_i} -\left(\frac{1}{w_i(x)}\right)' = 0$$

la dernière ligne du système dérivé montre alors que  $q^\top \beta'(x_i) = 0$ , c'est à dire en revenant aux notations habituelles

$$\sum_{i=1}^n c'(x_i)K_i(x_i) + \sum_{\nu=0}^{m-1} d'(x_i)\phi_\nu^{x_i}(x_i) = 0$$

or cette expression est exactement la différence entre la dérivée réelle et la pseudo-dérivée au nœud  $x_i$ . En remarquant que la construction habituelle des fonctions  $w_i$  assure que l'hypothèse sur sa dérivée est vraie (voir la même propriété pour l'approximation diffuse 23), on a la propriété suivante.

**Propriété 27** *La pseudo-dérivée de l'interpolation myope est égale à sa dérivée réelle aux nœuds d'interpolation.*

#### 1.5.4 Approximation Myope et Approximation Diffuse

La construction même de l'approximateur myope montre son lien aux interpolateurs radiaux (splines, krigeage et radial basis functions). Nous allons démontrer que l'interpolation myope admet l'interpolation diffuse comme cas limite lorsque  $\lambda$  tend vers l'infini.

**Propriété 28** *Lorsque  $\lambda \rightarrow \infty$ , l'interpolateur myope tend vers l'interpolation diffuse.*

Cette propriété peut se démontrer en faisant un développement limité en  $\lambda$  dans le système (1.8). On note  $\alpha$  la solution du système de construction de l'interpolation diffuse  $P^\top W P \alpha = P^\top W u$ . On a pour commencer ( $\epsilon$  désigne de façon générique une fonction de limite nulle lorsque  $\lambda \rightarrow \infty$ ).

$$\begin{aligned} S &= \lambda W^{-1} (Id + \frac{1}{\lambda} W G) \\ S^{-1} &= \frac{1}{\lambda} (Id - \frac{1}{\lambda} W G + \frac{1}{\lambda} \epsilon(\lambda)) W \\ &= \frac{1}{\lambda} W - \frac{1}{\lambda^2} W G W + \frac{1}{\lambda^2} \epsilon(\lambda) \end{aligned}$$

et la seconde équation de (1.8) se transforme alors en

$$\begin{aligned} d &= \left(\frac{1}{\lambda} (P^\top W P - \frac{1}{\lambda} P^\top W G W P + \frac{1}{\lambda} \epsilon(\lambda))\right)^{-1} P^\top \left(\frac{1}{\lambda} (W - \frac{1}{\lambda} W G W + \frac{1}{\lambda} \epsilon(\lambda))\right) u \\ &= \left((P^\top W P)^{-1} + \epsilon(\lambda)\right) (P^\top W + \epsilon(\lambda)) u \\ &= (P^\top W P)^{-1} P^\top W u + \epsilon(\lambda) \\ &= \alpha + \epsilon(\lambda) \end{aligned}$$

De la même façon, la première équation se transforme en

$$c = \frac{1}{\lambda} (W - W P (P^\top W P)^{-1} P^\top W) u + \frac{1}{\lambda} \epsilon(\lambda)$$

Finalement, on montre que la limite de  $(d, c)$  lorsque  $\lambda$  tend vers l'infini est  $(\alpha, 0)$ , ce qui assure la propriété.

#### 1.5.5 Approximation diffuse pénalisée

Nous avons construit une approximation glissante avec un terme supplémentaire de régulation de la dérivée seconde (d'ordre  $m$ ) de  $f_x^\lambda$ . La pénalisation porte cependant sur le comportement global de l'approximation. Nous pouvons imaginer une



pénalisation plus « locale », à partir de l'approximation diffuse en minimisant le critère discret

$$\mathcal{E}_x^\lambda(\{\alpha\}) = \sum_{i \in I(x)} w_i(x) (u_i - \langle p(x_i) \rangle \{\alpha\})^2 + \lambda \alpha_m^2 \quad (1.17)$$

où

- $\langle p(x_i) \rangle$  est le vecteur ligne des monômes centrés en  $x$  jusqu'au degré  $m$
- l'approximation est  $\tilde{u}(x) = \langle p(x) \rangle \{a(x)\}$ , et
- $\{\alpha(x)\}$  le vecteur  $\alpha$  minimisant  $\mathcal{E}_x^\lambda$ .

Il est assez aisé de calculer le système d'optimalité, qui prend la forme

$$\mathcal{A}(x)a(x) = b(x)$$

avec

- $\mathcal{A}(x) = P^\top W(x)P + \lambda Q$ ,
- $W(x)$  est la matrice diagonale des  $w_i(x)$ ,
- $P(x) = [p_j(x_i)]_{i \in I(x), j \in [0, m]}$  est la matrice des monômes centrés en  $x$  évalués aux points  $x_i$ ,
- la matrice  $Q$  est définie par ( $m^{\text{ième}}$  élément de la diagonale non nul)

$$Q = \begin{bmatrix} 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots \\ 0 & \cdots & 1 & \cdots & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 \end{bmatrix}$$

- $b(x) = P(x)^\top W(x)U$  où  $U$  est le vecteur des  $u_i$ .

**Propriété 29** *Les fonctions de forme de l'approximation diffuse pénalisée sont solutions du système*

$$\begin{bmatrix} W^{-1}(x) & P(x) \\ P^\top(x) & -\lambda Q \end{bmatrix} \begin{Bmatrix} N \\ \mu \end{Bmatrix} = \begin{Bmatrix} 0 \\ e^1 \end{Bmatrix} \quad (1.18)$$

En effet, le système 1.18 est équivalent à

$$\begin{cases} N = -W(x)P(x)\mu \\ P^\top(x)N - \lambda Q\mu = e_1 \end{cases}$$

ou

$$\begin{cases} \mu = -(P^\top(x)W(x)P(x) + \lambda Q)^{-1} e_1 \\ N = W(x)P(x)(P^\top(x)W(x)P(x) + \lambda Q)^{-1} e_1 \end{cases}$$

et finalement (en notant  $p(t)$  le vecteur  $\{\phi_\nu^x(t)\}$ )

$$\begin{aligned} N_i(x) &= e_i^\top W(x)P^\top(x)\mathcal{A}^{-1}(x)e_1 \\ &= e_1^\top \mathcal{A}^{-1}(x)w_i(x)p(x_i) \end{aligned}$$

L'approximation  $\tilde{u}$  s'écrit alors

$$\begin{aligned}\tilde{u}(x) &= \sum_{i \in I(x)} N_i(x) u_i \\ &= e_1^\top \sum_{i \in I(x)} \mathcal{A}^{-1}(x) w_i(x) p(x_i) u_i \\ &= p(x)^\top (\mathcal{A}^{-1}(x) W(x) P(x) U)\end{aligned}$$

et ceci termine la démonstration. En conséquence, nous pouvons énoncer le théorème de caractérisation des fonctions de forme diffuse pénalisée.

**Théorème 8** *Les fonctions de forme diffuses pénalisées sont solutions du problème de minimisation sous contraintes*

$$\left\{ \begin{array}{l} \min \left\{ \sum_{i=1}^n \frac{N_i^2}{w_i} + \lambda M^2 + \sum_{i=1}^n M N_i \phi_2^x(x_i) \right\} \\ P_m(x)^\top N = e_1 \end{array} \right\}$$

où

- $M$  est une inconnue auxiliaire.
- $P_m(x)$  est la matrice  $P(x)$  sans le degré  $m$ .

## 1.6 Expérimentation numérique

Nous avons testé l'interpolateur myope numériquement afin de mieux comprendre le rôle et la sensibilité de chaque paramètre. Pour cela, plusieurs tests ont été menés :

- Calcul des fonctions de forme,
- Approximation de la fonction sinus,
- Influence de  $\lambda$ ,

L'ensemble de ces tests porte à la fois six variantes de l'interpolateur myope et de ses limites :

1. Interpolation myope (IM),
2. Interpolation myope avec consistance d'ordre 2 (IM-2)
3. Interpolation diffuse (ID),
4. Interpolation diffuse avec consistance d'ordre 2 (ID-2),
5. Interpolation diffuse pénalisée (ID-p).
6. Interpolation spline-cubique (SC).

Nous pourrions ainsi comparer les comportements de ces différentes méthodes d'interpolation.

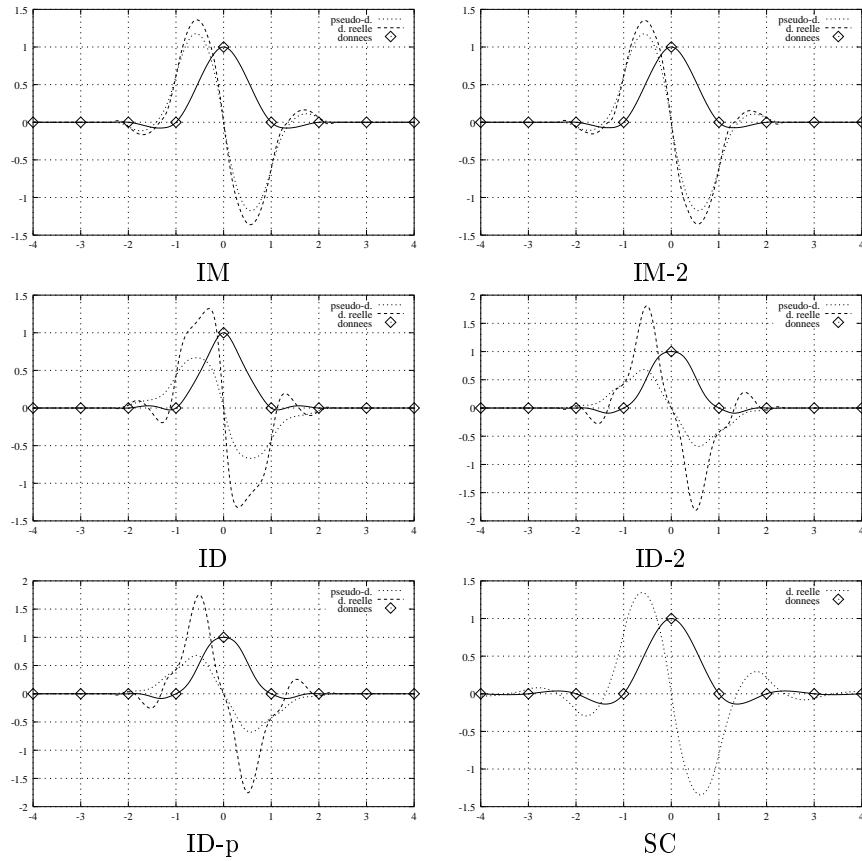


FIG. 1.1: Fonctions de formes pour des points répartis de façon régulière. Pseudo-dérivée (lorsqu'elle est définie) et dérivée « réelle ». Pour l'interpolateur myope, et l'interpolation diffuse pénalisée,  $\lambda = 1$ .

### 1.6.1 Fonctions de forme

Les figures 1.1 et 1.2 représentent les fonctions de forme des six interpolateurs que nous avons testés. Ces courbes mettent en évidence deux différences :

1. La pseudo-dérivée semble plus proche de la dérivée réelle dans le cas de l'approximation myope que dans le cas de l'approximation diffuse.
2. L'approximation myope est plus raide que l'approximation diffuse.

Le support des fonctions de forme est bien celui que nous attendions (ie: celui de  $w_i(\cdot)$ ).

Les variations de la dérivée réelle sont beaucoup plus marquées avec une consistance quadratique. On remarque aussi que le terme de pénalisation limite bien ces variations (ID-p) comme nous l'espérons.

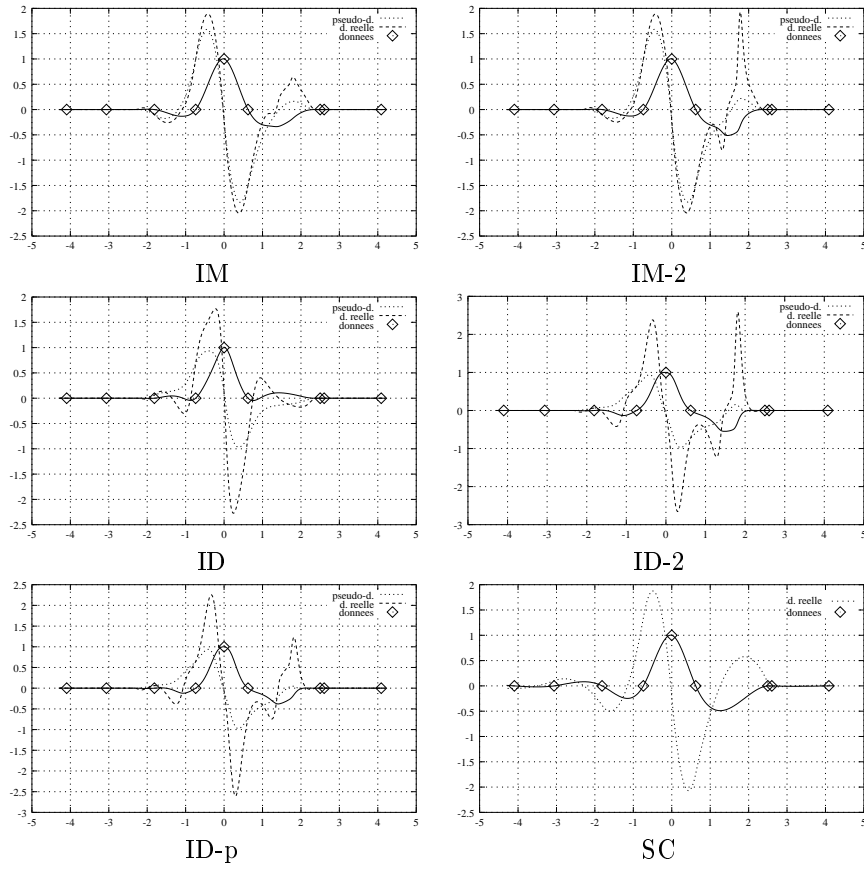


FIG. 1.2: Fonctions de forme pour des points d'abscisses perturbées aléatoirement.  $\lambda = 1$ .

### 1.6.2 Fonction sinus

L'approximation de la fonction sinus confirme nos observations précédentes (figures 1.3 et 1.4) :

- L'interpolateur myope semble plus raide que l'interpolation diffuse ;
- La pseudo-dérivée est plus proche de la dérivée réelle dans le cas de l'interpolateur myope ;
- Le terme de pénalisation lisse bien les dérivées réelle et diffuse.

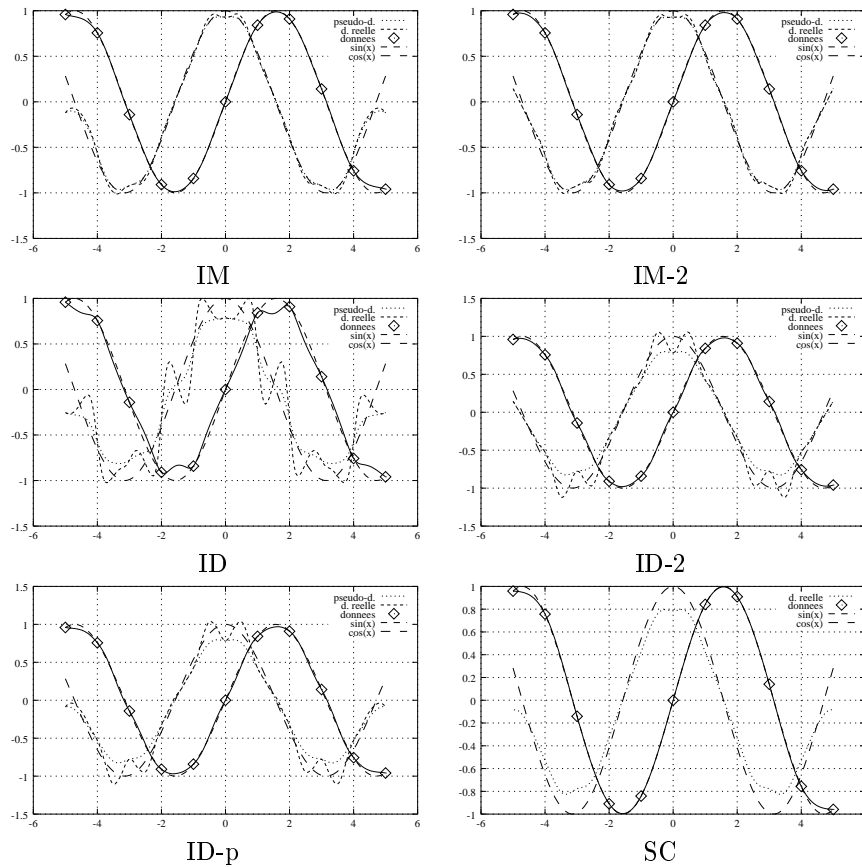


FIG. 1.3: Approximation de la fonction sinus échantillonnée aux points de coordonnées entières.  $\lambda = 1$ .

On note par ailleurs que la géométrie joue un rôle essentiel dans le comportement de l'approximation. En effet, la présence de deux nœuds très proches perturbe les différentes approximations. Cette perturbation est surtout sensible sur le tracé de la dérivée réelle. Même la spline cubique qui minimise la courbure

$$\int_a^b (f'')^2$$

dans  $C^2([a, b])$  est perturbée au voisinage de ces deux nœuds. Nous allons retrouver l'importance de la géométrie dans la suite.

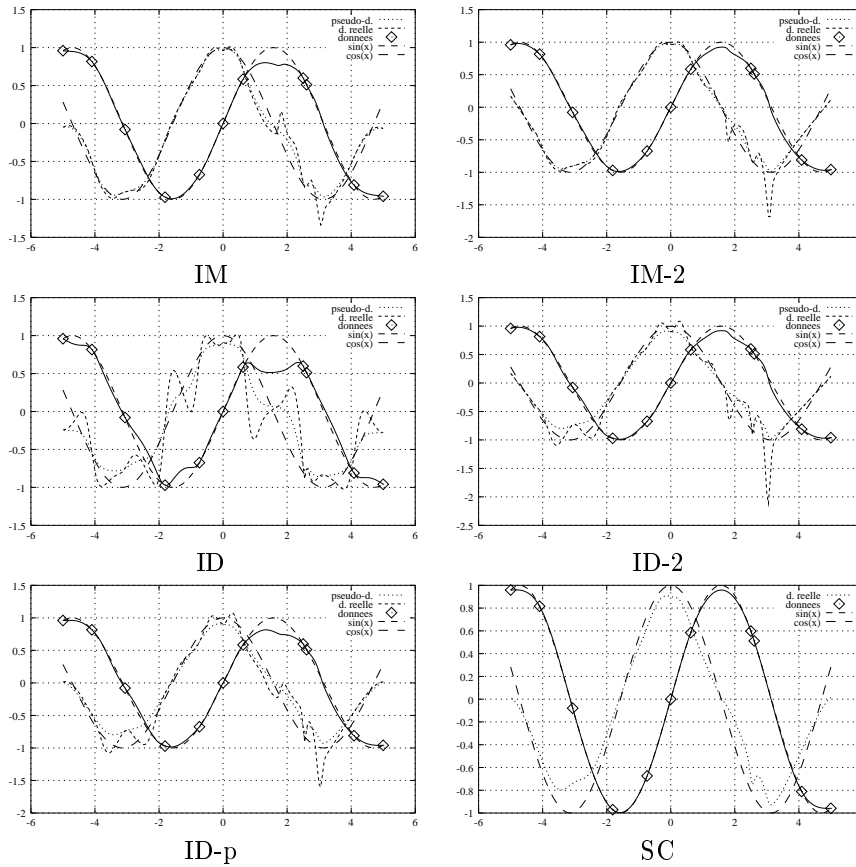


FIG. 1.4: Approximation de la fonction sinus échantillonnée en des points aléatoires.  $\lambda = 1$ .

### 1.6.3 Influence de $\lambda$

Nous avons vu dans le développement théorique que l'interpolateur myope tend vers l'interpolation diffuse lorsque  $\lambda$  tend vers l'infini. La figure 1.5 montre l'erreur relative pour  $\lambda = 10^6$  : On retrouve bien numériquement le comportement attendu.

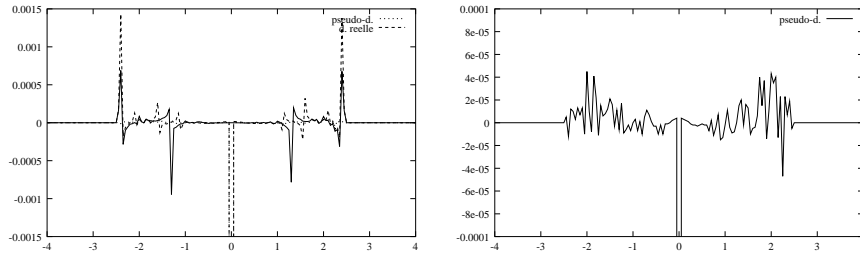


FIG. 1.5: Erreur relative entre l'interpolation diffuse et l'interpolateur myope pour  $\lambda = 10^6$ .

Lorsque  $\lambda$  tend vers 0, on observe des zones de très fortes variations (figure 1.7). La figure 1.6 montre qu'il ne s'agit pas de discontinuité conformément à la théorie. Cette zone de fortes variations correspond en fait à l'entrée d'un point de donnée dans le calcul de l'approximation. Notons que l'entrée et la sortie des points de données coïncident (la liste des nœuds d'appuit passe de  $\{1, 2, 3, 4\}$  à  $\{2, 3, 4, 5\}$ ) mais cela n'est pas la cause du décrochement. On a en fait des fonctions de forme splines qui se raccordent sur de petites zones grâce aux fonctions  $w_i$  qui n'ont qu'une très faible importance car  $\lambda$  est trop faible. Ainsi ce phénomène peut être atténué par deux facteurs :

- une augmentation de la valeur de  $\lambda$ ,
- la taille des zones d'influence des nœuds  $V_i$  : plus cette taille est grande, plus la valeur de la fonction de forme spline est faible, et le raccord est plus aisé.

Ce comportement des fonctions de forme myope est gênant, car l'écart entre dérivée réelle et pseudo dérivée est très faible pour les petites valeurs de  $\lambda$ , ce qui aurait pu être utile pour le calcul d'une intégration robuste pour les équations différentielles (voir la quatrième partie). Comme on cherche à limiter les supports de fonctions de forme pour réduire la largeur de bande de la matrice de raideur, aucune de ces deux solutions ne sont satisfaisantes.

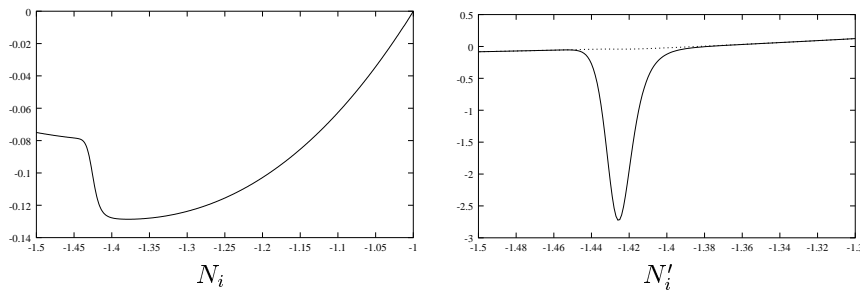
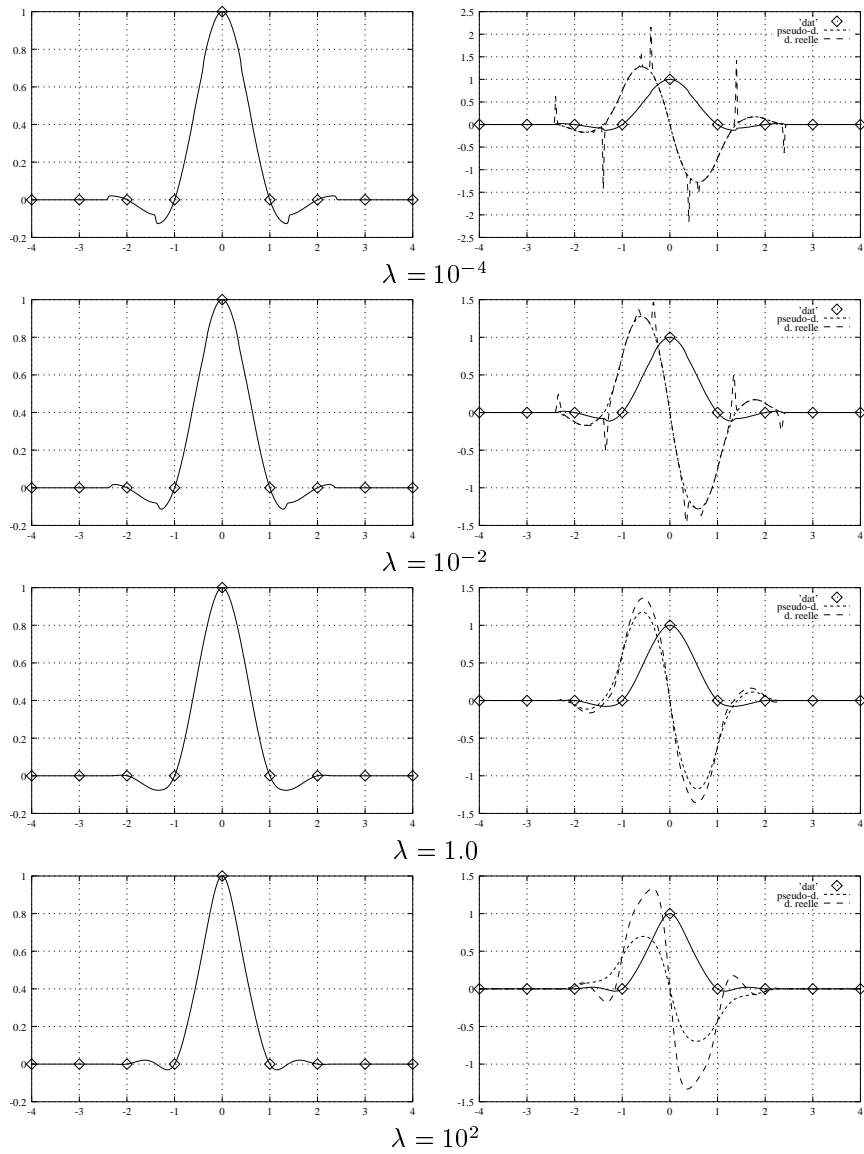


FIG. 1.6: Détail montrant l'entrée d'un nœud dans le calcul de la fonction de forme. Le point de frontière  $x_o$  est en  $-1.5$ , le retard s'explique par l'estimation numérique de  $w_i$  qui ne devient non nulle qu'à partir de  $-1.45$ .

FIG. 1.7: Évolution des fonctions de forme de l'interpolateur myope avec  $\lambda$ .



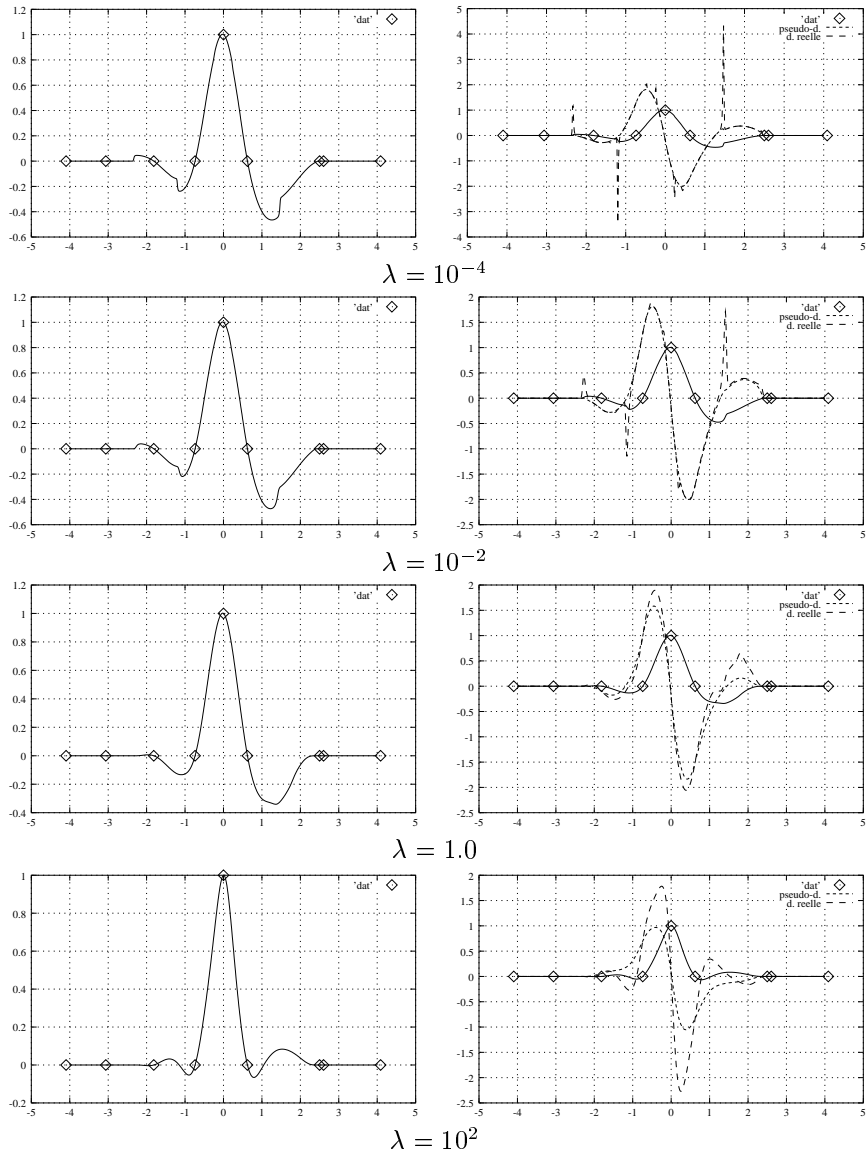


FIG. 1.8: Évolution des fonctions de forme de l'interpolateur myope avec  $\lambda$  (position des nœuds pseudo-aléatoire).

## 1.7 Conclusion

En conclusion, nous avons dans cette partie proposé une étude approfondie de la structure de  $\mathcal{H}^m$  qui permet de construire les splines plaques minces. La construction des splines cubiques est développée et une propriété originale (système centré) démontrée. Ceci nous permet de construire l'approximation myope et de démontrer ses principales propriétés : consistante, interpolation, fonctions de formes, selon le schéma adopté pour la présentation de l'approximation diffuse. Nous montrons aussi que l'approximation myope est une notion intermédiaire entre splines et approximation diffuse (propriétés 28 et  $w_i$  constants).

Ainsi, la construction de l'approximation myope montre que l'approximation diffuse et les interpolateurs radiaux sont différents. La structure particulière de  $\mathcal{H}^m$  que nous avons construite pour les splines est un modèle : il est bien connu que  $\int (f'')^2$  est une approximation de l'énergie de flexion d'une poutre et la structure est adaptée à ce critère. La forme de la fonction  $\tilde{u}$  découle de la structure, alors que l'approximation diffuse pose *a priori*  $\tilde{u}(x) = \sum a_l(x) f^l(x)$ .

L'approximation myope est une approximation locale et consistante, nous l'avons donc utilisée pour résoudre des équations différentielles (avec la pseudo-dérivation et une intégration numérique adaptée [SVS99]). Les résultats ne montrent pas de réelle amélioration vis-à-vis de l'approximation diffuse. Cependant, la pseudo-dérivée des fonctions de forme myopes est de meilleure qualité que celle de l'approximation diffuse. Cette propriété pourra être exploitée pour la construction de l'approximation dans les cas difficiles (voir le chapitre 2 de la partie IV). Les fonctions de forme myopes restent plus difficiles à calculer que les fonctions de formes diffuses, nous ne les utiliserons donc pas dans le cadre de ce mémoire.

Cependant, l'approximation myope contient un « certain » modèle (on ne le connaît pas car le modèle apparaît à travers la diagonalisation  $\tilde{u}(x) = f_x^\lambda(x)$ ) qui pourrait être utile pour des problèmes d'approximation. Le krigeage en voisinage glissant (on tronque les fonctions de forme au delà d'une certaine distance) peut aussi être envisagé différemment grâce à l'approximation myope. En conséquence, les courbes de niveau ne sont pas continues : on n'aurait pas ce désavantage avec des fonctions d'influence de type plateau.



## Chapitre 2

# Approximation Hermite

Dans leurs formulations classiques, l'approximation diffuse, les interpolateurs radiaux ou l'approximation myope sont des méthodes d'approximation de type Lagrangien. Il est parfois utile de savoir aussi interpoler des données différentielles. Nous avons vu dans l'introduction que l'interpolation de Hermite le permet. Nous nous proposons donc d'étudier une variante de type Hermite de l'approximation diffuse.

L'approximation de type Hermite a été proposée dans [LDT96] pour la prise en compte de conditions aux limites de type Neumann ; l'article [ACK99] en propose aussi une description. Liu *et al.* construisent une méthode RKPM de type Hermite dans [LCUC96]. Cependant, les auteurs n'étudient pas les propriétés de cette technique. Nous nous proposons donc d'étudier l'approximation diffuse de type Hermite en suivant le schéma habituel : construction, existence, propriétés de consistance et construction en fonction de forme, et enfin un théorème de convergence. Comme précédemment, l'étude se place dans le cadre monodimensionnel sans que cela nuise à la généralité des résultats.

Nous vérifions les taux de convergence numériquement et nous traçons des fonctions de forme sur quelques cas. Un exemple montre que l'interpolation obtenue est lisse et de bonne qualité visuelle, cependant on remarque un écart important entre dérivée « réelle » et dérivée diffuse. L'approximation diffuse de type Hermite sera le cœur du calcul de courbure des surfaces triangulées et elle sera aussi utilisée pour le calcul de poutre comme dans [ACK99] mais avec une formulation différente.

## 2.1 Construction et propriétés

### 2.1.1 Construction de l'approximation

Soit  $(x_i)_{i \in [1, n]}$  et  $(y_j)_{j \in [1, r]}$  deux ensembles de points auxquels sont attachées des données  $(u_i)_{i \in [1, n]}$  et  $(v_j)_{j \in [1, r]}$  valeurs et dérivées d'une grandeur à approximer sur un ouvert  $\Omega$ . On suppose que les points  $x_i$  sont distincts deux à deux ainsi que les  $y_j$  ; ces deux ensembles peuvent être confondus.

Nous attachons à chaque point de donnée une fonction  $w_i$  (ou  $w_j$ ) et on définit  $\bar{u}(x) = \sum_{k=0}^m a_k \frac{x^k}{k!} = \langle p(x) \rangle \{a\}$ . La construction de l'approximation  $\tilde{u}$  passe par la minimisation du critère quadratique ( $I(x)$  et  $J(x)$  ont le sens habituel)

$$\mathcal{E}_x(\{a\}) = \sum_{i \in I(x)} w_i(x) (u_i - \bar{u}(x_i))^2 + \lambda \sum_{j \in J(x)} r_j^2 w_j(x) (v_j - \bar{u}'(y_j))^2 \quad (2.1)$$

et on définit l'approximation diffuse par  $\tilde{u}(x) = \langle p(x) \rangle \{a(x)\}$  avec  $\{a(x)\}$  la valeur du vecteur  $\{a\}$  au point  $x$ .

**Remarque 6**  $\lambda$  est sans dimension physique (unité), et  $r_j$  a la dimension de l'espace géométrique.

Ceci est nécessaire pour assurer l'homogénéité de  $\mathcal{E}_x$ . Les paramètres  $r_j$  seront spécifiés dans l'étude de convergence où ils permettent d'assurer l'uniformité de la base des fonctions de forme [SF73]. Posons  $n_I = \text{Card}(I(x))$ ,  $n_J = \text{Card}(J(x))$ ,

$$P(x) = \begin{bmatrix} \left[ \begin{array}{cccc} 1 & x_1 & \dots & \frac{x_i^m}{m!} \end{array} \right]_{i \in I(x)} \\ \left[ \begin{array}{cccc} 0 & 1 & \dots & \frac{y_j^{m-1}}{(m-1)!} \end{array} \right]_{j \in J(x)} \end{bmatrix}, \quad U(x) = \begin{Bmatrix} \{u_i\}_{i \in I(x)} \\ \{v_j\}_{j \in J(x)} \end{Bmatrix}$$

$$W(x) = \begin{bmatrix} \text{diag}_{i \in I(x)} \{w_i(x)\} & 0 \\ 0 & \text{diag}_{j \in J(x)} \{r_j^2 w_j(x)\} \end{bmatrix}$$

Le système d'optimalité caractérisant  $a$  est

$$(P^\top(x)W(x)P(x))\{a(x)\} = A(x)\{a(x)\} = P^\top(x)W(x)U(x) \quad (2.2)$$

Il vient alors

$$\tilde{u}(x) = \langle p(x) \rangle A^{-1}(x)P^\top(x)W(x)U(x) \quad (2.3)$$

**Propriété 30** Conditions d'inversibilité de  $A(x)$ .

1. La matrice  $A$  est inversible si :

- $n_I \geq m + 1$ .
- $n_J \geq m$  et  $n_I \geq 1$ .
- Les semis  $x_i, y_j$  sont confondus et  $n_I + n_J \geq m + 1$ .

2. Conditions nécessaires d'inversibilité de  $A(x)$

- $n_I \geq 1$
- $n_I + n_J \geq m + 1$

On démontre cette propriété par contraposée: on cherche un polynôme  $p$  de degré  $m$  qui vérifie  $p(x_i) = 0, \forall i \in I(x)$  et  $p'(x_j) = 0, \forall j \in J(x)$  et qui soit non nul. Si ce polynôme existe alors  $A(x)$  est non inversible. Cas des conditions suffisantes :

- Un polynôme de degré  $m$  s'annulant au moins  $m + 1$  fois est nul.
- Si  $p'$  s'annule  $m$  fois, alors  $p' = 0$  et  $p$  est constant donc nul car  $n_I > 0$ .
- Le cas des semis  $x_i, y_j$  confondu et le cadre classique du théorème d'Hermite (voir par exemple [LS86]).

Les deux conditions nécessaires sont naturelles: si  $n_I$  est nul, alors  $p$  peut être constant non nul; si  $n_I + n_J \leq m$ , alors on a moins d'équations que de paramètres. L'étude du cas  $1 \leq n_I \leq m$  et  $n_J \leq m - 1$  est plus complexe: si le polynôme  $p$  s'annule au points  $x_i$ , alors  $p'$  s'annule nécessairement dans les segments successifs formés par les  $x_i$ . La condition dépend alors des positions relatives des points  $x_i$  et  $y_j$ :

- Exemple où  $A(x)$  est inversible: si le nombre de points  $y_j$  à l'extérieur de  $[\min(x_i), \max(x_i)]$ ,  $n_J^E$  est tel que  $n_I + n_J^E \geq m + 1$ , alors  $p = 0$ .

– Exemple où  $A(x)$  est singulière: le polynôme  $p$  s'écrit  $p = \omega q$  avec

$$\omega(y) = \prod_{i \in I(x)} (y - x_i)$$

et  $q$  polynôme de degré  $m - n_I$ . Soit  $n_J^I = \text{Card}(\{j \in J(x), \omega'(y_j) = 0\})$ , si  $n_J^I = n_J \geq m - n_I$ , alors  $A(x)$  est singulière. En effet, on a  $p' = \omega q' + q\omega'$  d'où  $q(y_j) = 0, \forall j \in J(x)$  est  $p$  s'écrit  $c\omega$ ,  $c \in \mathbb{R}$  (exemple:  $m = 2$ ,  $n_I = 2$ ,  $n_J = 1$  et  $y_1 = \frac{x_1+x_2}{2}$ ).

Dans la suite, nous admettons que les conditions de régularité de  $A(x)$  sont satisfaites.

**Propriété 31** *L'approximation diffuse Hermite est consistante d'ordre  $m$  :*

*s'il existe  $p$  dans  $\mathcal{P}_m$  tel que pour tout  $i, j$ ,  $p(x_i) = u_i$ ,  $p'(y_j) = v_j$ , alors  $\tilde{u} = p$ .*

Tout polynôme de degré inférieur ou égale à  $m$  annule le critère de construction et ceci démontre la propriété.

### 2.1.2 Fonctions de forme

L'expression de  $\tilde{u}$  montre que l'approximation diffuse de type Hermite est une fonction linéaire des données  $u_i$  et  $v_j$ . On peut donc écrire

$$\begin{aligned} \tilde{u}(x) &= \sum_{i \in I(x)} N_i(x)u_i + \sum_{j \in J(x)} M_j(x)v_j \\ &= \mathcal{N}^T(x)U(x) \end{aligned}$$

et ceci définit les fonctions de formes  $N_i, M_j$  de l'approximation diffuse Hermite des données  $u_i, v_j$ .

**Propriété 32**

$$\forall p \in \mathcal{P}_m, \forall x \in \Omega, \sum_{i \in I(x)} N_i(x)p(x_i) + \sum_{j \in J(x)} M_j(x)p'(y_j) = p(x)$$

Cette propriété est l'énoncé de la consistance selon les fonctions de forme de l'approximation diffuse Hermite. L'approximation Hermite de Liu *et. al.* s'appuie sur les mêmes conditions de consistance mais dans une formulation continue pour construire deux noyaux associés aux deux types de données selon la technique des fonctions de correction.

Nous pouvons maintenant énoncer le théorème de caractérisation des fonctions de forme diffuse Hermite

**Théorème 9** *Les fonctions de forme  $N_i, M_j$  de l'approximation diffuse minimisent le critère*

$$\mathcal{E}_x^c(N, M) = \sum_{i \in I(x)} \frac{N_i^2}{w_i} + \sum_{j \in J(x)} \frac{M_j^2}{r_j^2 w_j}$$

*sous les contraintes*

$$\forall k \in \llbracket 0, m \rrbracket, \forall x \in \Omega, \sum_{i \in I(x)} N_i(x)x_i^k + \sum_{j \in J(x)} M_j(x)ky_j^{k-1} = x^k$$

Introduisons les multiplicateurs de Lagrange  $\mu$  pour prendre en compte les contraintes et le système caractérisant les fonctions de forme au point  $x$  est (avec les notations précédentes)

$$\begin{bmatrix} W^{-1}(x) & P(x) \\ P(x)^\top & 0 \end{bmatrix} \begin{Bmatrix} \mathcal{N}(x) \\ -\mu(x) \end{Bmatrix} = \begin{Bmatrix} 0 \\ p(x) \end{Bmatrix} \quad (2.4)$$

Ce système se résout en écrivant

$$\begin{cases} \mathcal{N}(x) & = & W(x)P(x)\mu(x) \\ A(x)\mu(x) & = & p(x) \end{cases}$$

et finalement

$$\begin{cases} \mathcal{N}(x) & = & W(x)P(x)A^{-1}(x)p(x) \\ \mu(x) & = & A^{-1}(x)p(x) \end{cases}$$

Cependant la formule (2.3) montre par identification que

$$\mathcal{N}^\top(x) = \langle p(x) \rangle A^{-1}(x)P^\top(x)W(x)$$

Comme les matrices  $A^{-1}(x)$  et  $W(x)$  sont symétriques les fonctions de formes sont bien caractérisées par le théorème 9.

**Remarque 7** *Les conditions de consistances peuvent toujours être centrées au point d'évaluation.*

Les pseudo-dérivées de l'approximation Hermite se définissent de la même façon que pour l'approximation diffuse usuelle. Il existe de même une caractérisation optimale des pseudo-dérivées des fonctions de formes  $N_i, M_j$ ; on pourra consulter à ce sujet [Guy95a] sur le cas Lagrangien.

### 2.1.3 Interpolation

Nous savons que la condition  $\lim_{x \rightarrow x_i} w_i(x) = +\infty$  est suffisante pour assurer l'interpolation de la donnée  $u_i$ . On retrouve bien entendu la même propriété pour les données différentielles  $v_j$ . Nous allons nous pencher ici sur le cas de points de données  $x_{i_0}$  et  $y_{j_0}$  confondus.

En suivant la technique de Guyon [Guy95a], nous introduisons les inconnues supplémentaires

$$\begin{cases} \mu(x) & = & w_{i_0}(x)(\bar{u}(x_{i_0}) - u_{i_0}) \\ \nu(x) & = & w_{j_0}(x)(\bar{u}'(y_{j_0}) - v_{j_0}) \end{cases}$$

et le système de construction de l'approximation Hermite devient

$$\begin{bmatrix} A_0(x) & p(x_{i_0}) & p'(y_{j_0}) \\ p^\top(x_{i_0}) & \frac{-1}{w_{i_0}(x)} & 0 \\ p'^\top(y_{j_0}) & 0 & \frac{-1}{w_{j_0}(x)} \end{bmatrix} \begin{Bmatrix} a \\ \mu \\ \nu \end{Bmatrix} = \begin{Bmatrix} b_0(x) \\ u_{i_0} \\ v_{j_0} \end{Bmatrix} \quad (2.5)$$

où l'indice 0 signifie que l'on supprime la contribution des nœuds  $i_0$  et  $j_0$ . A partir de ce système, on peut transposer les propriétés usuelles de l'interpolation diffuse à l'interpolation diffuse Hermite. En particulier, les pseudo-dérivées du premier et du second ordre sont égales aux dérivées réelles aux nœuds où l'on a à la fois une donnée «  $u$  » et une donnée «  $v$  ».

## 2.2 Convergence

Nous allons étudier la convergence de l'approximation diffuse Hermite en nous appuyant sur sa consistance. Il existe en effet des résultats généraux pour les opérateurs linéaires continus de consistance polynomiale donnée (voir par exemple [RT98]). Ce sont ces mêmes idées qui sont utilisées dans [Vil91, DO96] (voir le chapitre 2, première partie). On suppose que les points de données sont dans la boule de centre  $x$  et de rayon  $\rho$  et on définit les coordonnées adimensionnelles :

$$\begin{cases} \xi_i = \frac{x_i - x}{\rho} \\ \eta_j = \frac{y_j - x}{\rho} \end{cases}$$

**Transformation de  $\mathcal{E}_x$**  Introduisons pour commencer le critère de construction centré

$$\mathcal{E}_x^c(\{\alpha\}) = \sum_{i \in I(x)} w_i(x) (u_i - \bar{u}_x^c(x_i))^2 + \sum_{j \in J(x)} r_j^2 w_j(x) (v_j - \bar{u}_x^c(y_j))^2 \quad (2.6)$$

avec  $\bar{u}_x^c(t) = \langle p(t - x) \rangle \{\alpha\}$ .

**Propriété 33** *Il est équivalent de construire l'approximation diffuse Hermite en formulation centrée (2.6) ou non centrée (2.1).*

Cette propriété est une conséquence de la remarque 7 et du théorème de caractérisation des fonctions de forme.

Supposons que les données sont issues d'une fonction  $u$  dans  $\mathcal{H}^{m+1}(\Omega)$ ,  $u$  admet un développement de Taylor en  $x$  et pour tout point de donnée (car  $C^{m+1}(\Omega)$  est dense dans  $\mathcal{H}^{m+1}(\Omega)$ ), on a

$$\begin{cases} u_i = \langle p(x_i - x) \rangle U_e(x) + \frac{(x_i - x)^{m+1}}{(m+1)!} r(x, x_i) \\ v_j = \langle p'(y_j - x) \rangle U_e(x) + \frac{(x_i - x)^m}{m!} s(x, y_j) \end{cases}$$

où  $U_e(x) = \langle u(x), u'(x), \dots, u^{(m)}(x) \rangle^\top$ . Soit  $p(x)$  la matrice  $P(x)$  adimensionnelle:  $p(x) = P(x)D^{-1}$ , avec

$$D = \begin{bmatrix} 1 & & & \\ & \rho & & \\ & & \ddots & \\ & & & \rho^m \end{bmatrix}$$

Définissons le vecteur des restes de Taylor, et la matrice de poids modifiée

$$\mathfrak{R}(x) = \begin{Bmatrix} \xi_1^{m+1} r(x, x_1) \\ \vdots \\ \xi_{n_x}^{m+1} r(x, x_{n_x}) \\ \eta_1^m s(x, y_1) \\ \vdots \\ \eta_{m_x}^m s(x, y_{m_x}) \end{Bmatrix}, \quad \Omega(x) = \begin{bmatrix} \text{diag}\{w_i(x)\} & 0 \\ 0 & \text{diag}\left\{\left(\frac{r_j}{\rho}\right)^2 w_j(x)\right\} \end{bmatrix}$$

Le critère de construction de l'approximation diffuse Hermite au point  $x$  s'écrit alors

$$\mathcal{E}_x(\{\alpha\}) = \left( p(x)D(\alpha - U_e) - \rho^{m+1}\mathfrak{R}(x) \right)^\top \Omega(x) \left( p(x)D(\alpha - U_e) - \rho^{m+1}\mathfrak{R}(x) \right)$$



**Convergence simple** La solution du système d'optimalité est

$$\begin{aligned} (\alpha - U_\varepsilon)(x) &= \rho^{m+1} D^{-1} (p^\top(x) \Omega(x) p)^{-1} p^\top(x) \Omega(x) \mathfrak{R}(x) \\ &= \begin{bmatrix} \rho^{m+1} & & 0 \\ & \ddots & \\ 0 & & \rho \end{bmatrix} [n(x)] \mathfrak{R}(x) \end{aligned} \quad (2.7)$$

Or les coordonnées du vecteur  $\mathfrak{R}$  sont bornées dès que la fonction  $u^{(m+1)}$  l'est : soit  $M^{(k+1)} > 0$  cette borne sur  $\Omega$ . Il reste donc à étudier la matrice des « fonctions de formes adimensionnelles »  $(p^\top(x) \Omega(x) p(x))^{-1} p^\top(x) \Omega(x)$ . Dans le cas de l'approximation diffuse, il suffit de supposer que l'on conserve l'unisolvance en faisant tendre  $\rho$  vers zéro et en raffinant le semi de nœuds (voir l'hypothèse  $\text{HINV}(x)$  dans [Vil91] ou le concept de boule *h-affine équivalente* dans [DO96, RT98]). Ici il faut aussi tenir compte des termes  $\frac{r_j}{\rho}$  qui peuvent dégénérer. Cependant, on prend en général comme fonction de pondération  $w_j(x) = w_{ref}(\frac{|x-y_j|}{\rho_j})$ , et en choisissant  $r_j = \rho_j$  on a l'encadrement

$$\frac{\rho_j}{\max_{j \in J(x)} (\rho_j)} \leq \frac{r_j}{\rho} \leq 1$$

Il faut donc aussi faire l'hypothèse :  $\forall j \in J, \frac{\rho_j}{\max_{j \in J(x)} (\rho_j)} > \tau > 0$  lorsque  $\rho$  varie<sup>1</sup>. Nous noterons encore  $\text{HINV}(x)$  les deux hypothèses nécessaires.

Il vient finalement le théorème de convergence simple suivant.

**Théorème 10** *Sous les hypothèses  $\text{HINV}(x)$ , si  $u \in \mathcal{H}^{m+1}(\Omega)$ , alors*

$$\forall x, \forall k \in \llbracket 0, m \rrbracket, \exists K(x) > 0, \left| u^{(k)}(x) - \frac{\delta^k \tilde{u}}{\delta x^k}(x) \right| \leq \rho^{m+1-k} K(x)$$

**Convergence uniforme** L'étude de la convergence uniforme demande des hypothèses plus fortes que précédemment. Nous les noterons (HFP) (voir [Vil91]) sans chercher à les expliciter. De plus, dans le cas 1d le théorème de convergence uniforme permet de passer à un majorant de l'erreur  $L^2$  (ou  $H^k$ ) du même ordre que le majorant ponctuel.

En conclusion, nous pouvons à l'aide de la convergence uniforme montrer que la correspondance  $u \rightarrow \tilde{u}$  est continue (avec les espaces et les topologies correctement définis), et dérouler les résultats généraux sur les approximations consistantes dans les espaces de Sobolev [RT98, DO96].

## 2.3 Expérimentation numérique

### 2.3.1 Topo-géométrie

En pratique, nous utilisons l'approximation diffuse Hermite pour la résolution d'équation différentielle. Nous choisissons donc le support des données : les nœuds porteront donc deux paramètres chacun :  $u$  et  $v$ . Les fonctions de pondération  $w_i, w_j$  sont définies selon la stratégie Voronoï annexes B). Nous utilisons les fonctions de pondération de même degré que l'approximation bien que cela ne soit pas nécessaire. On pourrait en effet se contenter des fonctions de pondération « linéaires » pour l'approximation diffuse quadratique car on a deux informations par nœud. Cependant les fonctions de forme présentent alors de très forts gradients.

1. Notons que cette hypothèse semble facile à vérifier.

### 2.3.2 Exemple

Pour mieux nous familiariser avec l'approximation diffuse Hermite, nous avons traité deux variantes d'un exemple simple (6 nœuds, base quadratique). Les courbes sont tracées figures 2.1 et 2.2. Les deux exemples se différencient par les valeurs des dérivées au second et au troisième nœud. Ceci permet de mettre en évidence la prise en compte de la dérivée dans la construction de l'approximation.

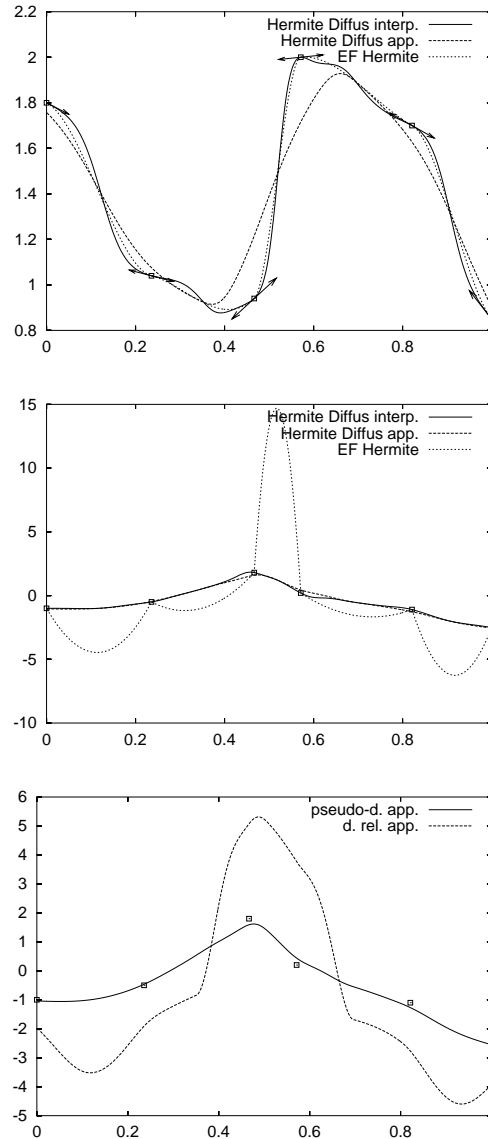


FIG. 2.1: Exemple d'approximation diffuse Hermite ( $\lambda = 1$ ), cas 1. De haut en bas : approximations, dérivées et dérivées réelle et diffuse dans le cas de l'approximation.

Le tracé de la dérivée réelle de l'approximation montre que l'écart entre dérivée réelle et dérivée diffuse est assez important. Cependant on vérifie que la propriété  $\tilde{u}'(x_i) = \frac{\partial \tilde{u}}{\partial x}(x_i)$  est bien vérifiée dans le cadre de l'interpolation diffuse. L'approximation diffuse Hermite donne une pseudo-dérivée très lisse et qui prend principalement en compte les données différentielles.

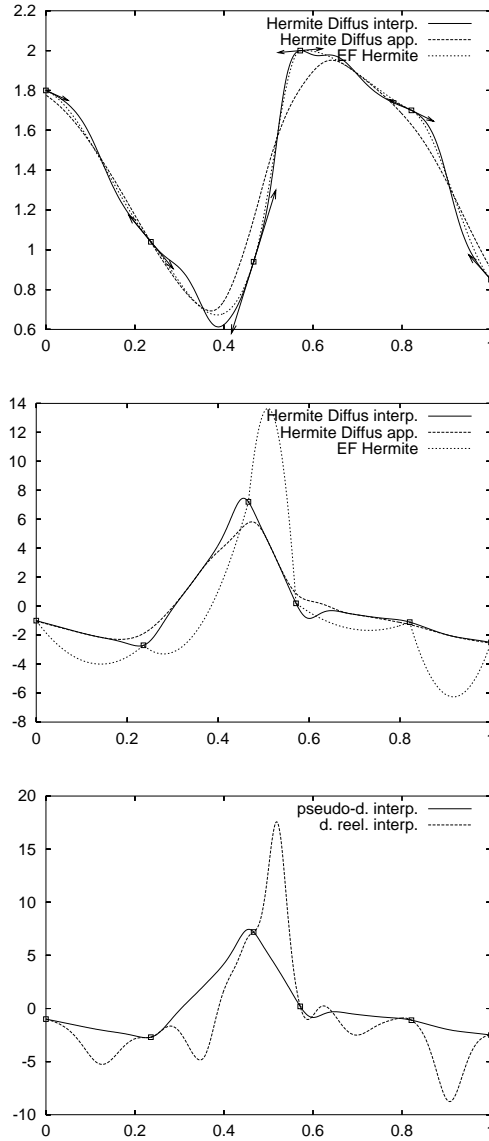


FIG. 2.2: Exemple d'approximation diffuse Hermite ( $\lambda = 1$ ), cas 2. De haut en bas : approximations, dérivées et dérivées réelle et diffuse dans le cas de l'interpolation.

### 2.3.3 Fonctions de forme

Le calcul des fonctions de forme s'appuie sur un algorithme proche de celui décrit en annexe A.1. Nous allons comparer les dérivées diffuse et réelle de l'approximation Hermite. On s'aperçoit que la pseudo-dérivée des fonctions de forme «  $N$  » est très éloignée de la dérivée au sens réel dans le cas de l'approximation.

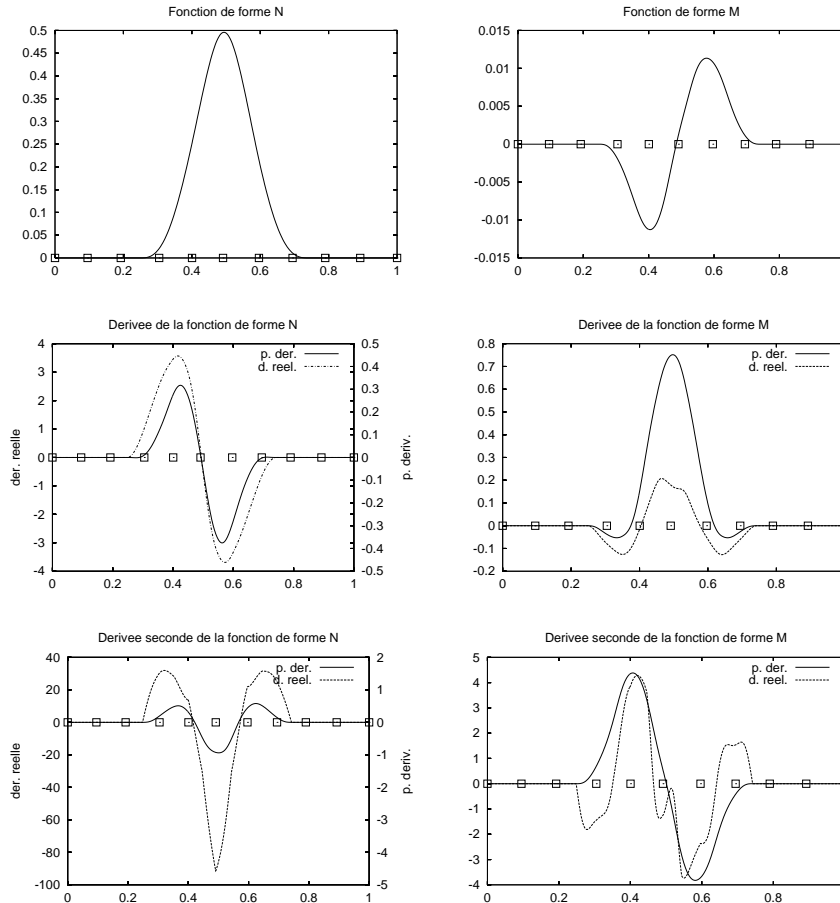
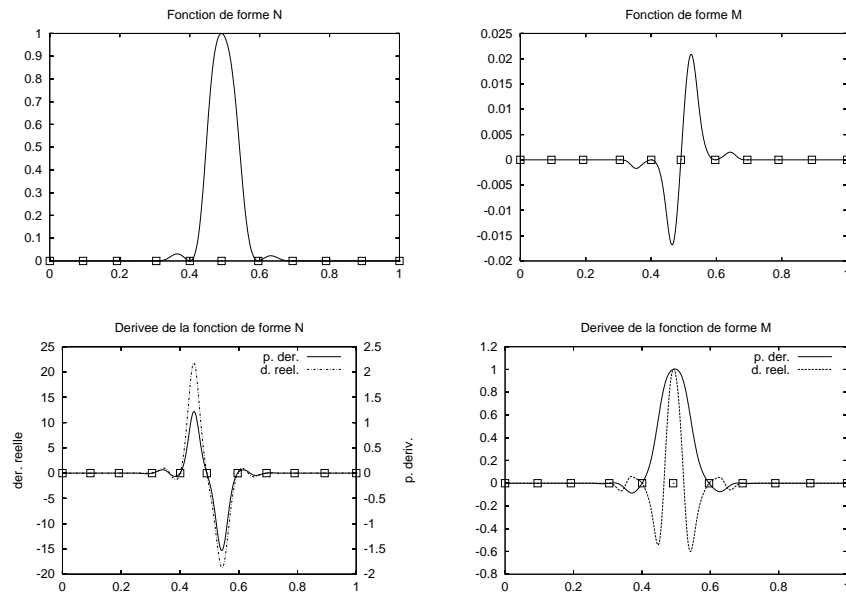
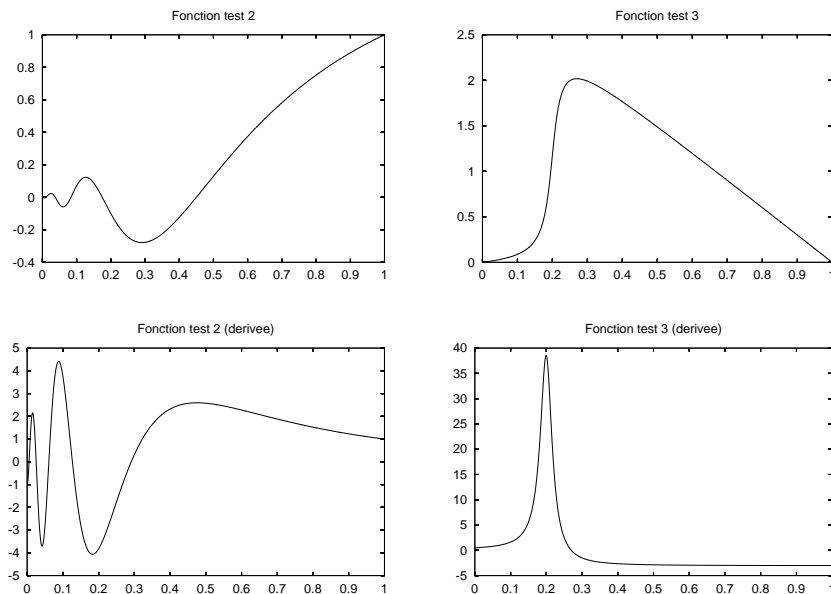


FIG. 2.3: Fonctions de formes de l'approximation diffuse Hermite.

L'écart entre dérivée diffuse et dérivée réelle est moins grand dans le cas de l'interpolation (figure 2.4). Cependant il reste assez important pour donner de grandes difficultés dans le calcul de l'intégration numérique pour la résolution d'équations différentielles.

FIG. 2.4: *Fonctions de formes de l'interpolation diffuse Hermite.*FIG. 2.5: *Fonctions tests 2 et 3.*

### 2.3.4 Convergence

Nous avons vérifié les taux de convergence<sup>2</sup> de l'approximation diffuse Hermite à partir de trois fonctions tests définies sur  $[0, 1]$ . Nous avons choisi la fonction sinus et deux fonctions représentées figure 2.5 page 84. Les nœuds sont répartis de façon régulière, puis perturbés aléatoirement ( $x_i = hi + \epsilon_i$ , où  $h$  est le pas et  $\epsilon_i$  est aléatoire uniforme dans  $[-\frac{h}{5}, \frac{h}{5}]$ ).

Nous avons calculé expérimentalement l'erreur sur  $u, u', u''$  avec les normes  $L^2$  des fonctions erreurs  $\tilde{u} - u, \frac{\delta \tilde{u}}{\delta x} - u', \dots$  (normes ramenées à celle de  $u, u', \dots$ ). Les résultats sont regroupés dans l'annexe D et le tableau 2.1; ils montrent que l'on retrouve bien les taux de convergence prévus au théorème 10. Pour les fonctions 2 et 3, la partie initiale de la courbe de convergence est perturbée par la capture des gradients de la fonction test.

Fonction 1

| base             | Taux de convergence |         |                 |
|------------------|---------------------|---------|-----------------|
|                  | fonction            | dérivée | dérivée seconde |
| $1, x$           | -2.01               | -1.01   |                 |
| $1, x, x^2$      | -3.05               | -2.00   | -1.11           |
| $1, x, x^2, x^3$ | -4.04               | -3.04   | -2.02           |

Fonction 2

| base             | Taux de convergence |         |                 |
|------------------|---------------------|---------|-----------------|
|                  | fonction            | dérivée | dérivée seconde |
| $1, x$           | -2.00               | -1.07   |                 |
| $1, x, x^2$      | -3.13               | -1.97   | -1.25           |
| $1, x, x^2, x^3$ | -4.03               | -3.26   | -1.98           |

Fonction 3

| base             | Taux de convergence |         |                 |
|------------------|---------------------|---------|-----------------|
|                  | fonction            | dérivée | dérivée seconde |
| $1, x$           | -2.00               | -0.99   |                 |
| $1, x, x^2$      | -3.07               | -2.01   | -1.09           |
| $1, x, x^2, x^3$ | -3.99               | -3.03   | -1.98           |

TAB. 2.1: *Taux de convergence expérimentaux de l'approximation diffuse Hermite.*

## 2.4 Conclusion

Nous avons étudié dans ce chapitre l'approximation diffuse Hermite; elle admet les propriétés usuelles de l'approximation diffuse et nous obtenons une formulation en fonctions de forme généralisée. Ce théorème permet de mettre en œuvre les techniques numériques les plus efficaces pour le calcul des fonctions de forme. Nous énonçons un théorème de convergence dont la démonstration fait apparaître l'importance du terme d'homogénéisation  $r_j^2$ . Dans le formalisme de [SF73, RT98, DO96], ce terme assure les propriétés des fonctions de forme vis-à-vis des transformations affines qui sont à la base de la convergence.

---

<sup>2</sup> La théorie prévoit que l'erreur tracée en diagramme logarithmique suit une droite dont la pente est le taux de convergence de l'approximation.

Les expérimentations numériques que nous avons menées montre que l'approximation diffuse Hermite donne des résultats satisfaisant et une pseudo-dérivée très lisse (exemples). Les taux de convergence obtenus numériquement sont conformes aux taux prévus par le théorème de convergence.

L'étude des fonctions de forme montre un écart important entre dérivée réelle et dérivée diffuse (cas des fonctions «  $N$  »). Ceci implique qu'il sera très difficile de définir une intégration numérique adaptée à ces fonctions de forme pour la résolution d'équations aux dérivées partielles. De plus, les données différentielles ont une grande importance dans le calcul de la pseudo-dérivée, il faudra donc être très vigilant lorsque ces données sont bruitées. Le calcul de la courbure de surfaces triangulées (partie III chapitre 1) donne un exemple de telles données (poids des normales).

Troisième partie

Application à la modélisation  
surfactive





## Chapitre 1

# Courbure d'une surface triangulée

La courbure donne une image de la variation des surfaces. En effet, lorsque la courbure est élevée localement, la surface varie beaucoup, elle est lisse dans le cas contraire. A la limite, les courbures d'un plan sont nulles toutes les deux. La figure 1.2 illustre cet aspect de la courbure.

En CAO classique, les surfaces utilisées sont souvent des portions de tore, cylindre, cône, ... or ces surfaces ont des signatures très caractéristiques en terme de courbure :

- le plan a deux courbures nulles,
- le cylindre:  $k_1$  est l'inverse du rayon, et  $k_2$  est nulle,
- la sphère a deux courbures constantes et égales à l'inverse de son rayon
- ...

La figure 1.1 montre le cas du tore et du cône.

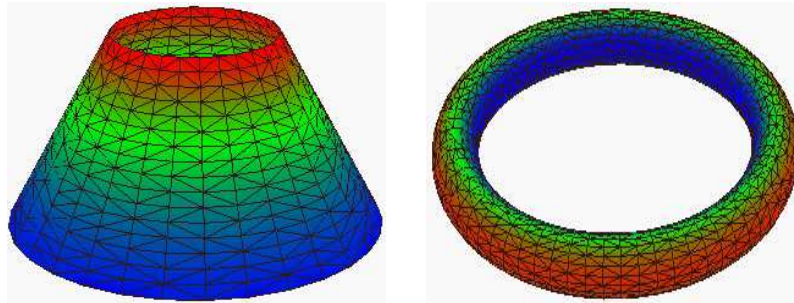


FIG. 1.1: Courbure  $k_1$  du cône et  $k_2$  du tore.

Enfin, les mécanismes de déformation des solides donnent certaines propriétés aux courbures des surfaces résultantes. Ainsi, en géologie, la courbure du toit (ou du mur) d'une formation donne des renseignements sur sa déformation. On sait par exemple qu'une surface de courbure de Gauss nulle peut-être dépliée par une transformation isométrique (voir [Lis94] par exemple).

La courbure est donc un **indicateur** qui nous permet de **détecter** certaines caractéristiques de la surface: ses variations, son expression paramétrique, son mode de déformation ...



Modèle complet



Détail (tête de la vache)

FIG. 1.2: *Modèle de vache. La courbure  $k_1$  montre les zones de fortes variation de la surface (la courbure est calculée au nœud puis interpolée dans les triangles selon l'échelle de couleur).*

Cependant, le calcul de la courbure d'une surface en trois dimensions est difficile car une description  $C^2$  est nécessaire. Or l'approximation diffuse est bien adaptée lorsqu'on veut calculer des dérivées d'ordre élevé. Il est donc naturel de penser l'appliquer au calcul de courbure. Cependant l'approximation diffuse est une méthode d'approximation fonctionnelle et son adaptation à des calculs surfaciques n'est pas immédiat ; nous avons donc choisi une description constructive de l'algorithme de calcul de la courbure par approximation diffuse Hermite.

Nous commençons par rappeler la définition de la courbure d'une surface, puis présentons rapidement les méthodes usuelles de calcul de la courbure. La troisième partie propose la description et l'analyse de l'algorithme proposé. Nous aboutissons à une version basée sur l'approximation diffuse Hermite que l'on peut qualifier d'algorithme « haute résolution ». Enfin, la quatrième partie compare la nouvelle méthode avec une méthode basée sur les patches de Bézier.

## 1.1 Courbure d'une surface 3d

La courbure est un élément de géométrie différentielle à laquelle l'ouvrage de Do Carmo [Car76] donne une introduction. Nous utilisons ici une présentation géométrique de la courbure qui s'appuie sur la courbure des courbes. Nous passons rapidement à un mode de calcul, puis nous décrivons son intérêt en modélisation surfacique.

Soit  $\mathcal{S}$  une surface de  $\mathbb{R}^3$  que nous supposons  $C^2$ . En chaque point  $P$ ,  $\mathcal{S}$  admet une normale  $\vec{n}$  et à chaque plan  $\Pi$  contenant le couple  $(P, \vec{n})$ , on associe la courbure de la trace de  $\mathcal{S}$  sur  $\Pi$  (qui existe car la courbe  $\Pi \cap \mathcal{S}$  est aussi  $C^2$ ). La courbe obtenue  $k(\theta)$  sur  $[0, 2\pi[$  admet un maximum  $k_1$  et un minimum  $k_2$ . On démontre par ailleurs que  $k_1$  et  $k_2$  sont portés par des plans orthogonaux, ce sont les courbures principales.

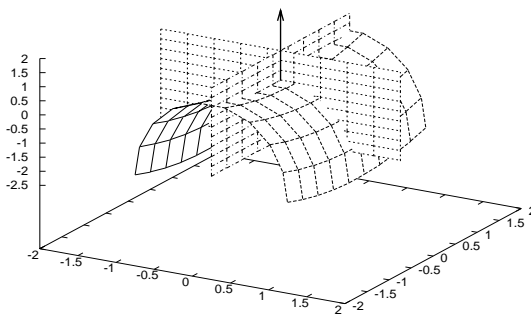


FIG. 1.3: Surface, normale et plans maximaux définissant les courbures principales  $k_1$  et  $k_2$ .

Il est possible de calculer  $k_1, k_2$  et les directions principales analytiquement lorsqu'on connaît un paramétrage  $(u, v)$  de  $\mathcal{S}$  (nous noterons  $\rho(u, v)$  le point courant de  $\mathcal{S}$ , et  $\cdot$  le produit scalaire usuel). Les courbures principales  $k_1$  et  $k_2$  sont les valeurs propres de la matrice symétrique  $G^{-1}L$  où

$$G = \begin{bmatrix} \frac{\partial \rho}{\partial u} \cdot \frac{\partial \rho}{\partial u} & \frac{\partial \rho}{\partial u} \cdot \frac{\partial \rho}{\partial v} & \frac{\partial \rho}{\partial v} \cdot \frac{\partial \rho}{\partial u} \\ \frac{\partial \rho}{\partial u} \cdot \frac{\partial \rho}{\partial v} & \frac{\partial \rho}{\partial v} \cdot \frac{\partial \rho}{\partial v} & \frac{\partial \rho}{\partial v} \cdot \frac{\partial \rho}{\partial v} \end{bmatrix} \quad L = \begin{bmatrix} \vec{n} \cdot \frac{\partial^2 \rho}{\partial u^2} & \vec{n} \cdot \frac{\partial^2 \rho}{\partial u \partial v} \\ \vec{n} \cdot \frac{\partial^2 \rho}{\partial v \partial u} & \vec{n} \cdot \frac{\partial^2 \rho}{\partial v^2} \end{bmatrix}$$

Les directions principales de courbures sont aussi associées aux directions principales de cette matrice (figure 1.4). On définit aussi habituellement

- la courbure de Gauss  $K$  par  $K = k_1 k_2$  (déterminant de  $G^{-1}L$ ),

- la courbure moyenne  $H$  par  $H = \frac{k_1+k_2}{2}$  (demi-trace de  $G^{-1}L$ ).

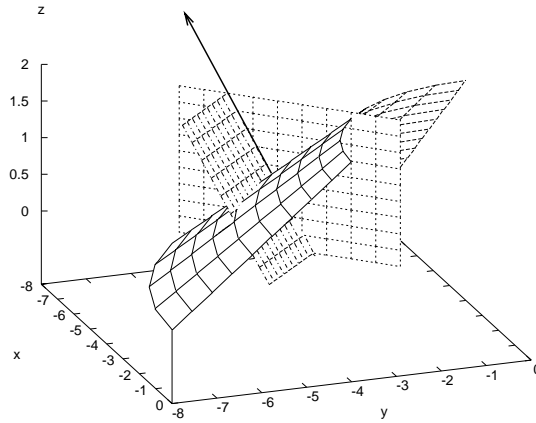


FIG. 1.4: Plans associés aux courbures principales pour un cône d'axe  $Oz$ .

## 1.2 Calcul de la courbure

Le calcul de la courbure est abondamment traité dans la littérature relative aux « range images »<sup>1</sup>, mais peu pour les surfaces triangulées. Nous allons donc décrire les méthodes relatives à ces données et aux nuages de points 3d, puis nous discuterons le cas des surfaces triangulées.

Dans leur étude bibliographique et expérimentale, Mc Ivor et Valkenburg [IW97] distinguent quatre classes de méthodes de calcul de la courbure d'une surface 3d :

- Les méthodes géométriques ;
- Les méthodes par ajustements ;
- Les méthodes par différences finies ;
- Les méthodes par facetisation.

Toutes ces méthodes s'appuient sur les voisins  $P$  de  $\rho$  le point d'évaluation de la courbure. Ce voisinage peut être calculé a priori ou être un résultat du calcul (voir la première méthode géométrique). De la même façon, la normale à la surface en chaque point peut être une donnée de l'algorithme ou un résultat. Dans le cas où ce n'est pas une donnée, il est toujours possible de l'évaluer comme celle du plan des moindres carrés du nœud et de ses voisins.

### 1.2.1 Méthodes géométriques

Il existe trois méthodes d'estimation de la courbure par analyse géométrique. La première étudie les variations des normales : dans le voisinage du point de calcul  $\rho$ , où la normale est  $\vec{n}(\rho)$ , on considère l'ensemble des points voisins de  $\rho$  dont les normales forment un angle faible avec celle de  $\rho$  :

$$\{P \in S, \vec{n}(\rho) \cdot \vec{n}(P) > C_l\}.$$

1. Les range images sont des cartes donnant la distance d'un objet à la focale d'un senseur 3d. On stocke ces profondeurs sur une grille régulière, selon un format image (pixels).

Cet ensemble projeté sur le plan tangent en  $\rho$  est proche d'une ellipse dont les grands axes marquent les directions principales de courbure. La valeur des courbures principales est liée aux rayons de l'ellipse. Cette méthode a été décrite dans [YT94] cité dans [IW97].

La seconde méthode est proposée par [LT90] (cité dans [IW97]) : l'idée de base est encore d'étudier les variations des normales. Pour cela, on calcule le plan des moindres carrés en prenant en compte les nœuds du voisinage de  $\rho$ , le point de calcul. Ce plan définit une normale que l'on considère être la normale  $\vec{n}$  à  $S$  en  $\rho$ . Pour chaque voisin de  $\rho$ ,  $P$ , on construit le point  $Q$  défini par  $\vec{OQ} = \vec{O\rho} + \vec{n}(P)$ . Ces points sont projetés sur le plan tangent en  $\rho$ , et la diagonalisation de la matrice  $2 \times 2$  ( $M$  est le projeté de  $Q$  sur le plan tangent en  $\rho$ ) :

$$C_p = \frac{1}{n} \sum_{i=1}^n (M_i - \bar{M})(M_i - \bar{M})^\top$$

avec

$$\bar{M} = \frac{1}{n} \sum_{i=1}^n M_i$$

permet de calculer les courbures principales et leurs directions.

La troisième méthode géométrique est très simple, car elle s'appuie sur l'analyse d'une portion de sphère  $(\Omega, r)$  ; elle est décrite dans [HK97]. Si  $d_{\rho P}$  est la distance de  $\rho$  et  $P$ , et  $\alpha_{\rho P}$  une mesure de l'angle  $(\vec{\Omega P}, \vec{\Omega \rho})$ , alors l'estimation de la courbure dans la direction  $\vec{\rho P}$  est (le signe dépend de l'orientation de la surface)

$$c_{\vec{\rho P}} = \frac{1}{r} \approx \frac{\alpha_{\rho P}}{d_{\rho P}} \approx \arccos(\vec{n}(\rho) \cdot \vec{n}(P)) / d_{\rho P}$$

Dans le cas général, les estimations de  $k_1$  et  $k_2$  sont alors données par

$$\begin{aligned} k_1 &= \max_{P \in V(\rho)} c_{\vec{\rho P}} \\ k_2 &= \min_{P \in V(\rho)} c_{\vec{\rho P}} \end{aligned}$$

Ce calcul approché permet aux auteurs de mailler les surfaces avec une densité liée à la courbure.

### 1.2.2 Méthodes par ajustement

Les méthodes de calcul de la courbure par ajustement utilisent la propriété fondamentale de la courbure : c'est une caractéristique intrinsèque de la surface ; c'est-à-dire qu'elle ne dépend pas de la paramétrisation choisie pour son calcul. L'idée de base est donc d'ajuster localement une fonction polynomiale d'ordre 2

$$s(x, y) = ax^2 + bxy + cy^2 + dx + ey + f$$

par moindres carrés au voisinage de chaque point dans le repère défini par l'estimation de la normale à la surface. Les dérivées de  $s$  donnent une estimation de celles de la surface et permettent ainsi le calcul de la courbure. La méthode diffuse s'inscrit dans cette catégorie et justifie la technique des moindres carrés glissants.

Plusieurs variantes de cette méthode sont décrites dans [IW97] basée sur une forme de  $s$  différente. La forme la plus générale permet ici de prendre en compte les erreurs potentielles sur la position ( $s$ ) et l'estimation de la normale ( $d$  et  $e$ ).

Cette forme de calcul présente l'intérêt de séparer le calcul en deux étapes distinctes :

1. Évaluation de la normale et calcul du repère local

## 2. Ajustement de $s$ et calcul de la courbure

On peut regrouper ces deux étapes en minimisant un critère dans lequel les paramètres du changement de repère (les trois angles d'Euler  $(\theta, \varphi, \phi)$ ) sont des inconnues non-linéaires du problème d'ajustement. Cette méthode est la plus robuste d'après [IW97], mais nécessite beaucoup plus de calcul.

Une analyse de la méthode par ajustement linéaire est proposée dans [Gou97], l'auteur montre en particulier que la méthode admet un biais systématique. Nous retrouverons ce biais avec l'approximation diffuse, mais ceci n'est pas pénalisant pour la reconnaissance des formes car c'est la séparation des courbures et non leur valeur qui importe.

### 1.2.3 Différences finies

Lorsqu'on travail sur des nuages de points 3d mesurés, ils ont souvent une structure simple qui permet de calculer les dérivées nécessaires au calcul de la courbure par différences finies. Cette méthode n'est pas très robuste d'après [IW97] si les opérateurs ne sont pas lissés (par un noyau gaussien par exemple). Ceci présente un grand désavantage, car le lissage peut gommer des discontinuités entre surfaces, qui sont des guides pour la reconnaissance des formes.

### 1.2.4 Facettes

Le *facet model* proposé par Haralick, (voir par exemple [HS92]) est une généralisation des différences finies. Ce modèle comporte plusieurs aspects :

1. Il ne s'applique qu'aux cartes de profondeur (range images).
2. En chaque point de donnée, on estime les dérivées par ajustement d'un polynôme comme dans la méthode par ajustement.
3. On remarque que l'estimation des dérivées est une fonction linéaire des profondeurs. On peut donc précalculer l'ajustement et obtenir les dérivées par application d'un opérateur de voisinage.

On note que cette méthode fait appel implicitement à la notion de fonction de forme. Notons le parallèle saisissant avec le travail de Litzka et Orkish dans le cadre des équations différentielles [LO80] où des formules de type différences finies sont aussi calculées par ajustement de polynômes.

### 1.2.5 Cas des surfaces triangulées

Le calcul des courbures d'une surface triangulée peut suivre différentes stratégies. La première est d'oublier la triangulation et de traiter le nuage de points comme précédemment. La seconde stratégie est de s'appuyer sur la connectique de la triangulation pour définir le voisinage du point  $\rho$  et d'utiliser ensuite une des méthodes précédentes, comme dans [SRD94] par exemple, où les auteurs ajustent une quadrique à différentes échelles (voisinage plus ou moins grand). Enfin, on peut considérer que les triangles sont des patches de Bézier et évaluer les dérivées à partir de ces patches. Cette dernière méthode est décrite dans [SM97], elle est très rapide, mais peu stable. Nous la décrirons dans la quatrième section de ce chapitre puis comparerons ses résultats avec le calcul que nous proposons.

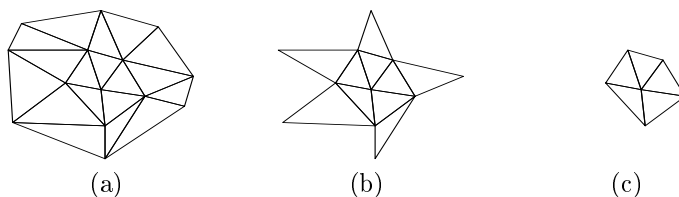


FIG. 1.5: De gauche à droite, le voisinage d'approximation complet (a), voisinage au sens restreint (b), et la première couronne de voisins (c) du nœud central.

### 1.3 Calcul diffus de la courbure

Nous décrivons dans cette partie les différents algorithmes de calcul de la courbure que nous avons testés. Nous commençons par définir les paramètres de l'approximation diffuse qui nous permettront de mener le calcul à bien, puis nous analysons l'influence de chacun d'entre eux. On remarque alors que la taille du voisinage est un paramètre très important pour la qualité des résultats, et en particulier près des intersections de surfaces. Ceci nous amène à utiliser l'approximation diffuse Hermite, afin de limiter la taille du voisinage de calcul. Cette nouvelle approximation est appliquée pour un calcul aux nœuds du maillage et un calcul aux centres des triangles.

#### 1.3.1 Voisinage

Le calcul analytique des courbures principales d'une surface nécessite un paramétrage. Or on n'en dispose pas lorsque la surface est décrite par un maillage non structuré. Nous utilisons donc, comme pour les méthodes par ajustement, un repère local dans lequel nous cherchons une approximation de la surface sous la forme  $s(x, y)$ . Comme  $\mathcal{S}$  est  $C^2$ , elle est toujours localement altimétrique, c'est-à-dire qu'en chaque point, il existe un voisinage tel que la surface soit altimétrique sur ce voisinage. Nous allons faire l'hypothèse que cette propriété est conservée sur la surface discrète  $\mathcal{S}_d$  décrite par les nœuds et leurs connections. Pour donner un sens à cette hypothèse, nous devons définir le voisinage d'approximation au nœud  $N_0$ . Comme nous ne connaissons pas la surface réelle  $\mathcal{S}$ , nous allons utiliser la topologie du maillage.

Le voisinage d'approximation le plus naturel pour  $N_0$  est constitué des nœuds qui lui sont connectés : **la première couronne** (ou la couronne de voisins). Cependant, nous avons besoin d'au moins 6+1 nœuds pour construire notre approximation de  $\mathcal{S}$  (6 pour  $1, x, y, x^2, xy, y^2$  et un pour la continuité), et la première couronne ne contient pas toujours assez de nœuds<sup>2</sup>. Aussi nous utilisons un voisinage étendu où l'on ajoute

- soit tous les nœuds connectés à ceux de la première couronne (voisinage complet)
- soit tous les nœuds connectés à deux nœuds de la première couronne (voisinage restreint).

Nous sommes maintenant en mesure de formuler notre hypothèse de base sur la surface discrète  $\mathcal{S}_d$  :

**Hypothèse 3** *Pour chaque nœud de la surface  $\mathcal{S}_d$ , la portion de surface définie par son voisinage (complet ou restreint) est altimétrique par rapport au plan des moindres carrés qu'il définit avec ses voisins (complets ou restreints).*

<sup>2</sup>. Surtout sur les bords de la surface



La figure 1.6 montre des surfaces ne vérifiant pas notre hypothèse.

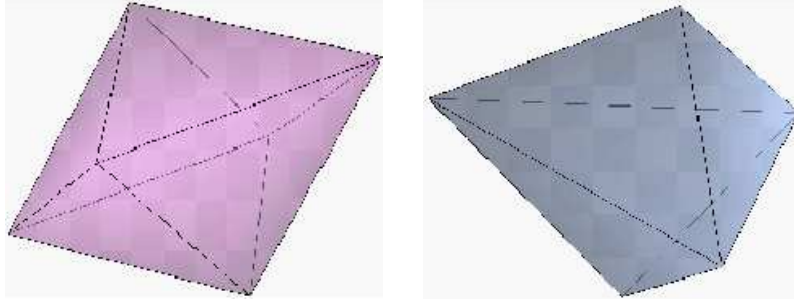


FIG. 1.6: Deux surfaces ne vérifiant pas l'hypothèse initiale

### 1.3.2 Pondérateurs

L'algorithme de calcul de la courbure d'une surface triangulée en ses nœuds, se déroule donc facilement :

pour chaque nœud  $N_0$  de la surface  $\mathcal{S}$

- chercher ses voisins
- calculer le plan des moindres carrés des voisins de repère  $(N_0, \vec{u}, \vec{v})$ , et passer dans le repère  $(N_0, \vec{u}, \vec{v}, \vec{u} \times \vec{v})$
- calculer l'interpolation diffuse dans ce repère, ainsi que  $G$  et  $L$ ,
- calculer les courbures principales, leurs directions.

Le calcul de l'approximation diffuse suppose que l'on soit capable de calculer les valeurs des fonctions  $w_i(\cdot)$ . Il est évident que la forme habituelle de  $w_i(\cdot)$  ne peut pas être utilisée car on devrait mesurer les distances sur la surface. Nous nous restreignons donc pour l'instant à calculer l'approximation diffuse de la surface en ses nœuds.

Avec notre restriction, il nous suffit de calculer les valeurs de  $w_i(x_j)$ , pour tout  $j$  du voisinage du nœud  $i$ . Deux stratégies sont possibles :

1. Utiliser des poids « discrets » avec les valeurs

- 1 au nœud
- $\frac{1}{2}$  sur la première couronne,
- $\frac{1}{3}$  aux autres nœuds du voisinage,
- 0 ailleurs.

On remarque qu'une fonction  $w_i(\cdot)$  admettant ces valeurs peut être construite en s'appuyant sur la triangulation (voir figure 1.7).

2. Utiliser des poids s'appuyant sur la distance *sur le plan des moindres carrés*. On écrit alors  $w(x, x_i) = w_{ref} \left( \frac{x - x_i}{\rho(x)} \right)$  et  $\rho(x)$  est choisi de façon à assurer l'existence de l'approximation partout. Cette stratégie pose cependant des problèmes de continuité pour le calcul en d'autres points que les nœuds (le système de coordonnées bouge avec le point d'évaluation, ce qui rend difficile l'analyse de cette approximation).

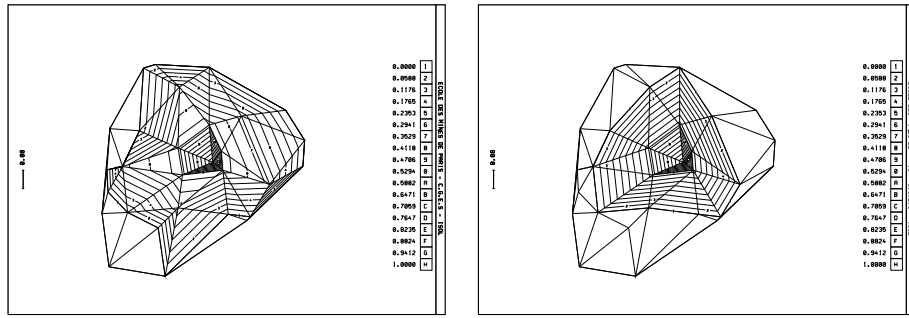


FIG. 1.7: Construction des fonctions  $w_i(\cdot)$  en s'appuyant sur la triangulation. À gauche, le cas d'un voisinage restreint. À droite, fonction appuyée sur la première couronne uniquement.

### 1.3.3 Premiers résultats

L'algorithme obtenu peut facilement être testé. L'idée de base est de se placer sur des surfaces dont on connaît les courbures a priori pour juger de la qualité de notre évaluation. Nous décrivons pour commencer deux cas particuliers (deux cylindres) pour analyser l'influence des deux paramètres de l'approximation diffuse sur le calcul. Ceci nous mène naturellement à une évolution de notre méthode de calcul.

#### Influence du voisinage

Le cas du cylindre nous permet de mettre en œuvre des outils statistiques simples pour l'analyse des résultats. Cependant, on peut commencer par une analyse qualitative en s'appuyant sur la figure 1.8. La courbure  $k_1$  est presque constante, mais toujours supérieure à la valeur réelle (ici 0.2). La courbure  $k_2$  est assez faible par rapport à la valeur  $k_1$  (le rapport des valeurs moyennes est supérieur à 100 pour les nœuds intérieurs). On s'aperçoit que les nœuds « de  $k_1$  suspects » sont sur les bords du cylindre; ce léger effet de bord semble se confirmer sur la figure concernant  $k_2$ , mais il n'est pas très marqué.

|       | v. restreint |          | v. complet |          |
|-------|--------------|----------|------------|----------|
|       | moyenne      | variance | moyenne    | variance |
| $k_1$ | 2.07e-01     | 8.35e-04 | 2.21e-01   | 5.76e-03 |
| $k_2$ | 1.63e-03     | 1.22e-04 | 1.59e-02   | 2.70e-03 |

TAB. 1.1: Moyennes et variances de  $k_1$  et  $k_2$  pour des voisinages complet et restreint. La courbure  $k_1$  théorique est 0.2 et la courbure  $k_2$  est nulle.

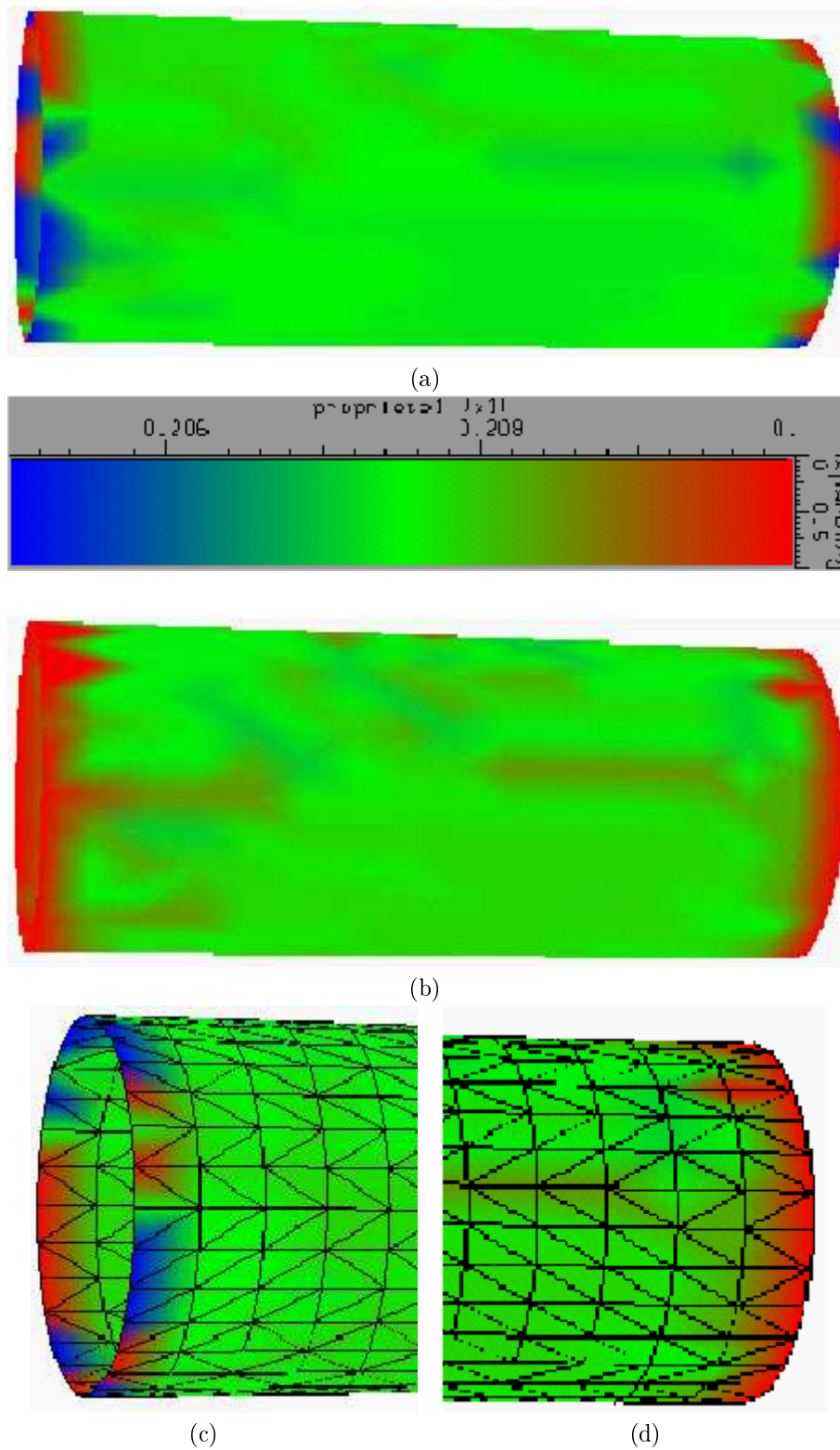


FIG. 1.8: *Courbures principales  $k_1$  (a) et  $k_2$  (b) sur un cylindre, voisinage complet. L'échelle de couleur correspond à  $k_1$  et est graduée de 0 à  $10^{-3}$  pour  $k_2$ . (c) et (d) montrent l'influence de la connectique sur la qualité du résultat aux extrémités.*

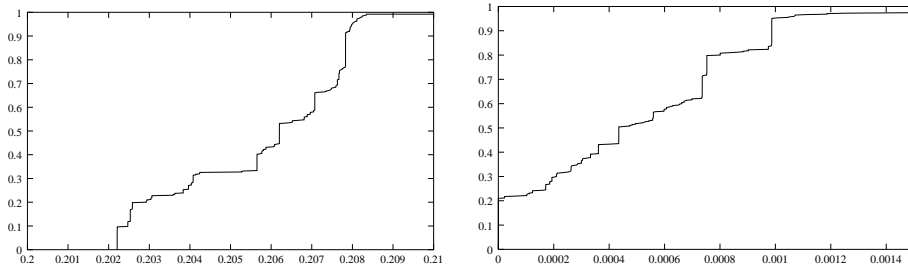


FIG. 1.9: Courbe d'accumulation des courbures  $k_1$  et  $k_2$  en voisinage restreint. En abscisse est reportée la valeur de la courbure, et en ordonnée, le nombre de points dont la courbure est inférieure à l'abscisse (ramené au nombre total de points). Les marches d'escalier marquent des groupes de nœuds de même connectique.

Les moyennes et variances expérimentales de  $k_1$  et  $k_2$  sont reportées dans le tableau 1.1. On remarque immédiatement que le voisinage restreint conduit à de meilleurs résultats que le voisinage complet. Les courbes d'accumulation (pour le calcul en voisinage restreint uniquement) se trouvent sur la figure 1.9. Ces courbes nous donnent une bonne image des performances de notre algorithme de calcul :

- 99% des nœuds ont une valeur de  $k_1$  correcte à 5%,
- mais seulement 33% à 2.5%.
- 20% des nœuds ont une valeur de  $k_2$  exacte (nulle),
- et 95% une valeur inférieure à  $1e-3$

Cette rapide analyse nous montre que les résultats sont corrects lorsqu'on travaille en voisinage restreint. Ce constat va dans le même sens que la déviation observée pour la valeur de  $k_1$  : **les points extérieurs ont tendance à donner une surface plus courbée qu'elle ne l'est réellement. Il est donc préférable de travailler avec des voisinages de petites tailles (restreints).**

#### Influence des pondérateurs $w_i$ .

Nous avons utilisé un maillage de densité non homogène (voir figure 1.13) pour évaluer l'influence des pondérateurs sur le calcul des courbures.

|          | vois. complet |          | vois. restreint |          |
|----------|---------------|----------|-----------------|----------|
|          | discret       | distance | discret         | distance |
| $m(k_1)$ | 1.004         | 1.005    | 1.004           | 1.004    |
| $v(k_1)$ | 1.26e-2       | 1.62e-2  | 1.37e-2         | 1.38e-2  |
| $m(k_2)$ | 4.95e-3       | 5.32e-3  | 9.35e-3         | 9.73e-3  |
| $v(k_2)$ | 1.56e-3       | 1.65e-3  | 1.68e-3         | 1.69e-3  |

TAB. 1.2: Moyennes et variances de  $k_1$  et  $k_2$  sur un cylindre maillé de façon hétérogène ; la valeur théorique de  $k_1$  est 1.

Les résultats chiffrés montrent que les calculs effectués avec des poids discrets sont légèrement meilleurs. On ne change cependant pas d'ordre de grandeur et l'expérience sur les cas réels montre que le calcul des  $w_i$  n'a pas de réelle influence

lorsqu'on exploite les résultats. Ceci ne nous permet pas de trancher de façon objective, mais les figures dans la suite concernent toutes (sauf mention contraire) des calculs utilisant des poids discrets.

On pouvait s'attendre a priori à ce résultat, car les maillages répondent à certaines contraintes sur l'angle minimal des triangles. Cela assure alors que les distances inter-nœuds varient peu au voisinage d'un nœud donné, et donnent ainsi un ensemble de pondérateurs presque égaux.

### Conclusion

Les exemples présentés tendent tous à montrer que le voisinage d'interpolation joue un grand rôle dans la qualité des résultats. Le cas réel montre que ce voisinage gouverne directement l'échelle minimale de travail. Ce constat nous invite donc à utiliser des voisinages plus locaux, mais nous avons vu que la première couronne ne contient pas assez de points pour évaluer l'approximation diffuse en un nœud. Nous devons donc trouver une autre information pour guider l'approximation.

### 1.3.4 Prise en compte des normales aux faces

Nous n'avons utilisé jusqu'ici que l'information topologique du maillage et les positions des nœuds. Cependant, chaque facette de la triangulation porte une information géométrique importante : sa normale. Cette information répond à notre objectif, car elle est aussi locale que la première couronne ; il faut donc utiliser l'approximation diffuse Hermite pour prendre en compte ces normales.

L'approximation Hermite a été présentée en 1d, nous allons donc en décrire rapidement la version que nous avons utilisée ici. Le critère de construction de l'approximation diffuse est (le terme  $r_j^2$  associé aux normales est inclu dans  $\lambda$ ) :

$$\mathcal{E}(\alpha) = \sum_{i \in I_x} w(x, x_i) (\bar{u}(x_i) - u_i)^2 + \lambda \sum_{t \in T_x} w(x, x_t) \|n_t - n(x_t)\|^2 \quad (1.1)$$

- $T_x$  désigne l'ensemble des indices des points où l'on connaît une normale à la surface,
- $n_t = \langle n_t^x, n_t^y, 1 \rangle$  désigne la normale connue,
- $n(x_t) = \langle -\frac{\partial \bar{u}}{\partial x}(x_t), -\frac{\partial \bar{u}}{\partial y}(x_t), 1 \rangle$  désigne la normale calculée pour  $\bar{u}$ ,
- $\lambda$  est un paramètre de réglage entre les deux termes de  $\mathcal{E}$ .

La minimisation de  $\mathcal{E}$  nous mène à un système linéaire en  $\alpha$  comme précédemment, où s'ajoutent les termes relatifs aux normales. Nous noterons  $W_n(x)$  la matrice diagonale des  $w(x, x_t)$ ,  $N_x$  le vecteur des  $n_t^x$ ,  $N_y$  celui des  $n_t^y$ , et

$$P_x(x) = \begin{bmatrix} 0 & 1 & 0 & 2(x - x_{t_1}) & (y - y_{t_1}) & 0 & 3(x - x_{t_1})^2 & \dots \\ 0 & 1 & 0 & 2(x - x_{t_2}) & (y - y_{t_2}) & 0 & 3(x - x_{t_2})^2 & \dots \\ 0 & 1 & 0 & \vdots & \vdots & \vdots & \vdots & \\ 0 & 1 & 0 & 2(x - x_{t_{n_t}}) & (y - y_{t_{n_t}}) & 0 & 3(x - x_{t_{n_t}})^2 & \dots \end{bmatrix}$$

$$P_y(x) = \begin{bmatrix} 0 & 0 & 1 & 0 & (x - x_{t_1}) & 2(y - y_{t_1}) & 0 & \dots \\ 0 & 0 & 1 & \vdots & \vdots & \vdots & \vdots & \\ 0 & 0 & 1 & 0 & (x - x_{t_{n_t}}) & 2(y - y_{t_{n_t}}) & 0 & \dots \end{bmatrix}$$

Le vecteur  $\{\alpha\}$  est alors solution du système

$$\begin{cases} A(x)\{\alpha(x)\} &= \{b(x)\} \\ A(x) &= P^\top(x)W(x)P(x) + \\ &\quad \lambda(P_x^\top(x)W_n(x)P_x(x) + P_y^\top(x)W_n(x)P_y(x)) \\ \{b(x)\} &= P(x)W(x)U + \lambda(P_x(x)W_n(x)N_x + P_y(x)W_n(x)N_y) \end{cases} \quad (1.2)$$

Dans notre cadre, l'approximation au nœud  $N_0$  se construit avec les paramètres suivants :

- $I_x$  est composé des indices des nœuds de la première couronne, et de celui de  $N_0$ ,
- les pondérateurs discrets  $w_i$  sont conservés,
- $T_x$  est composé des indices des triangles admettant  $N_0$  comme sommet,
- $x_t$  est un point porté par un triangle de  $T_x$ ,
- les pondérateurs  $w_t$  seront choisis tous égaux en suivant la même idée que pour les pondérateurs discrets (on peut construire une fonction  $w_t(\cdot)$  en s'appuyant sur le triangle  $t$ )

Pour finir, il faut choisir le point du triangle que nous utiliserons pour porter la normale. Deux candidats se présentent naturellement :

1. Supposons que notre surface soit une sphère. Le couple  $(x_t, \vec{n}_t)$  définit une droite qui intersecte la sphère en  $q$  où la normale à la sphère est  $\vec{n}_q$ . Les vecteurs  $\vec{n}_t$  et  $\vec{n}_q$  sont parallèles si et seulement si  $x_t$  est le centre du cercle circonscrit au triangle  $t$ .
2. Les arêtes des triangles portent les discontinuités de la surface triangulée. Le point à égale distance de ces discontinuités est le centre du cercle inscrit.

J'ai choisi le centre du cercle inscrit, car il est toujours à l'intérieur du triangle. Sa justification semble aussi plus générale.

Nous sommes en mesure d'appliquer cette nouvelle approximation diffuse dans notre algorithme. Notons aussi que l'hypothèse de base (surface localement altimétrique) est toujours vérifiée avec ce type de voisinage.

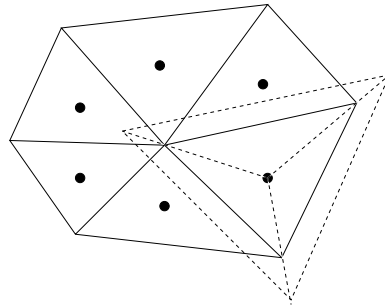


FIG. 1.10: Construction d'une fonction d'atténuation pour les points  $x_t$  sur une surface triangulée. La fonction d'atténuation a la forme d'une pyramide à base triangulaire (tiretés) qui « déborde » légèrement du triangle concerné.

### 1.3.5 Calcul du paramètre de réglage $\lambda$

Le nouveau calcul de l'approximation diffuse fait intervenir un paramètre de réglage  $\lambda$  entre les termes « positions » et « normales ». Étudions le cas de notre premier cylindre (rayon 5) pour comprendre son influence sur le calcul des courbures.

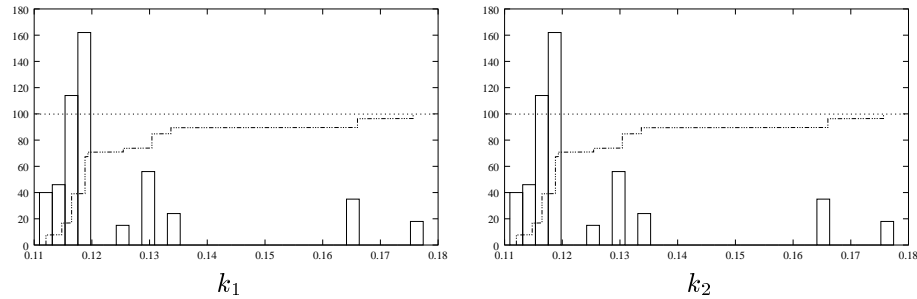


FIG. 1.11: *Histogramme et courbe d'accumulation de  $k_1$  et  $k_2$  pour la valeur  $\lambda = 1$ .*

Pour  $\lambda = 1$ , les moyennes et variances des courbures principales sont reportés dans le tableau 1.3. Le calcul de  $k_1$  est de mauvaise qualité, et il présente d'assez forte variation. Ceci est la conséquence de la nature des normales que nous utilisons : la triangulation n'est qu'une approximation de la surface réelle, et les normales ne sont pas celles de la surface. Ainsi, elles perturbent l'approximation lorsqu'on leur donne un poids égal aux positions des points dans le critère  $\mathcal{E}$ .

|       | moyenne  | variance |
|-------|----------|----------|
| $k_1$ | 0.125    | 0.263e-3 |
| $k_2$ | 0.526e-2 | 0.317e-4 |

TAB. 1.3: *Moyenne et variance pour la technique avec normale. La courbure théorique  $k_1$  est 0.2 et la courbure  $k_2$  est nulle.*

Pour aller plus loin, nous avons tracé les courbes de la figure 1.12. On remarque immédiatement les pics de variances pour  $\lambda \in [0.01, 100]$ , qui montrent qu'en fonction de la connectique, les normales où les positions jouent un rôle plus ou moins important. Par ailleurs, à partir de  $\lambda = 10^{-3}$  la valeur de  $k_1$  devient très proche de la valeur réelle (l'erreur relative maximale est inférieure à 1%), et la variance est très faible. Ces résultats reflètent la nature même des deux informations que nous traitons :

- d'une part les positions des nœuds qui sont exactes (aux erreurs d'arrondis près),
- d'autre part les normales qui sont une approximation (plus ou moins correcte) des normales réelles.

Nous allons donc dans la suite utiliser la valeur  $\lambda = 10^{-6}$  qui donne la plus grande importance aux positions. L'objectif que nous nous étions fixé est cependant atteint : le calcul en chaque nœud ne s'appuie que sur la première couronne. Ce choix est cohérent avec l'analyse pratique de l'approximation diffuse Hermite : les pseudo-dérivées sont très proches des données. Or nous utilisons des normales très bruitées. Une alternative aurait pu être de moyenner les normales sur les triangles voisins, mais cette technique n'a pas été retenue car elle cache une extension du voisinage d'approximation.

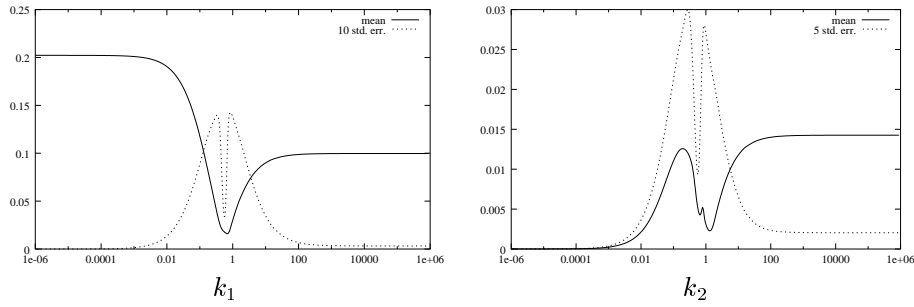


FIG. 1.12: Moyenne et variance de  $k_1$  (la variance est multipliée par 1000) et  $k_2$  (par 100) sur un cylindre en fonction de  $\lambda$ . L'abscisse est logarithmique.

Le cas du cône montre cependant que les bords sont une zone très sensible où certaines connectiques font apparaître des artefacts de courbure. Ces points sont les nœuds pour lesquels on n'a que 6 points de données pour calculer l'approximation diffuse. Comme on ne dispose que de 6 paramètres, la surface calculée est l'interpolation polynomiale des données: les pondérateurs  $w_i$ ,  $w_t$ , et  $\lambda$  ne jouent plus aucun rôle. Pour éviter cela, on peut utiliser en ces points des voisinages étendus (on ajoute par exemple les voisins du deuxième ordre au sens restreint).

L'algorithme de calcul final se présente donc sous la forme suivante pour chaque nœud  $N_0$  de la surface  $\mathcal{S}$ :

- chercher les nœuds de la première couronne
- si c'est un nœud à 6 données (bord) ajouter les voisins d'ordre 2 au sens restreint
- calculer les normales aux triangles dont  $N_0$  est un sommet
- calculer les centres des cercles inscrits
- calculer le plan des moindres carrés des voisins et des centres des triangles: son repère  $(N_0, \vec{u}, \vec{v})$  est complété par le vecteur  $\vec{u} \times \vec{v}$
- calculer l'interpolation diffuse dans ce repère, ainsi que  $G$  et  $L$ ,
- calculer les courbures principales, leurs directions.

### 1.3.6 Robustesse de l'algorithme avec normales

Afin de tester la robustesse de notre algorithme d'un point de vue expérimental, nous avons calculé la courbure sur le second maillage test (triangles de tailles non-homogènes). Le tableau 1.4 montre les statistiques pour des calculs basés uniquement sur la première couronne de nœuds, et sur le voisinage restreint. Les résultats sont à première vue catastrophiques comparés au cas d'un maillage régulier !!

Pour comprendre la cause de ces mauvais résultats, nous allons tracer les courbes d'accumulation pour les différentes grandeurs étudiées (figure 1.15), ainsi que  $k_1$  et  $k_2$  sur le cylindre (figure 1.14).



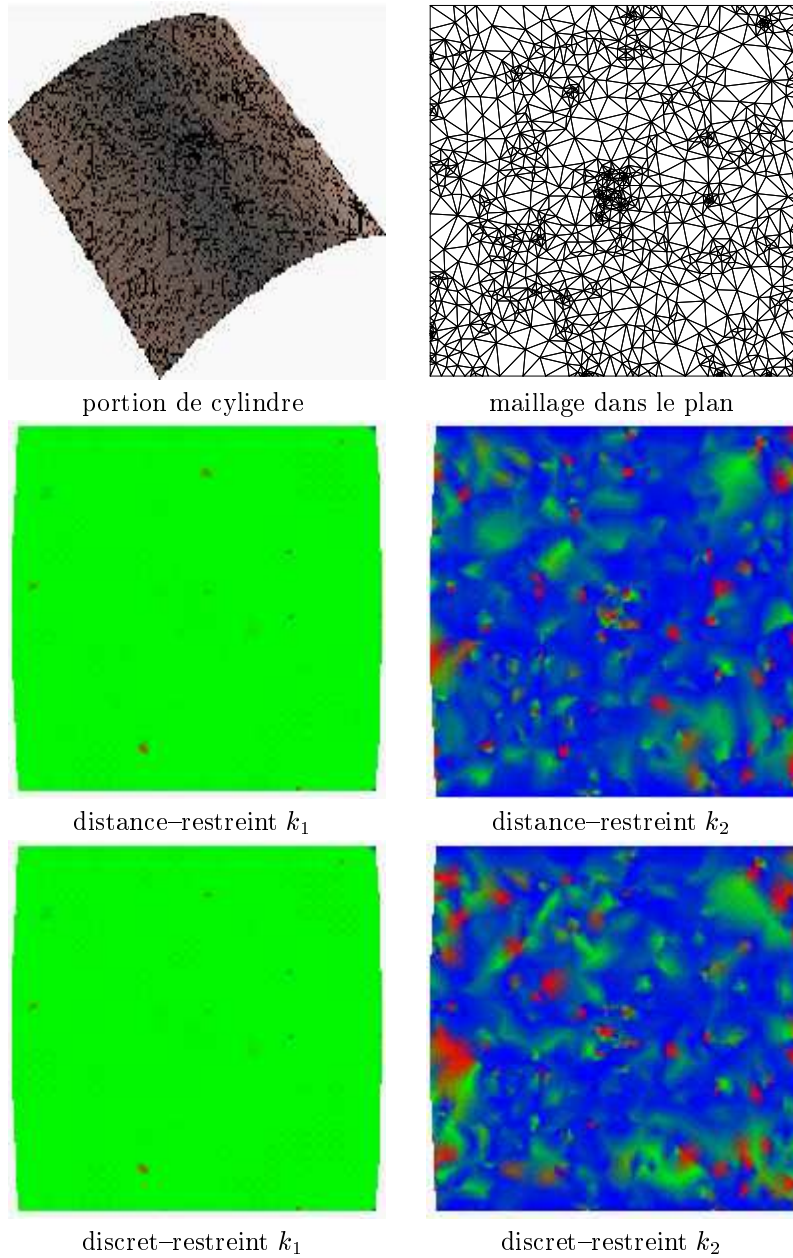


FIG. 1.13: *Maillage hétérogène sur une portion de cylindre. L'échelle de couleur correspond à  $k_1 \in [0.9; 1.1]$  et  $k_2 \in [0; 1e-3]$ . On remarque quelques points rouges sur les figures représentant  $k_1$  qui sont des nœuds de faible connectique. La cartographie de  $k_2$  n'est que peu modifiée par le mode de calcul: on retrouve les mêmes zones à « fort »  $k_2$  sur les deux surfaces.*

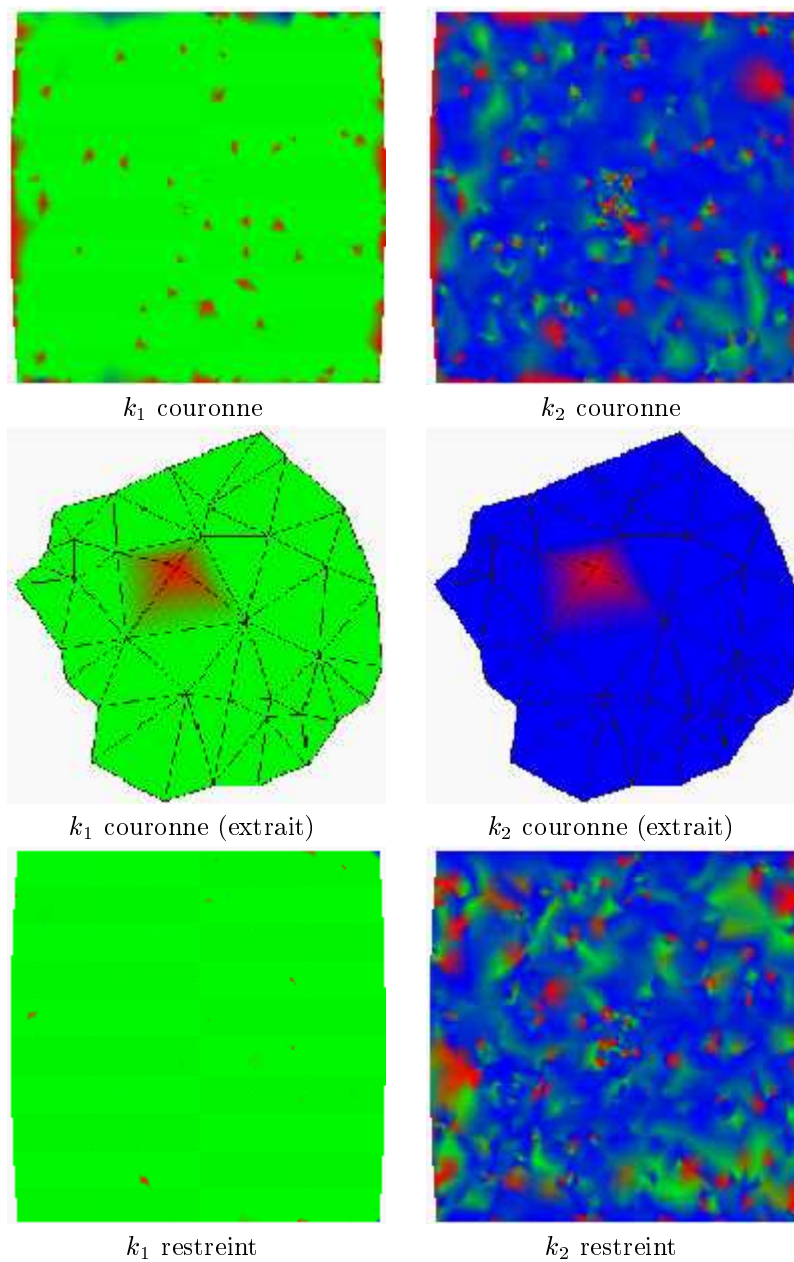


FIG. 1.14: Calcul avec normales pour le cylindre à maillage hétérogène. L'analyse de points de fortes courbures montre que ce sont des nœuds comptant 3 ou 4 connections uniquement.

|       | v. restreint |          | v. couronne |          |
|-------|--------------|----------|-------------|----------|
|       | moyenne      | variance | moyenne     | variance |
| $k_1$ | 1.14         | 4.26     | 6.25        | 4.39e+03 |
| $k_2$ | 0.109        | 2.95     | 0.495       | 7.46     |

TAB. 1.4: Moyennes et variances de  $k_1$  et  $k_2$  pour des calculs avec normales et deux voisinages différents. La courbure  $k_1$  théorique est 1.0, et la courbure  $k_2$  est nulle.

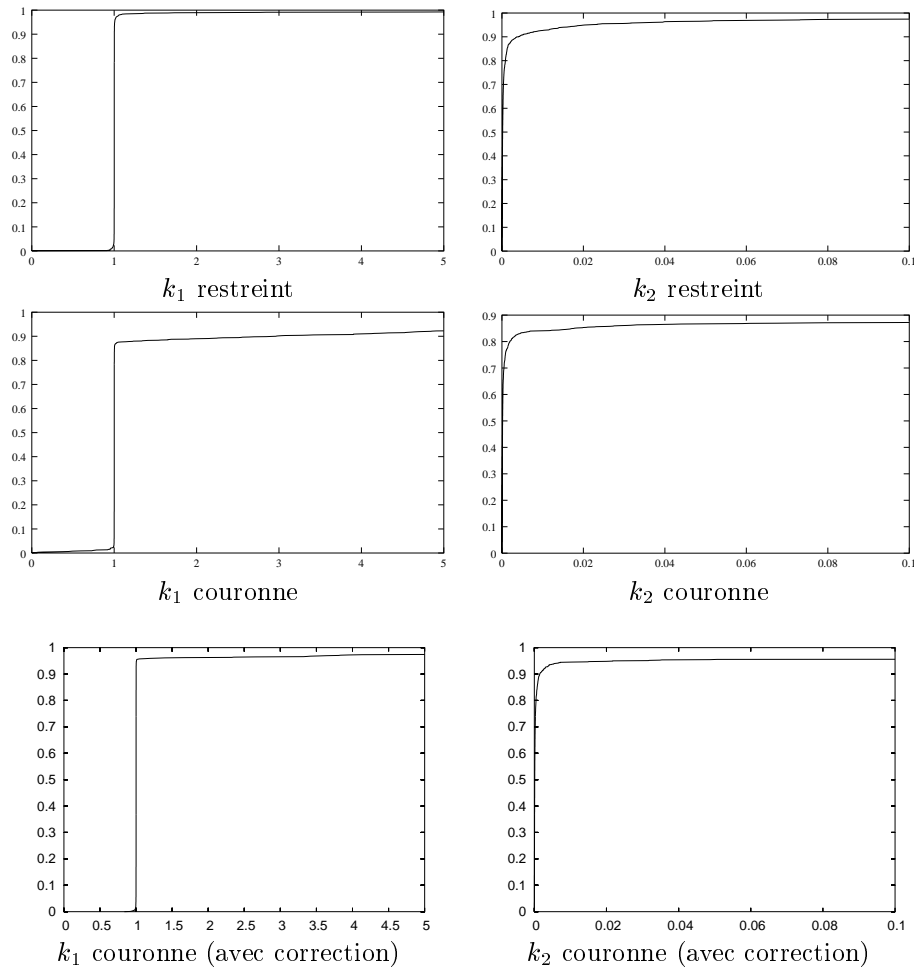


FIG. 1.15: Courbes d'accumulation relatives aux trois types de calculs sur le cylindre test (maillage hétérogène).

La figure 1.14 montre que ce sont en fait les nœuds à 3 et 4 connections qui donnent un biais important à la moyenne. On attribue à ces nœuds des courbures beaucoup trop élevées. Ce type de nœuds apparaît aussi sur notre premier cylindre, mais la position des voisins est très particulière: ils sont alignés selon les deux directions principales de courbure (« position filtrante »), ce qui efface l'effet de la connectique qui est trop faible. Ceci nous amène donc à une contrainte sur le maillage triangulé (lorsqu'on le contrôle): **il faut limiter les nœuds de connectiques 3 et 4 (et si l'on en génère, on peut toujours les placer selon la position filtrante).**

Cependant, on ne contrôle pas le maillage en général, et on cherche à éviter de modifier les données qui sont les positions des nœuds et la connectique. Nous avons testé une technique qui consiste à ajouter au moins un voisin du second ordre au voisinage des nœuds trois et quatre connectés. Ceci permet de retrouver une bonne estimation de courbure en ces nœuds comme le montre les résultats figure 1.4. Certains nœuds de ce type gardent dans ce cas d'un maillages (très) hétérogène une sur-estimation de la courbure, mais ils sont suffisamment isolés et peu nombreux pour qu'on leur applique des traitements *a posteriori*. Nous verrons plus loin l'importance des voisins du second ordre pour la qualité du calcul de courbure près des frontières (voir figure 1.18 par exemple): on peut ajouter un voisin à la fois et calculer une moyenne où les valeurs trop éloignées ne sont pas prises en compte.

La méthode de calcul de la courbure que nous avons présentée permet d'obtenir une bonne estimation des courbures principales. Nous avons limité le calcul aux nœuds du maillage; il est possible cependant d'étendre la méthode aux éléments dans une formulation « duale ».

### 1.3.7 Calcul « quasi-dual »

#### Description

Nous cherchons maintenant à calculer les courbures associées aux triangles, et ce indépendamment du calcul précédent. Ceci pose plusieurs problèmes<sup>3</sup>:

- en quels points évaluons-nous la courbure pour chaque triangle?
- quelles sont les données, le voisinage de calcul?
- quels sont les pondérateurs  $w_i(t)$ ?

Nous allons évaluer les courbures associées à un triangle  $t$  de sommets  $S_1^t, S_2^t$ , et  $S_3^t$  en son centre inscrit (pour les mêmes raisons que précédemment). Le voisinage de calcul est composé des sommets de tous les triangles contenant  $S_1^t, S_2^t$  ou  $S_3^t$ .

Les pondérateurs sont calculés à partir des connectivités: c'est-à-dire que les sommets de  $t$  ont le même poids  $w^1$ , les nœuds connectés à deux sommets ont un poids  $w^2$ , les autres nœuds un troisième  $w^3$ .

Nous avons par ailleurs défini une approximation de type pseudo-Hermite (les gradients ne sont pas connus aux points de positions dans le cas précédent). Nous allons donc utiliser ce même calcul, en nous appuyant sur

- les positions des nœuds,
- les normales, supposées connues aux nœuds.

Ceci pose le problème de calcul des normales aux nœuds qui en pratique se fera par moyennage (sur tous les triangles contenant le nœud par exemple). Nous obtenons

3. Notons que ces problèmes ont été résolus pour l'algorithme précédent.

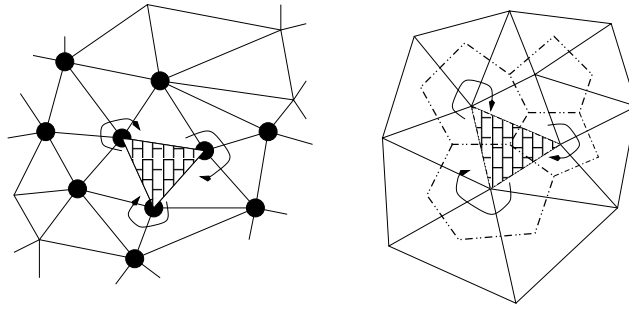


FIG. 1.16: Calcul du voisinage pour l'algorithme de calcul des courbures associées aux triangles. Triangulation et son graphe dual: les nœuds utilisés pour le calcul au triangle  $t$  sont les sommets des triangles définissant une cellule (tireté) dont  $t$  est un sommet.

maintenant une interpolation de type Hermite « pure », car nous connaissons les gradients aux points de positions. Ceci nous permettra, si c'est nécessaire de calculer une interpolation des positions **et** des gradients, ce qui n'était pas possible précédemment (on ne pouvait pas interpoler les normales). Cette technique a été proposée par Alain Rassineux *et al.* dans [RSSV00].

#### « Quasi-dualité »

L'algorithme que nous venons de décrire est presque le dual de l'algorithme de calcul aux nœuds. En effet, le graphe dual de celui défini par le maillage admet comme nœuds les triangles et les nœuds géométriques correspondent aux polygones de ce graphe. Le tableau 1.5 montre les divers éléments en dualité pour les deux algorithmes. Notons que les voisinages de calcul ne sont pas évalués à partir des simples nœuds (au sens des graphes) connectés, mais peuvent aussi l'être avec les nœuds connectés aux nœuds connectés.

|               | Calc. aux nœuds | Calc. sur les triangles |
|---------------|-----------------|-------------------------|
| Point d'éval. | nœuds           | triangles               |
| Voisinage     | nœuds connectés | triangles connectés     |
| Positions     | nœuds           | nœuds                   |
| Normales      | triangles       | nœuds                   |

TAB. 1.5: Tableau comparatif des méthodes d'évaluation de la courbure.

La dualité n'est pas « parfaite » car les positions sont utilisées dans les deux cas aux nœuds géométriques. On pourrait utiliser les positions des centres inscrits des triangles dans l'algorithme d'évaluation aux nœuds, mais ceci reviendrait à utiliser le maillage comme polyèdre caractéristique de la surface (voir [Léo91]). Ceci justifie donc le terme quasi-dualité.

#### Application du calcul sur les triangles

La courbure associée aux triangles est utile pour des automates de remaillage. En effet, on sait calculer une erreur associée à chaque élément et qui est une fonction de la courbure. Cette erreur est ensuite utilisée pour repositionner les points en cours de remaillage ou pour décider du ou des points à supprimer dans le maillage [RSSV99].

On peut apporter des raffinements algorithmiques à différentes étapes du calcul (calcul des normales aux nœuds, détection d'arêtes vives par méthode géométrique, ...). Nous ne les décrivons cependant pas pour nous pencher plus particulièrement sur le calcul aux nœuds et l'une de ses applications.

## 1.4 Comparaison avec la méthode de Samson

Samson et Mallet propose dans [SM97] une méthode de calcul de la courbure d'une surface triangulée basée sur une interpolation de Bézier pour des applications à la géologie structurale. Nous décrivons rapidement leur méthode, puis nous passons à une comparaison expérimentale de la méthode Samson et de l'approche que nous proposons.

### 1.4.1 Méthode de Samson

La méthode proposée dans [SM97] calcule aussi la courbure aux nœuds  $N_0$  de la surface; elle est basée sur la continuité  $G^2$  assurée par un patch de Bézier cubique. Le calcul se fait en coordonnées locales  $(u, v, w)$ , mais avec l'axe  $w$  parallèle à la normale, ce qui permet de ne calculer que la matrice  $L$  (on a alors  $G = Id$ ). Décrivons maintenant plus en détail les différents éléments de cette méthode.

Le calcul de la normale  $\vec{n}$  en  $N_0$  est un moyennage des normales des triangles dont  $N_0$  est un sommet (soit  $\mathcal{T}(N_0)$  l'ensemble de ces triangles, et  $\vec{n}_t$  la normale définie par le triangle  $t$ )

$$\vec{n} = \frac{\sum_{t \in \mathcal{T}(N_0)} \vec{n}_t}{\left\| \sum_{t \in \mathcal{T}(N_0)} \vec{n}_t \right\|}.$$

Puis les tangentes sont définies de façon à être cohérentes avec la normale  $\vec{n}$ : si  $\vec{a}_{01}$  et  $\vec{a}_{02}$  sont les arêtes au nœud  $N_0$  (vers  $N_1$  et  $N_2$ ), on choisit les directions normales à  $\vec{n}$  dans les plans contenant  $\vec{a}_{01}$  et  $\vec{a}_{02}$

$$T_{0i} = \frac{\{\vec{n} \times a_{0i}\} \times \vec{n}}{\|\{\vec{n} \times a_{0i}\} \times \vec{n}\|}.$$

Le patch de Bézier utilisé est alors défini par le mapping (on pose  $w = 1 - u - v$ , et  $(u, v) \in \{(u, v) \in \mathbb{R}_+^2, u + v \leq 1\}$ )

$$\begin{aligned} p(u, v) = & w^3 N_0 + u^3 N_1 + v^3 N_2 \\ & + 3(uw^2 p_{01} + u^2 w p_{10}) \\ & + 3(vw^2 p_{02} + v^2 w p_{20}) \\ & + 3(vu^2 p_{12} + v^2 u p_{21}) \\ & + 6uvwq \end{aligned}$$

où les nœuds du réseau de contrôle sont définis par les relations

$$\begin{aligned} p_{ij} &= N_i + \frac{1}{3} T_{ij} \\ q &= \frac{1}{3} \{p_{01} + p_{10} + p_{02} + p_{20} + p_{12} + p_{21} - N_0 - N_1 - N_2\}. \end{aligned}$$

Cette approximation assure la cohérence avec la normale  $\vec{n}$  calculée au nœud  $N_0$  (et celles des nœuds  $N_1$  et  $N_2$ ), mais elle est seulement  $G^2$ . L'idée est alors de définir un repère local  $(\vec{u}, \vec{v})$  dans lequel on calcule les patches de Bézier de tous les triangles de  $\mathcal{T}(N_0)$ ,  $\vec{p}_t(\vec{u}, \vec{v})$ . Comme l'expression est aussi valide à l'extérieur du triangle  $t$ , on prolonge l'approximation pour tout  $(\vec{u}, \vec{v})$ , et on moyenne au voisinage

de  $N_0$  pour obtenir une formule unique d'approximation ( $A_t$  est l'aire du triangle  $t$ )

$$p(\bar{u}, \bar{v}) = \frac{\sum_{t \in \mathcal{T}(N_0)} A_t \bar{p}_t(\bar{u}, \bar{v})}{\sum_{t \in \mathcal{T}(N_0)} A_t}.$$

On peut alors évaluer les dérivées nécessaires au calcul de la courbure au nœud  $N_0$ .

Cette méthode revient donc à utiliser une approximation cubique dans un repère local attaché au nœud de calcul. Ainsi exposée, la méthode ne semble pas très différente de la nôtre qui calcule une approximation cubique de la surface par moindres carrés glissants. Ces deux approximations interpolent le nœud  $N_0$ , mais on n'a pas d'autres éléments de comparaison a priori. Nous allons donc comparer expérimentalement les deux méthodes.

### 1.4.2 Comparaison expérimentale

Nous comparons les résultats issus des calculs de courbures par la méthode de Samson et par approximation diffuse. Les calculs par la méthode de Samson ont été effectués avec le logiciel *gocad* (version 1.3.5) qui intègre le code développé par son auteur.

Le tableau 1.6 regroupe les données statistiques des calculs sur les deux cylindres, et la figure 1.17 montre les courbes d'accumulation associées.

|          | Cas régulier |         | Cas non-reg. |         |
|----------|--------------|---------|--------------|---------|
|          | Samson       | diffus  | Samson       | diffus  |
| $m(k_1)$ | 0.190        | 0.202   | 2.31         | 6.25    |
| $v(k_1)$ | 3.11e-3      | 2.5e-10 | 295.         | 4.39e+3 |
| $m(k_2)$ | 4.98e-3      | 3.91e-4 | 1.20         | 0.495   |
| $v(k_2)$ | 3.83e-4      | 1.74e-6 | 295.         | 7.46    |

TAB. 1.6: *Données statistiques comparées pour les algorithmes de Samson et diffus ( $k_1=0.2$ ,  $k_2=0$ .)*

Le tableau des données statistiques montre que l'algorithme diffus donne de meilleurs résultats sur le maillage régulier. La moyenne de  $k_1$  sur le second cas donne cependant un très mauvais résultat pour l'algorithme diffus comme nous l'avons déjà vu, alors que l'algorithme de Samson semble plus robuste. Cette première analyse simpliste ne résiste pas à l'analyse de la variance expérimentale: les courbes d'accumulation montrent en effet que de très nombreux points ont une valeur correcte (environ 85%) et la valeur moyenne est tirée vers le haut par les nœuds de faible connectique, ce qui explique la variance très élevée. Par ailleurs, la variance plus faible de l'algorithme de Samson ne montre pas l'étalement important des valeurs de  $k_1$ .

Les courbes d'accumulation nous montrent que l'algorithme diffus fournit une estimation de meilleure qualité et surtout plus homogène lorsqu'on sait traiter les cas anormaux. L'étalement des valeurs avec la méthode de Samson rend difficile l'utilisation de la courbure pour caractériser les surfaces car deux surfaces de courbures voisines risquent d'être confondues.

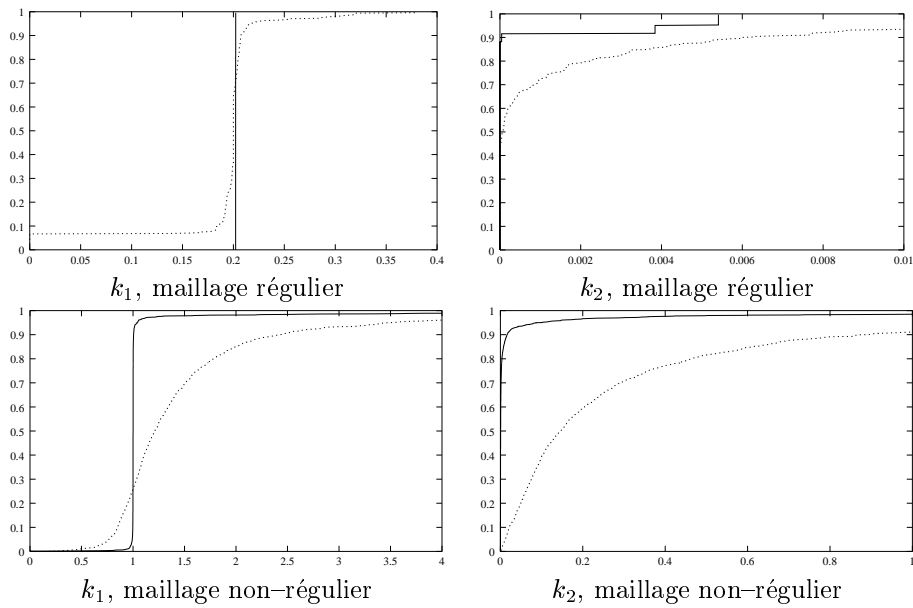


FIG. 1.17: Courbes d'accumulation comparées des algorithmes de Samson et diffus restreint (la courbe continue est le cas diffus)

## 1.5 Conclusion

En conclusion nous avons appliqué l'approximation diffuse Hermite au calcul de courbure des surfaces triangulées. Ce travail est une extension et une justification de la méthode d'ajustement utilisée usuellement. En effet, l'arrière plan théorique (théorème de convergence en particulier) de l'approximation diffuse Hermite permet de justifier l'approximation de dérivée partielle d'une surface par une surface quadratique tangente (notion de pseudo-dérivée); et l'introduction des normales (ici les normales aux faces) est un raffinement de la méthode. On pourrait imaginer des prétraitements pour calculer ces normales lorsqu'on traite des nuages de points.

Nous avons donné les limites d'application de cette technique sur un cas expérimental (cas des nœuds 3 et 4 connectés). Les problèmes rencontrés peuvent être facilement contournés (par extension du voisinage, voir page 107). Nous montrons aussi que la classe des méthodes ajustement / approximation diffuse est plus robuste sur les surfaces triangulées que les méthodes basées sur des patches, trop sensibles à la triangulation.

L'algorithme proposé donne une bonne approximation de la courbure en utilisant un voisinage minimal, on peut donc dire que c'est une méthode « haute résolution ». La figure suivante montre combien l'effet du voisinage est important sur des surfaces issues de modèle CAO. Cette propriété nous sera très utile dans la suite pour construire l'algorithme de reconnaissance des surfaces.



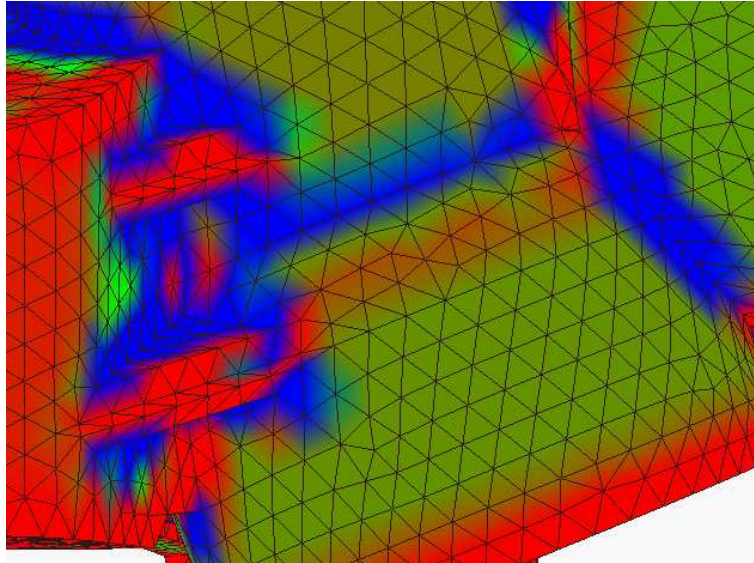
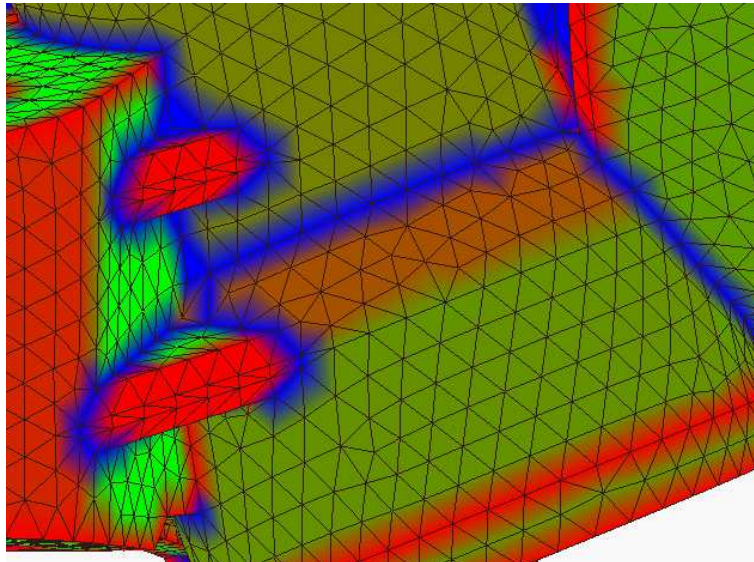
Courbure  $k_1$  par ajustementCourbure  $k_1$  avec normale

FIG. 1.18: Amélioration du calcul des courbures dans le cas de surface complexes.

## Chapitre 2

# Reconnaissance des surfaces

Dans le chapitre précédent nous avons défini une méthode de calcul de la courbure s'appuyant sur l'approximation diffuse Hermite. Nous avons montré qu'elle est performante et donné ses limites d'application. Le but de cette partie est de montrer son intérêt en utilisant les courbures calculées pour un algorithme de reconnaissance des surfaces. La haute résolution du calcul initial est primordiale car elle permet la détection des courbes d'intersection.

Dans un premier temps, nous donnons le cadre du travail avec de rapides repères bibliographiques. L'algorithme de reconnaissance des surfaces est ensuite présenté dans un cadre théorique fort (avec des hypothèses de travail restrictives), ce qui nous permet d'en démontrer quelques propriétés. La robustesse est ensuite abordée par la présentation de cas pratiques.

## 2.1 Courbure et reconnaissance des surfaces

### 2.1.1 Le Graphe Global des Courbures

Nous exposons ici rapidement deux éléments développés dans [Gou97], on se reportera à ce document pour plus de détails. Ces deux notions sont :

- Le Graphe Global des Courbures (GGC), d'une part ;
- Les centres de courbures, d'autre part.

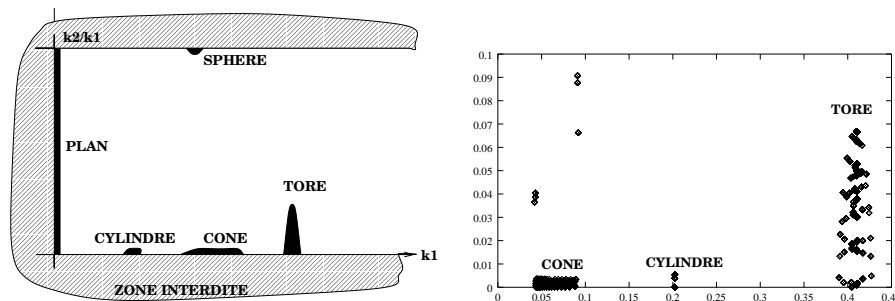


FIG. 2.1: Position théorique des primitives de base dans le GGC. Cas réel d'un cylindre, d'un cône et d'un tore.

Le graphe global des courbures est composé des courbures des points d'évaluation dans le plan  $(k_1, \frac{k_2}{k_1})$ . Il combine à la fois une grandeur dimensionnelle, et une grandeur de forme, et présente l'avantage d'avoir une expression simple. Les primitives

usuelles de la CAO ont des projections caractéristiques dans le graphe global des courbures comme le montre la figure 2.1. Notons qu'il peut être intéressant de transformer  $k_1$  en  $t(k_1) = k_1 - \frac{1}{k_1}$  qui permet de mieux séparer les primitives dont la courbure maximale est dans  $]0, 1[$  (car la fonction  $f(x) = x - 1/x$  est une bijection de  $\mathbb{R}^{+*}$  vers  $\mathbb{R}$ , et le segment  $]0, 1[$  est transformé en  $\mathbb{R}^{-*}$ ).

Le centre de courbure associée à un point  $p$  d'une surface  $\mathcal{C}^2$  est le translaté de  $p$  par  $\frac{1}{k_1} \vec{n}_p$ . Cette notion n'a pas de sens pour un plan (les centres de courbures sont à l'infini). Par cette définition, les centres de courbures des primitives simples ont une position très caractéristique : par exemple, le lieu des centres de courbure d'un cylindre est son axe.

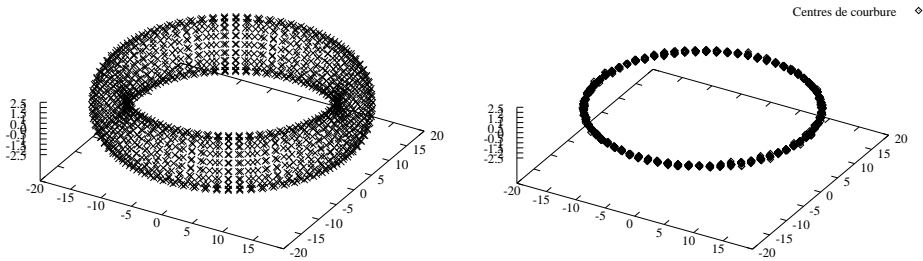


FIG. 2.2: Noeuds définissant un tore. Centres de courbure associés.

## 2.1.2 Bibliographie

La plupart des publications relatives au *reverse engineering* concerne l'analyse de cartes de profondeurs ou de nuages de points 3d. Nous n'avons pas trouvé d'articles s'appuyant sur des surfaces triangulées.

L'article d'introduction de Besl et Jain [BJ85] est une bonne référence sur le sujet. Une étude bibliographique plus récente est proposée par Varady *et al.* dans [VMC97].

On distingue principalement deux types de méthodes pour la reconnaissance des surfaces : les unes s'appuyant sur la courbure, les autres s'appuyant sur des ajustements locaux. La première catégorie se sépare en deux sous-ensembles, qui utilisent les courbures principales (voir par exemple [IW98]), ou les courbures moyenne et de Gauss [TB95]. Goulette propose dans sa thèse [Gou97] l'emploi des centres de courbure qui simplifie le problème pour certaines surfaces (cylindres, tores). Les méthodes de la seconde catégorie utilisent des ajustements de surfaces plus ou moins complexes, puis s'attachent à reconnaître les surfaces ajustées. Le travail de [LGB95] semble le plus prometteur : il mixe des ajustements locaux, avec des stratégies de découpage et de regroupement.

Notre méthode se place dans la première catégorie, car l'algorithme de calcul de la courbure que nous utilisons admet une haute résolution. Il utilise aussi une stratégie de regroupement basée sur le maillage. Notre donnée de base est bien différente du cas traité usuellement dans la littérature : les positions ne sont pas bruitées et le maillage décrit une topologie de la surface. Nous allons tirer partie de ces conditions idéales pour montrer les propriétés de l'algorithme et ses limites.

Ce travail s'inscrit donc plus dans le retour d'une surface triangulée vers de la CAO que dans le cadre « classique » du *reverse engineering*.

## 2.2 Présentation de l'algorithme proposé

### 2.2.1 Hypothèse de travail/ Présentation générale

#### Hypothèse de travail

Nous allons présenter l'algorithme de reconstruction dans le cadre d'hypothèses fortes qui nous permettent de justifier chaque étape de l'algorithme, puis de démontrer quelques propriétés dans ce cadre. Ces hypothèses seront discutées dans la partie suivante où l'on présente des cas réels.

**Hypothèse 4** *Hypothèses relatives à la surface et sa discrétisation :*

**H1** *Nous supposons que les surfaces sont bien discrétisées, i.e. l'ensemble des triangles s'appuyant sur des nœuds intérieurs à une surface élémentaire est égal à cette surface élémentaire (au sens de la triangulation).*

**H2** *Nous considérons une surface dont chaque portion est une partie de PLAN, CYLINDRE, CONE, TORE ou SPHERE. Les intersections entre deux portions sont contenues dans le maillage c'est-à-dire qu'elles sont représentées par une chaîne d'arêtes du maillage.*

L'hypothèse H1 est très restrictive (voir figure 2.3), mais elle permet de simplifier les démonstrations.

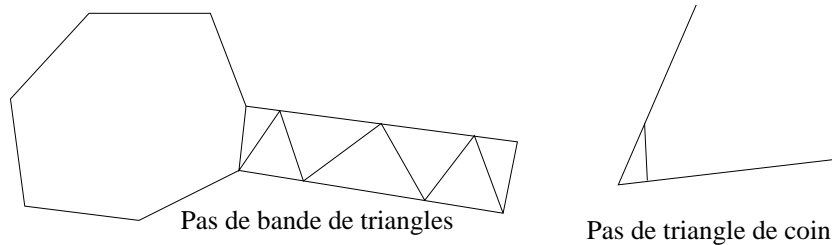


FIG. 2.3: Cas de maillage de surface élémentaire interdit par H1.

H2 implique deux propriétés importantes pour la suite :

- (i) Un nœud appartient
  - soit à l'intérieur d'une portion (nœud dit « intérieur »),
  - soit à l'intersection de deux (ou plusieurs) portions (nœud dit « de frontière »).
- (ii) Dans le cas d'un nœud intérieur, le calcul diffus estime la courbure de la portion (grâce à sa haute résolution).

La troisième hypothèse (H3) est la plus forte :

**Hypothèse 5** *H3 On suppose que le calcul de la courbure est exact en chaque point intérieur.*

**Remarque 8** *On ne peut pas faire l'hypothèse H3 aux nœuds de frontière, car la courbure n'y est pas définie.*

### Présentation générale de l'algorithme

La reconnaissance des surfaces se fait par type: un nœud est reconnu comme appartenant à l'intérieur de l'un des cinq types de base. Lorsqu'un nœud est ainsi classé, il est dit « intérieur », et permet de construire de nouvelles informations, qui sont reportées sur les triangles, arêtes et nœuds voisins. On construit nécessairement des composantes connexes de même type (hypothèse H2).

Le processus de reconnaissance des surfaces suit donc trois étapes distinctes :

1. Classement initial des nœuds intérieurs: ce sont les nœuds « sûrs ».
2. Propagation de cette information.
3. Calcul des frontières.

Le classement initial s'appuie sur l'analyse des positions des nœuds dans le graphe global des courbures. Comme on ne peut pas classer tous les nœuds lors de cette étape, on propage l'information en s'appuyant sur la topologie du maillage. Enfin, lorsqu'on ne peut plus classer de nœuds, on calcule les frontières entre les différentes composantes connexes.

Cet algorithme de reconnaissance ne donne pas l'expression du modèle de chaque portion de surface. On sait en effet que l'estimateur de courbure par moindres carrés est biaisé d'après [Gou97], et cette propriété s'étend à l'estimateur diffus, même si le biais est plus faible. Une étape d'ajustement pour chaque portion sera donc nécessaire pour finir la reconnaissance. L'analyse des surfaces ajustées nous permettra de confirmer (ou d'infirmer) le classement que nous avons effectué dans la première étape.

### 2.2.2 Classification initiale

#### Classement ponctuel

Dans le cadre de l'hypothèse H2 (limitation aux cinq types de primitive), il est possible de classer certains nœuds par analyse ponctuelle dans le graphe global des courbures. Pour cela, on applique dans l'ordre les règles suivantes, en s'arrêtant dès que le nœud est classé.

- Si  $k_1 = k_2 = 0$  alors le nœud est de type PLAN.
- Si  $k_2 \neq 0$  et  $\frac{k_2}{k_1} = 1$  alors le nœud est de type SPHERE.
- Si  $k_2 \neq 0$  alors le nœud est de type TORE.
- Si  $k_2 = 0$  alors le nœud est de type CYLINDRE ou CONE ou TORE.

Il n'est pas possible par cette analyse de distinguer les nœuds de la dernière règle, car ce sont les variations de  $k_1$  ou  $k_2$  qui sont différentes: une analyse locale est nécessaire. Pour le TORE, cela concerne les nœuds qui sont situés sur les deux cercles polaires.

#### Classement local

La distinction entre les nœuds de TORE, CYLINDRE et CONE s'appuie sur les variations de  $k_1$  et de sa direction principale en appliquant les règles suivantes comme précédemment (i.e. on arrête lorsque le nœud est classé). Soit  $N$ , un nœud de ce type

- si les voisins de  $N$  ont même  $k_1$  et même directions principales associées, alors il est de type CYLINDRE.

- si les voisins de  $N$  ont même  $k_1$  et des directions principales différentes (au moins une), alors c'est un nœud de type TORE,
- sinon, c'est un nœud de type CONE.

À ce point de l'analyse, on n'a jamais vérifié que l'information construite est pertinente: par exemple, l'intersection entre un cône et un cylindre ressemble à un tore, mais n'en est pas un (conséquence de H1). Ces erreurs sont la conséquence de notre ignorance des frontières entre portions de surfaces. Il est donc nécessaire de vérifier la **cohérence** du modèle.

### Phase de cohérence

La seconde partie de l'hypothèse H2 nous permet d'exprimer une règle de cohérence simple: **les nœuds classés « sûrs » forment des composantes connexes homogènes**. C'est-à-dire que les nœuds classés forment des composantes connexes de même type et cohérentes: par exemple une composante de type TORE admet des nœuds de  $k_1$  identiques. Nous allons donc parcourir tous les couples de nœuds pour vérifier que leurs classements sont cohérents; ceci assure la cohérence des composantes connexes.

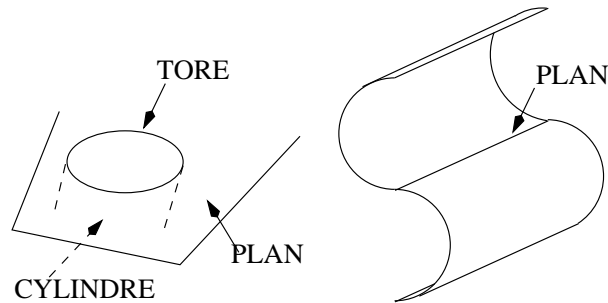


FIG. 2.4: Type de problème traité lors de l'analyse de cohérence. À gauche, une erreur du classement ponctuel, à droite problème posé par le classement PLAN

La première règle que nous appliquons concerne les couples de nœuds de types différents qui ne respectent pas l'hypothèse H2. Toutes les associations de types ne sont pas possibles: en effet, l'analyse locale implique qu'un nœud de type CYLINDRE est

- soit connecté à un nœud de type CYLINDRE et de même  $k_1$  que lui,
- soit connecté à un nœud de type CONE de même  $k_1$  que lui.

Dans le second cas, on sait que le nœud de type CONE est intérieur à la portion de surface et connecté à la frontière de cette portion. Malheureusement, on peut trouver des associations de tous les autres types, par exemple, l'ellipse d'intersection d'un cylindre et d'un plan et classée TORE dans la phase d'analyse ponctuelle, et fait apparaître des connections PLAN–TORE et TORE–CONE (figure 2.4).

Par ailleurs, le classement PLAN est presque toujours correct. La figure 2.4 montre le cas de nœud de courbure  $k_1 = k_2 = 0$  qui n'est pas sur un plan. Ce cas nous invite à regarder tous les raccords  $C^1$  des cinq primitives que nous considérons. La figure 2.5 montre la plupart des cas particuliers de ce type d'intersection. Ces classements erronés ont une connectique très particulière; ce qui permet de les détecter: nous supposons donc dans la suite que les nœuds de type PLAN sont

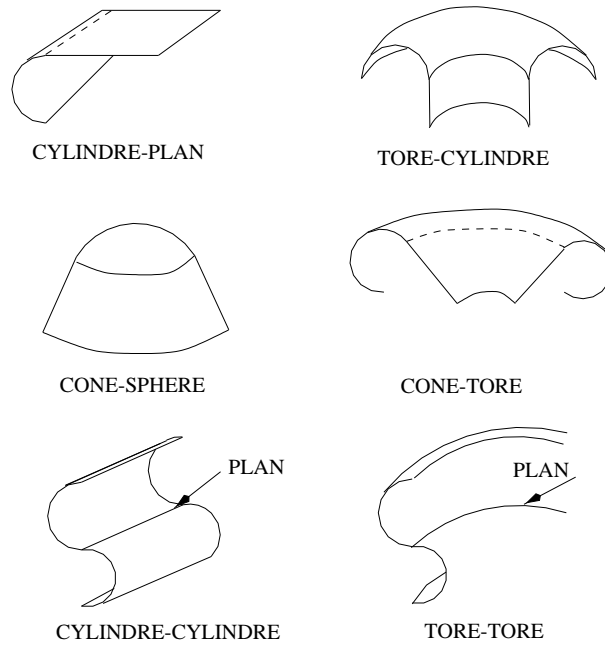


FIG. 2.5: Cas particuliers de connexions  $C^1$  entre primitives.

fiables. Ainsi la première règle de cohérence s'écrit :

Si deux nœuds connectés sont de types différents, alors

- si l'un d'eux est de type PLAN, déclasser l'autre,
- sinon
  - si l'un d'eux est de type CYLINDRE, déclasser l'autre
  - sinon déclasser les deux.

Cet algorithme de cohérence appelle deux commentaires :

1. Il s'appuie sur la qualité croissante d'estimation de la courbure dans les cas pratiques.
2. L'ordre de traitement n'a pas d'influence grâce à l'hypothèse H1.

Il nous reste maintenant à traiter le cas des nœuds connectés de même type. Pour les mêmes raisons que précédemment, deux nœuds de type CYLINDRE ou PLAN connectés forment toujours un couple cohérent.

Les couples TORE-TORE et SPHERE-SPHERE vont être traités par les critères suivants :

**TORE** on teste l'égalité des  $k_1$ .

**SPHERE** on teste l'égalité des  $k_1$  (et donc des  $k_2$ ).

Nous avons maintenant des composantes connexes de nœuds connectés et cohérents. Nous allons donc propager l'information sur le maillage pour reclasser les nœuds qui ne le sont pas. Comme nous l'avons vu, l'ensemble des éléments développés ici s'appuie sur les hypothèses H2 et H3. Nous pensons que tous les nœuds classés le sont correctement :

**Conjecture 1** *A l'issue de la phase de cohérence, les nœuds classés sont correctement classés.*

### 2.2.3 Propagation

Le classement actuel ne concerne que les nœuds du maillage. La phase de propagation va aussi classer les triangles et les arêtes. Ces éléments peuvent sembler n'être que des intermédiaires commodes à l'exposé, mais ils sont en réalité essentiels : ils décrivent la topologie de la surface.

#### Opérateur(s) topologique(s)

Les opérateurs topologiques traduisent le transport du classement « intérieur » d'un nœud sur les éléments qui lui sont connectés. La figure 2.6 montre le cas d'un triangle dont un nœud est classé « intérieur » et les informations construites à partir de ce nœud. Nous préférons par souci de clarté séparer cet opérateur unique en quatre opérateurs élémentaires<sup>1</sup>:

**N2T\_int** : un triangle (surface intérieur) contenant un sommet classé « intérieur » a le même type « intérieur ».

**N2E\_int** : une arête dont les deux extrémités sont classées a ce type (qui est le même par hypothèse et vérification).

**T2N\_app** : Les nœuds (non-classés « intérieur ») d'un triangle « intérieur » appartiennent au moins à une portion de surface du type du triangle ( « app » )

**T2E\_app** : Les arêtes (non-classées « intérieures » ) d'un triangle « intérieur » appartiennent au moins à un patch de ce type.

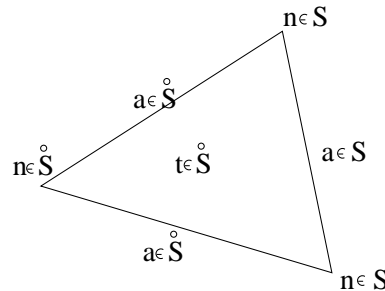


FIG. 2.6: Si le nœud de gauche appartient à l'intérieur  $\overset{\circ}{S}$  d'une surface  $S$ , alors l'hypothèse H2 nous permet d'exprimer les autres relations de la figure.

Les opérateurs ci-dessus sont appliqués aux triangles et aux arêtes non-classés dans l'ordre de la description : tous les triangles sont traités par **N2T\_int**, puis toutes les arêtes par **N2E\_int**, ... ceci permet d'assurer que toute l'information issue des nœuds sûrs est prise en compte.

Cet opérateur est valide car nous avons supposé que les intersections entre primitives sont décrites par des arêtes de la triangulation (H2).

1. La convention de dénomination est la suivante : **N2T\_int** signifie : transport de l'information portée par un **N**œud vers un **T**riangle pour construire une information **sur**. La lettre **E** (pour edge) représente les arêtes et le suffixe **\_app** signifie appartenance à la surface. Rappelons qu'un nœud « intérieur » appartient à l'intérieur  $\overset{\circ}{S}$  (au sens topologique) d'une surface  $S$ .



### Opérateurs de classement

Les opérateurs précédents transmettent une information « dégradée » (mais l'information globale augmente) : dans la première étape, on classe les nœuds à l'intérieur de chaque portion au sens topologique, alors que nous n'obtenons ici qu'une relation ensembliste (i.e. un nœud appartient ou non à une surface). Ceci est imposé par la nature même des traitements topologiques. L'objectif est maintenant de transformer cette information en information « intérieur », ou nœud « de frontière ».

Nous distinguons deux opérateurs dont le but est différent :

**TT2EN\_frt** : qui traite l'arête commune à deux triangles, ainsi que les nœuds communs, son but est de détecter les frontières.

**app2N\_int** : s'appuie sur la liste des portions de surface auxquels appartient un nœud pour le classer définitivement. Cet opérateur s'applique dès que la liste est non vide.

Dans le cadre de l'hypothèse  $H3$  (calcul exact), l'opérateur **TT2EN\_frt** distingue deux cas : si les deux triangles sont classés, alors l'arête commune et les nœuds communs sont sur une frontière, sinon, on ne peut rien faire. Nous modifions cet opérateur, ce qui nous permettra de démontrer les propriétés de l'algorithme.

**TT2EN\_frt** distingue donc trois cas :

- L'un des triangles n'est pas classé, ne rien faire.
- Les deux triangles sont du même type
  - si l'un des nœuds de l'arête est classé frontière, classer l'arête et le nœud frontière<sup>2</sup>,
  - sinon, ne rien faire.
- Les deux triangles sont de types différents, classer l'arête et les nœuds « de frontière ».

L'opérateur **app2N\_int** cherche à classer intérieurs les nœuds appartenant à une portion (propriété issue de l'opérateur **T2N\_app**). Pour cela, on gère la liste des appartenances, et l'opérateur distingue deux cas :

1. Il existe (au moins) deux types différents dans la liste d'appartenance, le nœud appartient à une intersection de primitives.
2. Toutes les appartenances sont du même type. Il faut alors vérifier s'il s'agit d'un nœud intérieur ou de frontière.

Le second cas est traité par comparaison du nœud et de ses voisins classés intérieurs dans le GGC, c'est le seul opérateur de la phase de propagation pour lequel on utilise la courbure. Dans le cas où le nœud est sur une frontière, on peut se contenter de l'ôter de la liste des nœuds à traiter, car c'est un nœud terminal ; c'est-à-dire qu'on ne construit plus d'information à partir d'un tel nœud. En effet, dans cette phase, toute l'information est construite à partir des nœuds intérieurs. Nous verrons plus tard que ceci assure la convergence de l'algorithme dans le cadre théorique. Il semble aussi plus naturel de construire les frontières à partir des arêtes (**TT2EN\_frt**).

La phase de propagation consiste alors à itérer l'(es) opérateur(s) topologique(s) puis les opérateurs de classements, tant que l'on peut classer de nouveaux nœuds intérieurs.

---

2. Deux triangles du même type n'appartiennent pas forcément à la même portion de surface

### 2.2.4 Post-traitements

Lors de la phase de cohérence, on détruit parfois tous les nœuds intérieurs d'une portion. Comme le montre la figure 2.7.

Ce type de portion peut être traité a posteriori, en recommençant le cycle d'analyse où seuls les nœuds non-connectés à des intérieurs sont traités dans la première phase. Ceci permet ainsi de finir l'analyse des nœuds intérieurs.

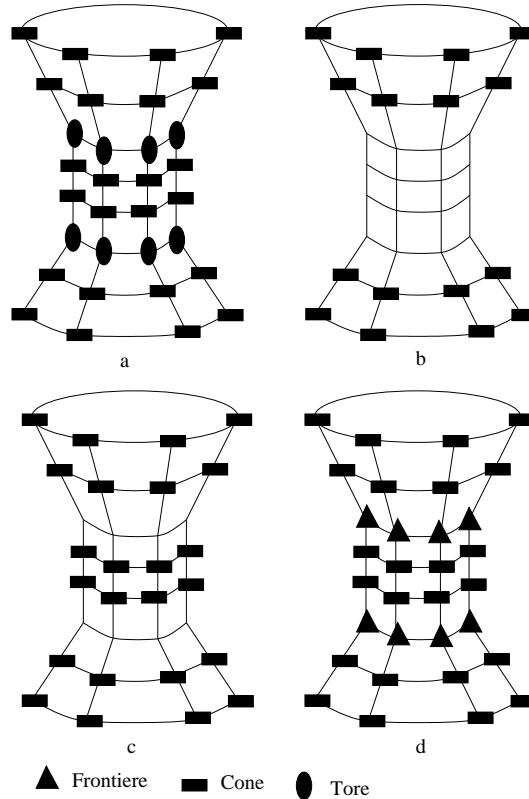


FIG. 2.7: De gauche à droite et de haut en bas : classement initial (a), classement après la phase de cohérence (b), classement des nœuds non connectés (c), classement final (d).

Nous avons maintenant une chaîne de traitement complète. Nous allons en décrire quelques aspects théoriques dans le cadre de nos hypothèses (H1,H2,H3), avant de passer aux aspects pratiques.

### 2.2.5 Démonstration de l'algorithme

La figure 2.8 donne le synopsis de l'algorithme de reconnaissance construit à partir des phases élémentaires que nous venons de décrire. Nous appelons germe chaque composante connexe de l'ensemble de nœuds classés intérieurs à l'issue de la phase de cohérence. Dans le cadre de l'hypothèse H3, on compte un ou aucun (cas des portions « oubliées ») germe par portion. Pour traiter les portions sans germe, nous reprenons les phases de traitement. Nous pensons qu'il existe des cas (en théorie) où **une** itération n'est pas suffisante. Nous considérons donc dans la suite l'ensemble des portions contenant un germe à l'étape  $i$ ,  $S_i$ . L'étude de la phase de classement initiale présente peu d'intérêt sous l'hypothèse H3 (on classe

tous les nœuds correctement à l'exception des frontières et des portions oubliées). La convergence de la phase itérative est démontrée en deux étapes :

1. Convergence de l'itération des opérateurs de propagation et classement.
2. Calcul du nombre de nœuds classés.

On suppose de plus que l'itération sur les surfaces  $S_i$  est convergente.

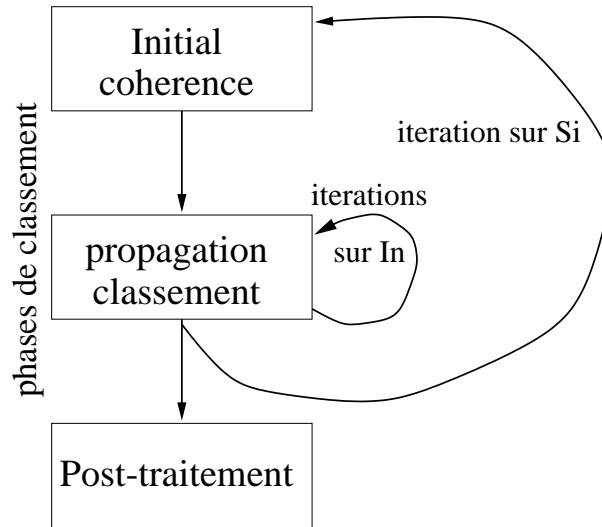


FIG. 2.8: *Synopsis de l'algorithme de convergence*

### Convergence de la phase itérative

La convergence de l'algorithme de reconnaissance se démontre en étudiant le cardinal de l'ensemble des nœuds classés intérieurs à une portion. Soit  $I_n$  le terme général de cette suite définie comme le nombre de nœuds classés intérieurs à l'issue de l'itération  $n$  et  $I_0$  comme le nombre de nœuds classés à l'issue du classement initial et de la phase de cohérence. Une itération est l'application en cascade des opérateurs topologiques et des opérateurs de classement.

**Propriété 34** *La suite  $(I_n)$  admet les propriétés suivantes :*

- (i)  $(I_n)_n$  est croissante car on ne décline jamais de nœuds dans cette phase ;
- (ii) Elle est bornée par le nombre de nœuds total ;
- (iii) Elle est donc convergente (d'après (i) et (ii)) ;
- (iv) Dès que  $I_{n+1} = I_n$ , la limite est atteinte.

Cette dernière propriété est vraie, car on construit toute l'information possible à partir des nœuds intérieurs par les opérateurs topologiques. Ainsi, l'algorithme est convergent, et la limite est atteinte dès que la suite  $I_n$  est stationnaire.

### Nombre théorique de nœuds classés

Nous cherchons à démontrer que sous les hypothèses, (H1,H2,H3), l'algorithme de reconnaissance sait reconnaître tous les nœuds de  $S_i$ . Pour cela, nous allons démontrer deux propriétés des opérateurs de propagation.

**Propriété 35** *Les opérateurs de propagation admettent les propriétés suivantes :*

1. *Si une portion contient un germe, alors*
  - tous les nœuds intérieurs sont reconnus.
  - seulement les nœuds intérieurs sont reconnus.
2. *Les frontières entre deux portions (i.e. germes) sont détectées :*
  - les nœuds sont classés frontière.
  - les arêtes sont classées frontière.

La première propriété s'appuie sur

- l'hypothèse H3: un nœud intérieur d'une portion connecté à un nœud classé intérieur sera classé par l'opérateur `app2N_int`;
- la définition d'une portion, qui est une composante connexe.

Il existe donc sur la triangulation un chemin entre tout couple de nœuds dans la portion (hypothèse H1). Ainsi, le germe peut être réduit à un nœud unique.

La seconde partie de la première propriété s'appuie sur le fait que les intersections de primitives admettent une estimation numérique<sup>3</sup> de  $k_1$  et  $k_2$  différente de la leur. Pour cela, nous devons étudier tous les cas d'intersections entre les cinq primitives. Seuls des cas particuliers mènent à des solutions ambiguës, mais qu'on ne peut pas considérer comme des erreurs. En effet, la description de la surface par une triangulation est trop pauvre pour décider ce genre de cas (voir figure 2.9(a)).

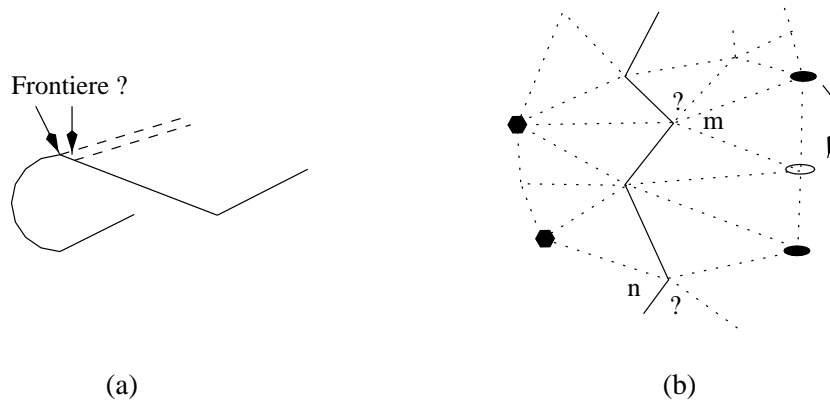


FIG. 2.9: (a) Cas où la frontière ne peut pas être détectée précisément à partir de la triangulation. (b) Frontière entre patch, cas élémentaires.

La seconde propriété se démontre en vérifiant que l'application des opérateurs de classement conduit bien à une définition correcte des frontières. On commence par remarquer que le cas des nœuds  $n$  et  $m$  de la figure 2.9(b) sont représentatifs de

<sup>3</sup> La courbure n'exite sur les intersections de patches que dans des cas particuliers de raccord  $C^2$  de primitives.

tous les cas possibles (i.e. ils posent tous les types de problèmes). Le cas du nœud  $n$  est évident : on est amené à appliquer l'opérateur `TT2EN_frt`. Si les deux patches sont différents, alors on classe l'arête et les nœuds de frontière. Sinon, l'opérateur `app2N_int` classe le nœud de frontière (d'après la propriété précédente), et on classera l'arête au pas suivant. Le cas du nœud  $m$  se ramène au cas précédent par transmission de l'appartenance dans l'intérieur du patch.

Enfin, les portions ne contenant pas de germe, mais des nœuds intérieurs seront classées par itération. Nous proposons la conjecture suivante.

**Conjecture 2** *L'itération sur les surfaces  $S_i$  est convergente et tous les patches sont atteints dès qu'elle est stationnaire.*

Ceci nous permet de donner le théorème de convergence de l'algorithme.

**Théorème 11** *Sous les hypothèses (H1,H2,H3), l'algorithme de reconnaissance sait classer tous les nœuds et tous les triangles d'une surface triangulée.*

Le cas des triangles est une conséquence immédiate de l'hypothèse H1.

**Remarque 9** *Le théorème n'assure pas que la classification est « vraie », mais cohérente avec les courbures calculées par l'approximation diffuse Hermite.*

## 2.3 Robustesse, généralisation

Notons que seule l'hypothèse H3 ne peut jamais être vérifiée, alors que H1 et H2 peuvent être des contraintes lors de la construction des surfaces triangulées (cas des modèles CAO). Nous allons montrer à partir d'exemples, que le calcul diffus de la courbure nous permet de nous passer de H3 pour une hypothèse H3' du type : les courbures des primitives sont « assez bien séparées par le calcul diffus » pour les reconnaître entre elles. Nous allons voir que la résolution très fine du calcul diffus joue aussi un très grand rôle.

### 2.3.1 Classement initial (discussion de H3)

Le classement initial s'appuie principalement sur l'hypothèse H3, qui signifie que pour chaque nœud intérieur on estime bien la courbure de la primitive, et ceci de façon exacte.

Le calcul diffus répond aux deux aspects de H3 en s'appuyant sur l'hypothèse H2. En effet, le voisinage utilisé pour le calcul de la courbure ne prend en compte pour un nœud intérieur que des données relatives à « sa » primitive (voisins du premier ordre, et normale des triangles dont il est le sommet). C'est l'emploi des normales qui permet d'obtenir une estimation de la courbure haute résolution, c'est-à-dire avec une meilleure localisation que les méthodes usuelles qui s'appuient sur des voisinages plus grands comme le montre le chapitre précédent. Ceci assure donc la première partie de H3. Bien entendu, on ne peut pas prétendre au calcul exact, mais le calcul diffus avec normales est très stable. Rappelons simplement à titre d'exemple les moyennes et variances de  $k_1$  et  $k_2$  pour un cylindre de rayon 5.

C'est la qualité de l'estimation de la courbure (et essentiellement la faible variance) qui nous permet d'utiliser des égalités numériques (i.e. à  $\epsilon$  près) pour le classement initial. Rappelons que les paramètres de chaque primitive détectée seront estimés par ajustement en fin de traitement. Ainsi le biais sur  $k_1$  est de moindre importance, et c'est bien la variance expérimentale des estimations qui nous importe ici.

|       | moyenne | variance | m. intérieur | v. intérieur |
|-------|---------|----------|--------------|--------------|
| $k_1$ | 0.202   | 2.5e-10  | 0.202        | 1.7e-10      |
| $k_2$ | 3.92e-4 | 1.74e-6  | 5.06e-8      | 9.97e-16     |

TAB. 2.1: Moyenne et variance de  $k_1$  et  $k_2$  sur un cylindre de rayon 5. Les deux premières colonnes concernent tous les nœuds, les suivantes les nœuds intérieurs seulement, c'est-à-dire ceux qui nous intéressent ... ).

### 2.3.2 Propagation (discussion de H2)

Le second point essentiel que nous avons utilisé est la topologie du maillage. Dans le cas de données non structurées (nuage de points), il faudrait être capable de définir cette topologie; la seconde partie de l'hypothèse H2 (les intersections entre portions sont contenues dans le maillage) posera alors un problème qui peut être une piste d'investigations futures. On pourrait utiliser la haute résolution du calcul de courbure pour détecter *a priori* les intersections inter-primitives. Cet aspect concerne les opérateurs de propagation (car la topologie n'est pas respectée ... ) et le calcul des frontières par `TT2EN_frt`.

Le cas de l'opérateur `app2N_int` est différent, puisqu'on doit comparer les estimations des courbures pour prendre nos décisions. L'expérience montre que la qualité de l'estimation de la courbure assure des tests corrects.

### 2.3.3 Discrétisation (discussion de H1)

L'hypothèse H1 permet de classer tous les triangles d'une surface et simplifie l'analyse de l'algorithme. Elle est cependant difficile à respecter en pratique, en particulier dans le cas des triangles de coins (voir figure 2.3). Ces triangles ont trois sommets sur des frontières (qui ne peuvent jamais être classés): il apparaissent aux intersections de plusieurs portions de surface. Nous devons faire une hypothèse supplémentaire pour être capable de traiter ce type de cas : **on suppose qu'aucun patch plan n'est décrit par un unique triangle** (conséquence de l'hypothèse H1 si elle est respectée). Ainsi, un triangle dont les trois nœuds sont sur une frontière admet exactement deux arêtes de frontières.

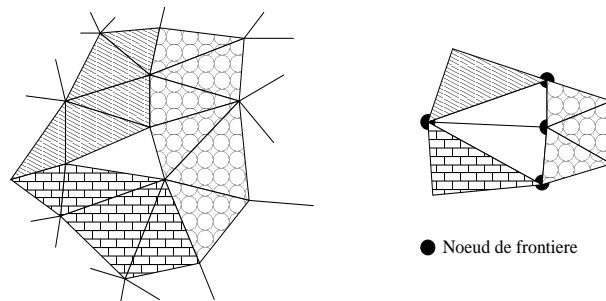


FIG. 2.10: A gauche, le cas d'un triangle à trois nœuds de frontière (chaque remplissage est un patch différent). A droite, un cas topologiquement décidable.

Les opérateurs topologiques locaux ne permettent pas de décider quelles sont les arêtes effectivement de frontières et l'arête interne à un patch (figure 2.10). Nous devons donc utiliser des informations géométriques ou relatives aux courbures. Cependant, les courbures n'existent pas en ces nœuds, bien que nous la calculions

numériquement. Nous avons donc choisi de baser notre critère sur l'angle entre le triangle et ses voisins : les deux arêtes de plus grand angle sont classées frontière. Cette méthode empirique fonctionne sur les cas que nous avons traités, mais n'est pas totalement satisfaisante. Une stratégie de « switch » (inversion de diagonale dans la triangulation) pourrait fournir une solution. On peut aussi reporter le classement de ce type de triangle à la phase ultérieure où les différentes portions reconnues sont ajustées à des modèles surfaciques.

### 2.3.4 Cas réels traités

Nous avons appliqué les algorithmes diffus de calcul de la courbure, et de reconnaissance des types des surfaces à trois pièces issues de la CAO (ces surfaces sont décrites en détail en annexe C). Le tableau 2.2 montre la performance de l'ensemble sur chacune des trois pièces. Les figures 2.13, 2.17 et 2.15 montrent des extraits de ces surfaces. La pièce dite « benz » ne respecte pas les hypothèses de travail, ce cas montre que l'algorithme de reconnaissance permet de mieux comprendre la structure d'une surface triangulée.

|                    | sony  | french | psa  |
|--------------------|-------|--------|------|
| classement initial | 95 %  | 99.9 % | 86 % |
| cohérence          | 87 %  | 80 %   | 71 % |
| propagation        | 98 %  | 88 %   | 90 % |
| post-traitement    | 100 % | 88 %   | 92 % |

TAB. 2.2: Nombre de nœuds classés à l'issue de chaque étape de l'algorithme de reconnaissance en pourcentage du nombre de nœuds initial : sony 4425 nœuds, french 3798 nœuds et psa : 11805 nœuds.

On obtient un très bon taux de reconnaissance sur chacune des pièces. La pièce « sony » respecte les hypothèses H1 et H2, ce qui nous permet de reconnaître le type de surface de chaque nœud. On remarque que la phase de post-traitement classe de nouveaux nœuds car plusieurs portions de primitives sont du type de la figure 2.7. Les deux autres pièces ne vérifient pas l'hypothèse H2

- dans le cas de « french », des intersections sphère-plan ne sont pas contenues dans le maillage : ce sont les nœuds non classés.
- le cas de « psa » est plus complexe :
  - H1 n'est pas vérifiée, car certaines portions ne sont pas assez bien discrétisée (cas de plan décrit par les nœuds de bords uniquement par exemple).
  - H2 n'est pas vérifiée ; ceci concerne surtout dans ce cas les intersections de trois primitives, qui sont lissées.

On peut aussi ajouter quelques prétraitements géométriques : par exemple, les arêtes dont l'angle des triangles de définitions dépasse une certaine valeur sont classés frontière a priori ; les plans peuvent être détectés de la même façon. On obtient donc en sortie une segmentation de la surface en composantes connexes par les arêtes de frontière. La figure 2.11 montre le cas de la pièce « sony ».

## 2.4 Conclusion

Nous avons montré par l'algorithme de reconnaissance des surfaces que le calcul diffus de la courbure sur une surface triangulée présente un grand intérêt pratique.

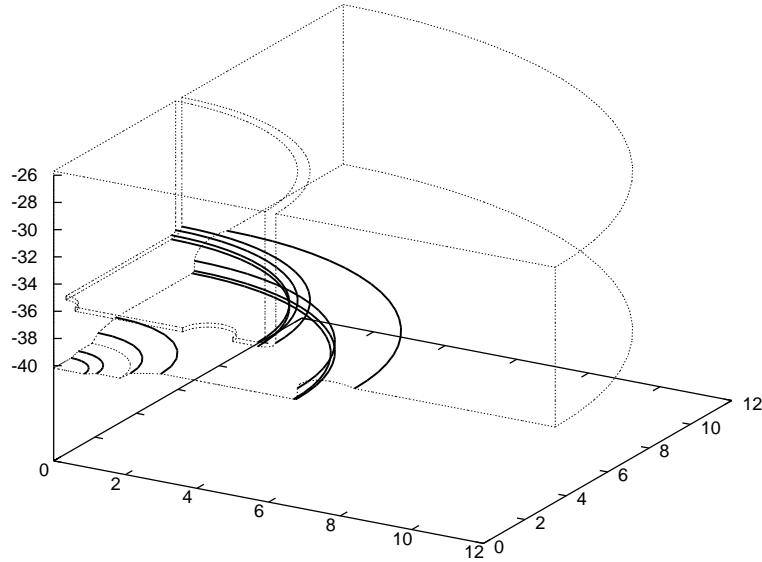


FIG. 2.11: *Frontière d'une pièce : les frontières tracées en gras ont été définies par l'analyse de la courbure, les autres sont détectées par test géométrique.*

D'une part, la grande qualité des résultats (variance faible), ainsi que la haute résolution de l'estimation permettent de développer un algorithme de reconnaissance très simple et rapide. Cette simplicité permet d'envisager des évolutions. On pourrait par exemple s'appuyer sur les centres de courbures pour traiter d'autres surfaces (surfaces de révolution) ; la détection d'arêtes est aussi une voie de recherche particulièrement attrayante.

D'autre part, nous obtenons des résultats très encourageants en terme de reconnaissance : lorsqu'une surface respecte les hypothèses  $H1$  et  $H2$ , on reconnaît le type de tous les nœuds. Le cas de la pièce « sony » le montre. Ces succès montrent combien la notion de pseudo-dérivée est utile en pratique (même pour l'approximation diffuse Hermite avec données différentielles bruitées) et qu'elle permet une excellente estimation de grandeurs dérivées.

Nous montrons aussi l'importance de la connectique, qui est définie dans notre cas par le maillage. Ceci permet d'orienter la recherche (propagation), mais aussi de simplifier les opérateurs. Pour traiter des nuages de points, on peut utiliser des méthodes de triangulation ([Boi84] par exemple) ou de calcul de la connectique ( $\alpha$ -shapes [EM94] par exemple).



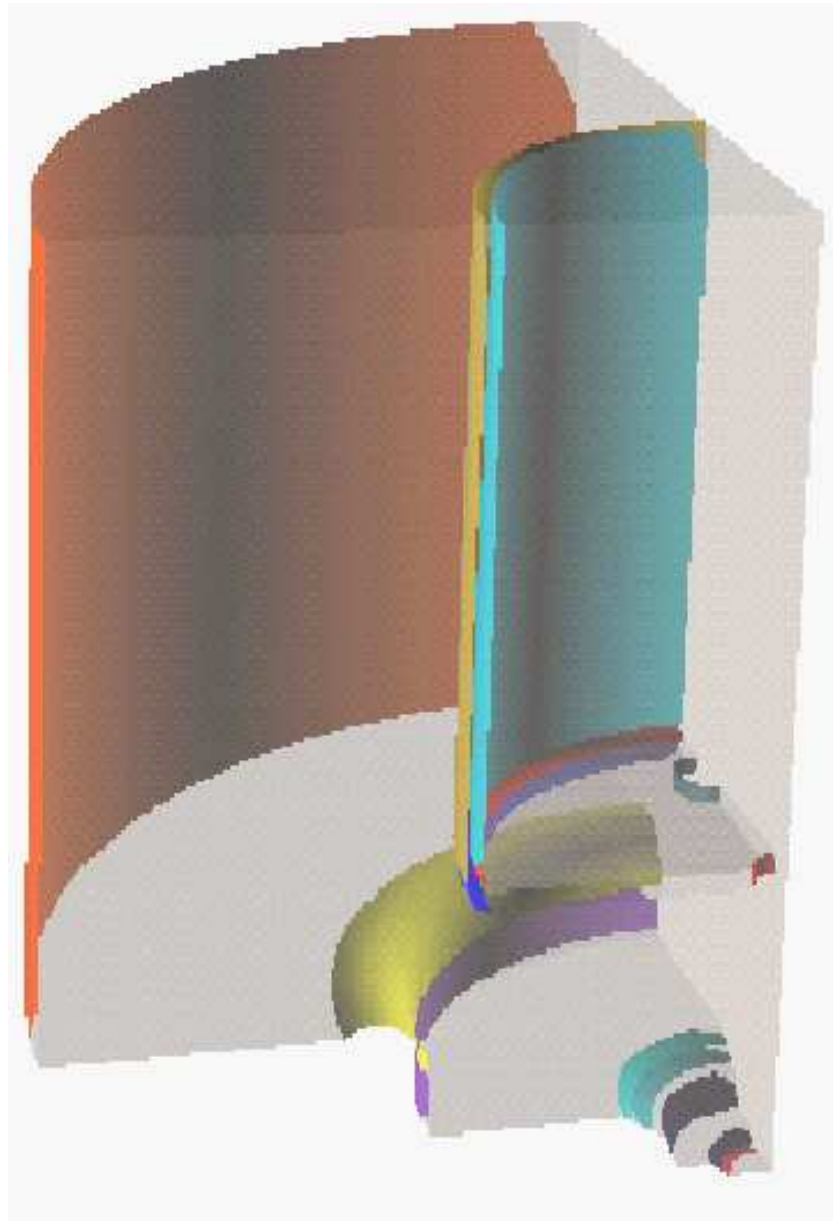


FIG. 2.12: *Sony* : triangles classés de la pièce sony (les plans sont en transparence).

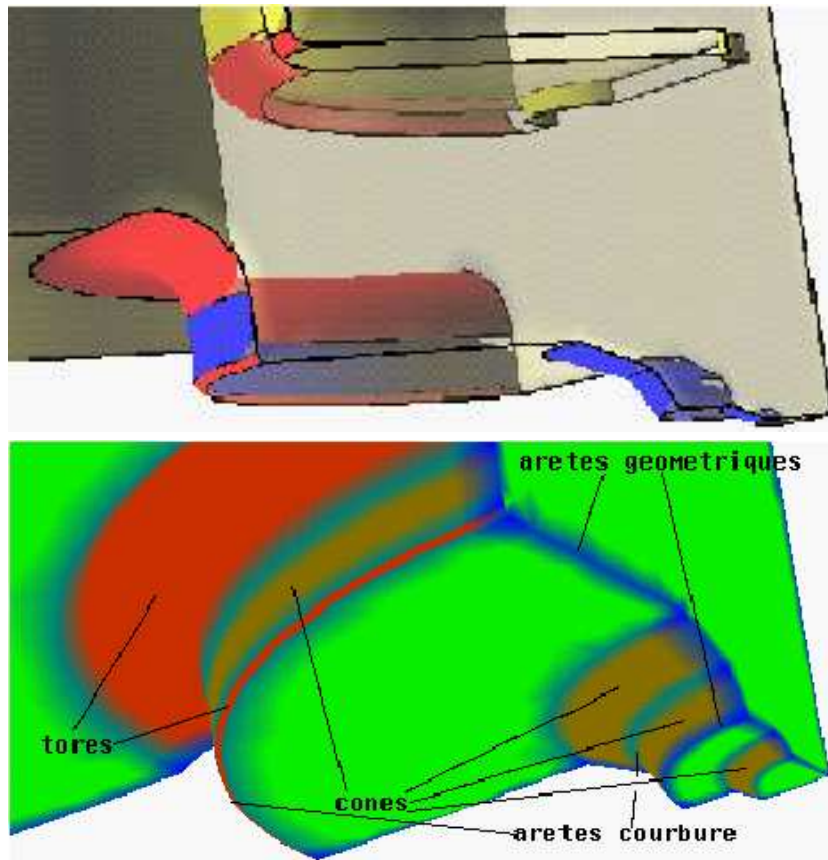


FIG. 2.13: Sony : Classement des nœuds en fin de traitement

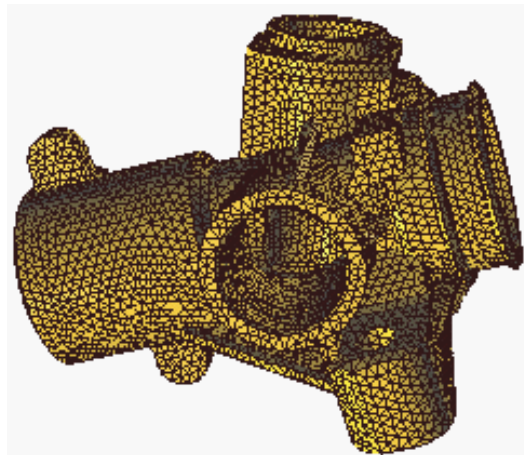
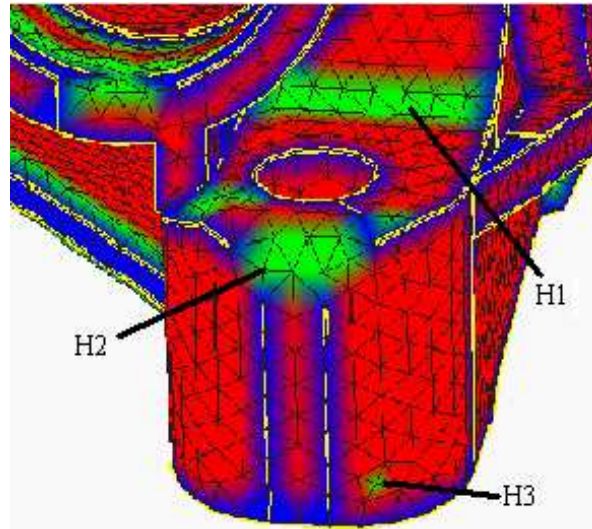
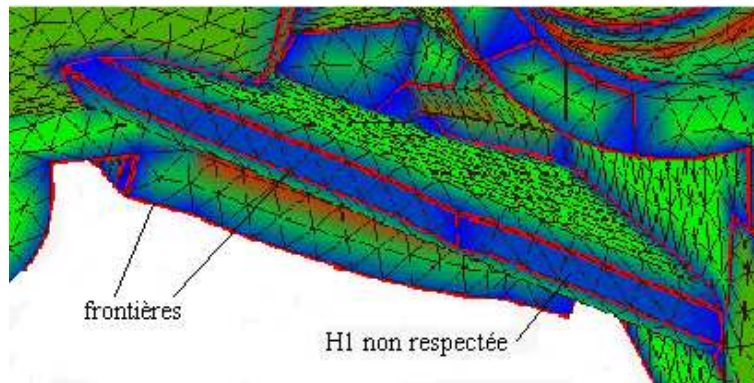


FIG. 2.14: La pièce psa et de son maillage.



rouge: nœuds classés, bleu: frontières, vert: nœuds non classés

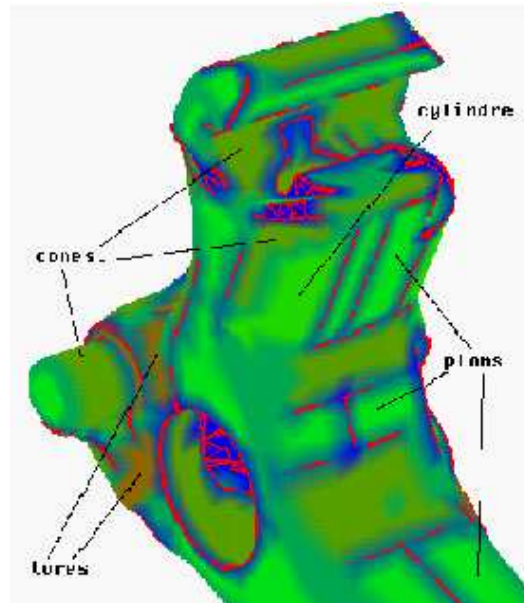


FIG. 2.15: Extraits de la pièce psa (avec le maillage) et de la pièce benz.

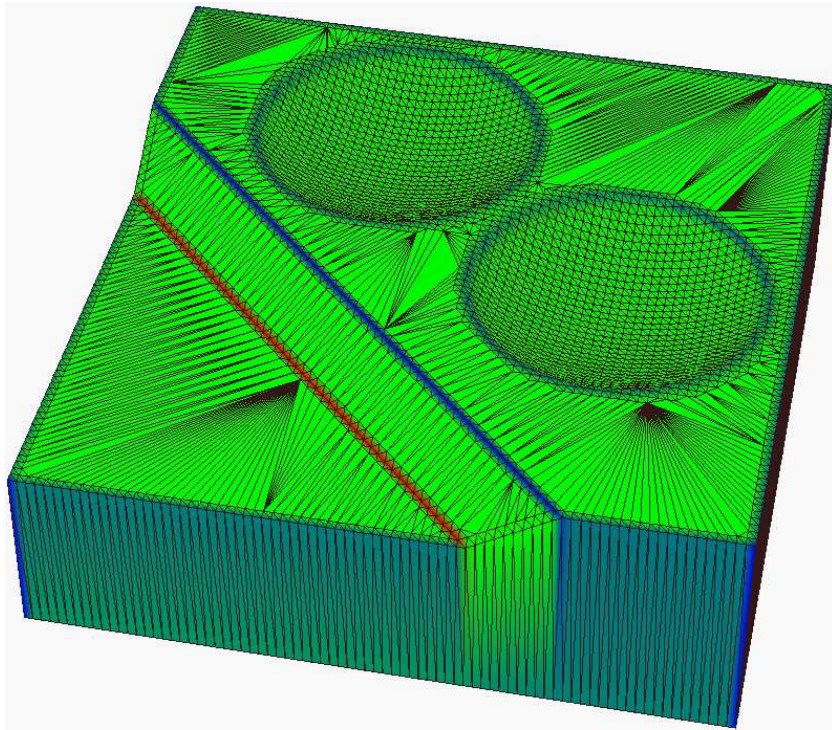


FIG. 2.16: *French*: vue d'ensemble, maillage et première courbure principale  $k_1$ .

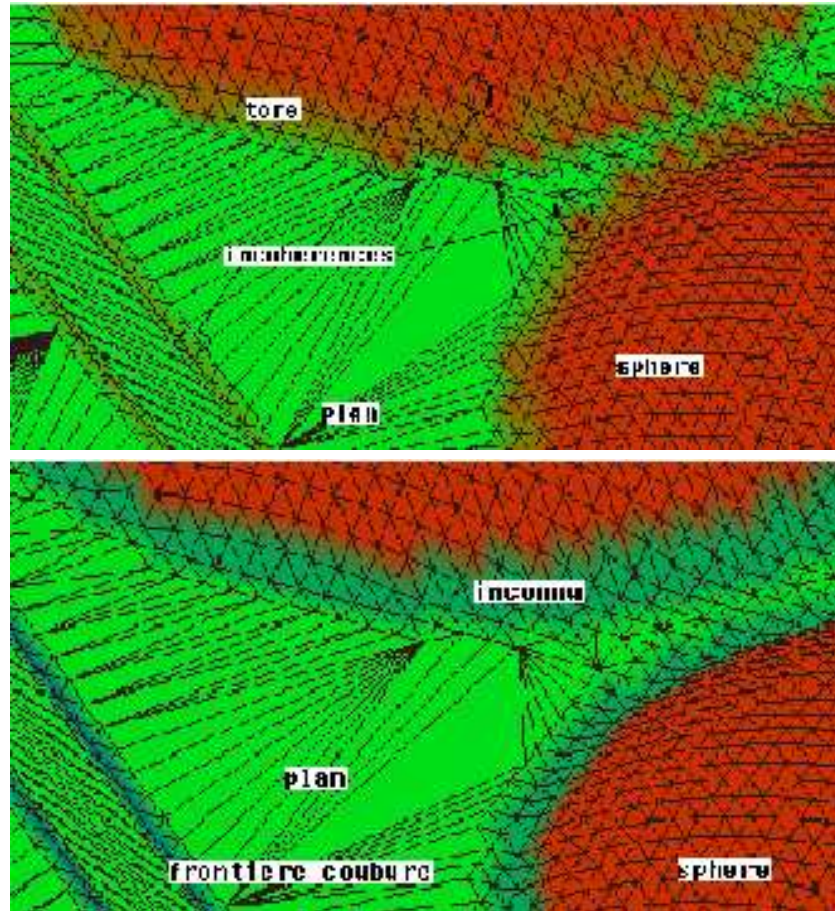


FIG. 2.17: *French: frontière sphère-plan avant la phase de cohérence et en fin de traitement.*

Quatrième partie  
**Éléments Diffus**



# Chapitre 1

## Méthodes sans maillage

Les problèmes abordés par les techniques de type éléments finis sont de plus en plus complexes, et cherchent à décrire la physique des phénomènes de la façon la plus précise possible. Dans le monde du sous-sol, la simulation des phénomènes géologiques est l'un des thèmes d'avenir et présente la plupart des défis des éléments finis : prise en compte de très grands déplacements, fortes non linéairités, fracturation . . . Les méthodes sans maillage ont été développées pour répondre (au moins partiellement) à certains de ces problèmes comme le rappelle [BKO<sup>+</sup>96]. De la même façon, Zienkiewicz [Zie00] cite l'équation d'Helmholtz dans le cas des courtes longueurs d'ondes comme l'un des problèmes ouverts pour les éléments finis. Ce problème particulier a été abordé avec succès par Melenk et Babuska [Mel95] par la technique dite *Partition of Unity Finite Element Methods* (PUFEM). Belytschko *et al.* considère PUFEM comme la racine des différentes méthodes sans maillage [BKO<sup>+</sup>96].

La première méthode de résolution d'équation aux dérivées partielles « sans maillage » est certainement la technique dite *Smooth Particle Hydrodynamics* (SPH) proposée par Lucy [Luc77] et développée par Monaghan *et al.* (voir par exemple l'étude bibliographique [Mon92]). SPH est une technique particulière. A la même époque, Liszka et Orkisz propose une technique de collocation (FDM) s'appuyant sur l'idée de la pseudo-dérivée d'une approximation par moindres carrées [LO80]. Bien que la formulation originale ne fasse pas apparaître de fonction d'influence  $w_i$ , FDM est la première méthode sans maillage avec moindres carrées glissants (MLS). Cependant, les approximations sans maillage pour la résolution d'équations différentielles restent confidentielles jusqu'au début des années 90, lorsque Nayroles, Touzot et Villon proposent la technique des Éléments Diffus [NTV92]. Cette publication qui mélange l'approximation diffuse et sa pseudo-dérivée dans une formulation variationnelle semble relancer l'intérêt pour les méthodes sans maillage. Depuis, les publications se sont succédées et l'on a vu naître de nombreuses méthodes : EFG [BLG94], RKPM [LJZ95], hp-Clouds [DO96], finite point [OIZT96], . . .

Pour chaque publication, les auteurs vont s'attacher à l'un (ou plusieurs) des thèmes suivants :

- Définition et (ou) utilisation d'une approximation particulière pour la résolution d'équations différentielles.
- Type de formulation du problème physique pour la mise en œuvre de la solution approchée.
- Techniques d'intégration des matrices élémentaires lorsque c'est nécessaire.
- Résolution d'un problème physique particulier.



- Multi-résolution et (ou) adaptativité du semis de nœuds.

Ce découpage propose ainsi naturellement la structure d'une étude bibliographique. Dans un premier temps nous proposons une rapide description des formulations généralement utilisées pour la discrétisation des équations aux dérivées partielles en nous appuyant fortement sur [DT84]. Le patch test est ensuite décrit car il joue un rôle central dans l'analyse des techniques sans maillage [Lao96, Kro96]. Puis nous décrirons successivement les approximations utilisées pour les équations différentielles, la formulation employée et la prise en compte des conditions aux limites et l'intégration numérique lorsqu'elle est nécessaire.

## 1.1 Introduction

### 1.1.1 Résidus pondérés

La méthode des résidus pondérés permet d'englober plusieurs méthodes de résolution d'équations aux dérivées partielles. Nous verrons aussi dans la seconde partie que la notion de fonctionnelle d'énergie est un cas particulier de ce formalisme. La troisième partie montre comment les résidus pondérés se déclinent en

- La formulation de Galerkin (Ritz),
- La collocation,
- La méthode des moindres carrés.

Enfin, la présentation initiale exhibe aussi le principe des éléments frontières.

Le problème physique que nous allons prendre en compte est volontairement restreint à l'équilibre

$$\begin{cases} \mathcal{L}(u) + f_\Omega = 0 & \text{sur } \Omega \text{ le domaine d'intérêt} \\ \mathcal{C}(u) = f_{\partial\Omega} & \text{sur la frontière } \partial\Omega \text{ de } \Omega \end{cases} \quad (1.1)$$

et sera illustré par l'équation de la chaleur (ou équation de Poisson)

$$\begin{cases} -\Delta u + Q = 0 \\ \frac{\partial u}{\partial n} = \phi & \text{sur } \partial\Omega_\phi \\ u = u_{\partial\Omega_u} & \text{sur } \partial\Omega_u \end{cases} \quad (1.2)$$

En suivant la figure page viii, ces dernières équations sont obtenues grâce aux lois fondamentales que sont la loi de Fourier (diffusion de la chaleur) et la conservation de l'énergie. Les conditions aux limites sont de types Neumann (flux) sur  $\partial\Omega_\phi$  ou Dirichlet (température imposée) sur  $\partial\Omega_u$ , avec  $\partial\Omega = \partial\Omega_\phi \cup \partial\Omega_u$ . La méthode des résidus pondérés va nous permettre de transformer les équations ainsi obtenues.

Le résidu associé à notre problème 1.1 se définit de façon naturelle par

$$R(u) = \mathcal{L}(u) + f_\Omega \quad (1.3)$$

et écrire que  $u$  vérifie les équations du problème revient à dire que le résidu associé à  $u$  est nul. Nous allons donc chercher  $u$  qui annule la forme intégrale

$$W(u) = \int_\Omega \psi R(u) dV = \int_\Omega \psi (\mathcal{L}(u) + f_\Omega) dV \quad (1.4)$$

pour tous les  $\psi$  d'un ensemble de recherche, et pour  $u$  dans l'ensemble des solutions admissibles (vérifiant les conditions aux limites). Notons que  $R$  est nul si  $W(u)$  est nul pour tous les  $\psi$ , mais qu'on ne sait pas « tester tous les  $\psi$  ». Nous chercherons

donc  $u$  dans un espace de dimension finie  $E_u$  et choisissons  $\psi$  dans un espace de dimension finie  $E_\psi$ . Il paraît nécessaire de demander à ces deux espaces certaines propriétés qui assureront la convergence de notre solution lorsqu'on augmente les dimensions de  $E_u$  et  $E_\psi$ . Il s'agit de

- *La condition d'approximabilité*, qui signifie que l'on approxime « bien » une fonction dans  $E_u$ , c'est-à-dire que cette approximation converge lorsque les dimensions de  $E_u$  et  $E_\psi$  augmentent.
- *La condition de stabilité* qui concerne l'opérateur  $A(u, \psi) = \int_{\Omega} \psi \mathcal{L}(u) dV$ .

Par ailleurs, la définition 1.4 fait intervenir l'opérateur  $\mathcal{L}(u)$  directement, il est commode dans certains cas d'utiliser une intégration par partie,

$$\int_{\Omega} \psi \nabla(u) dV = - \int_{\Omega} u \nabla(\psi) dV + \oint_{\partial\Omega} \psi u \vec{n} dS$$

pour transformer  $W$ , et obtenir une forme plus simple. Dans le cas de l'équation de la chaleur, cette transformation s'écrit

$$\begin{aligned} W(u) &= \int_{\Omega} \psi(-\Delta u + Q) dV \\ &= \int_{\Omega} (\nabla(\psi) \cdot \nabla(u) + \psi Q) dV - \oint_{\partial\Omega} \psi \frac{\partial u}{\partial n} dS \\ &= \int_{\Omega} (\nabla(\psi) \cdot \nabla(u) + \psi Q) dV - \int_{\partial\Omega_\phi} \psi \phi dS \end{aligned}$$

sous les conditions (pour la seconde égalité)

$$\begin{cases} \psi = 0 \text{ sur } \partial\Omega_u \\ u = u_{\partial\Omega_u} \text{ sur } \partial\Omega_u \end{cases}$$

Cette nouvelle forme de  $W$  est appelée forme faible de l'équation de Poisson. La forme faible est utilisée car elle ne propose que des dérivées du premier ordre, ce qui simplifie (en général) grandement la construction de l'espace d'approximation. De plus,  $\int_{\Omega} \nabla(u) \cdot \nabla(\psi) dV$  est symétrique.

Nous pouvons intégrer une nouvelle fois par partie, pour obtenir

$$W(u) = \int_{\Omega} (-\Delta \psi u + \psi Q) dV - \oint_{\partial\Omega} \left( \psi \frac{\partial u}{\partial n} - u \frac{\partial \psi}{\partial n} \right) dS$$

Dès que  $\Delta \psi = 0$  sur  $\Omega$  et  $Q = 0$ ,  $W$  se réduit à l'intégrale de contour. C'est la base de la méthode des éléments frontières (équations intégrales).

**Fonctionnelles et résidus pondérés** On a l'habitude en physique de caractériser un système par son énergie. Les positions d'équilibre sont alors les états qui rendent stationnaire la fonctionnelle d'énergie potentielle. Nous avons donc deux caractérisations intégrales de l'équilibre d'un système :

1. La stationnarité de la fonctionnelle d'énergie
2. La nullité du critère  $W$  issu de la méthode des résidus pondérés.

Il existe un lien entre ces deux formulations : En choisissant  $\psi = \delta u$ , il existe une fonctionnelle dont  $W$  est la première variation (sous certaines conditions sur le système physique).

Prenons le cas de l'équation de Poisson,  $\psi = \delta u$  nous donne

$$W(u) = \int_{\Omega} (\nabla(\delta u) \cdot \nabla(u) + \delta u Q) dV - \int_{\partial\Omega_{\phi}} \delta u \phi dS$$

qui est bien la première variation de la fonctionnelle

$$\pi(u, \nabla(u)) = \int_{\Omega} \left( \frac{1}{2} \nabla(u) \cdot \nabla(u) + uQ \right) dV - \int_{\partial\Omega_{\phi}} u \phi dS$$

Remarquons pour finir que l'on peut enrichir la notion de fonctionnelle à l'aide des multiplicateurs de Lagrange, qui permettent de prendre en compte des contraintes supplémentaires comme les conditions aux limites par exemple.

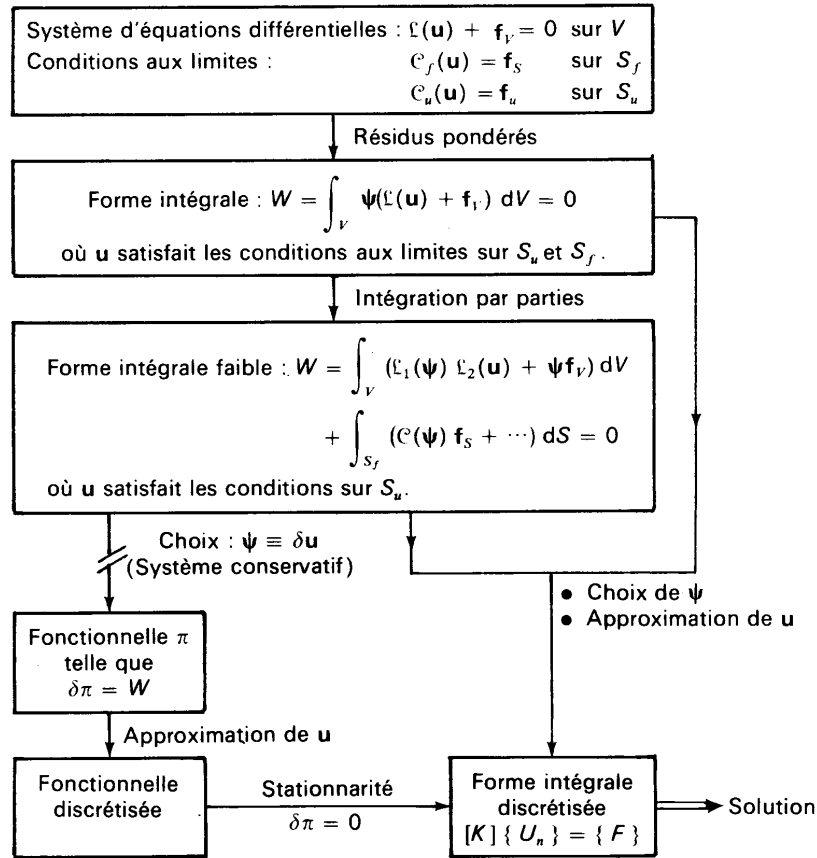


FIG. 1.1: Des équations du système physique au système discret. La partie de gauche concerne la formulation de Ritz, celle de droite la méthode des résidus pondérés (d'après [DT84]).

**Forme discrète** Jusqu'à présent nous n'avons pas fait intervenir les espaces d'approximation  $E_u$  et  $E_\psi$ . La restriction de la recherche de solutions à ces espaces va nous permettre de construire le système fini que nous savons résoudre. En effet, si  $E_u$  est de dimension finie  $n$ , alors tout  $u$  dépend exactement de  $n$  paramètres  $a_1, \dots, a_n$ . Il suffit donc de choisir  $n$  fonctions de tests  $\psi_1, \dots, \psi_n$  pour construire le système d'équations discret. Notons que le choix des fonctions  $\psi$  est contraint

par les propriétés du système linéaire que l'on veut inversible. Ce raisonnement s'applique à la fonctionnelle d'énergie que l'on peut minimiser par rapport aux paramètres  $a_1, \dots, a_n$ , c'est la méthode de Ritz (figure 1.2). La figure 1.1 résume le processus de traitement des équations aux dérivées partielles que nous avons décrit jusqu'à présent.

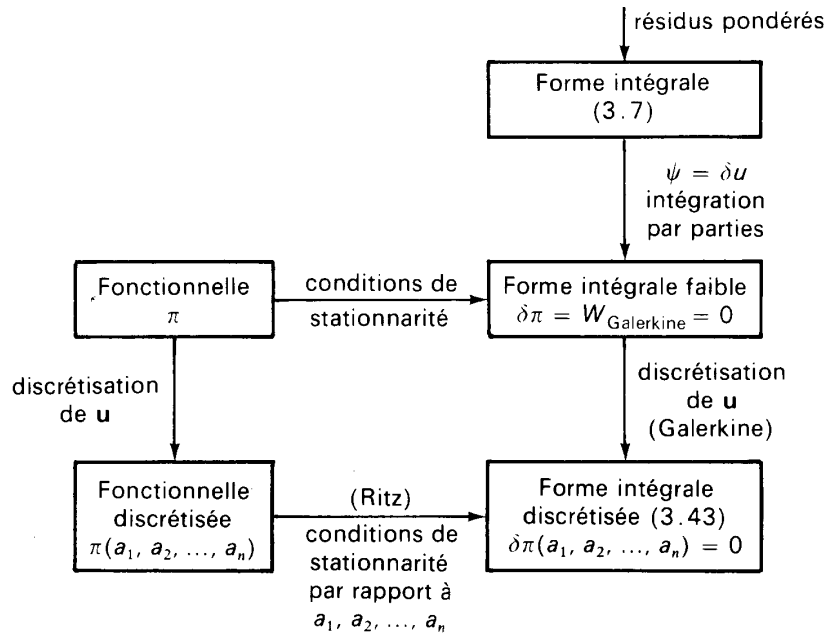


FIG. 1.2: Relation entre les formulations de Galerkin et de Ritz. D'après [DT84].

**Choix de  $\psi$**  Nous venons de voir que la construction de  $W$  et le choix des deux espaces  $E_u$  et  $E_\psi$  permettent de construire un système discret d'où l'on tire une solution approchée du problème posé. Comme  $u$  appartient à un espace de dimension finie, il peut prendre l'une des deux formes suivantes

$$u = \langle N_i \rangle \{u_i\} = N^T U = \langle a_i \rangle \{P_i\}$$

Parmi les choix d'ensemble  $E_\psi$  possibles, nous allons décrire les quatre plus courants, qui donnent les méthodes dites de Collocation (ponctuelle et sous-domaine), de Galerkin, et des Moindres carrés.

- *Collocation ponctuelle* : On choisit  $n$  points distincts dans  $\Omega$  et les fonctions  $\psi_i$  sont les diracs en ces points. La méthode des différences finies est un cas particulier de collocation.
- *Collocation par sous-domaines* : On considère une partition finie de  $\Omega = \sum_{i=1}^n \Omega_i$ . Les fonctions  $\psi_i$  sont les fonctions caractéristiques des  $\Omega_i$ .
- *Galerkin* : le choix  $\psi = \delta u$  amène à la méthode de Galerkin, il est équivalent de choisir  $\psi_i = N_i$ . L'utilisation de l'approximation par sous-domaine pour définir les fonctions de forme  $N_i$  donne la méthode des éléments finis.
- *Moindres carrés* : On choisit  $\psi = \mathcal{L}(u)$  ou  $\psi_i = \mathcal{L}(N_i)$ .

La formulation de Galerkin « usuelle » utilise les résidus pondérés avec intégration par partie. Une variante est d'utiliser les fonctions de forme comme fonction test sans intégration par parties, c'est la formulation de Petrov–Galerkin.

La discrétisation de la forme faible de l'équation de Poisson va donc nous amener à évaluer des termes de la forme (dans la formulation de Galerkin)

$$K_{ij} = \int_{\Omega} \nabla(N_i) \cdot \nabla(N_j) dV$$

et

$$F_i = \int_{\Omega} N_i Q dV - \int_{\partial\Omega_{\phi}} N_i \phi dS$$

pour former le système  $KU = F$  où  $K = [K_{ij}]$  et  $F = \{F_i\}$ .

**Formulation de Galerkin et Collocation** Les deux formulations le plus souvent employées sont la collocation (différences finies) et la formulation de Galerkin (éléments finis). Ces deux techniques ont des propriétés bien différentes :

- La collocation
  - N'admet pas d'interprétation énergétique ;
  - Utilise une « matrice de raideur » non-symétrique ;
  - Traite les conditions aux limites ponctuellement ;
  - Ne nécessite pas d'intégration ;
  - Demande un haut degré de régularité ;
- La formulation de Galerkin
  - Bénéficie d'une théorie bien développée (voir [RT98]) ;
  - Utilise une matrice de raideur symétrique ;
  - Demande des intégrations ;
  - Doit traiter les conditions aux limites sur toute la frontière ;
  - Fait intervenir des dérivées d'ordre moins élevé.

Ainsi, les méthodes sans maillage s'adaptent-t-elles bien à la collocation car il est facile de contruire des dérivées d'ordre élevé sur des semis de nœuds arbitraires. Cependant, la largeur de bande de la matrice de raideur augmente et elle n'est pas symétrique. Les expériences menées par différentes équipes montrent que la collocation n'est pas très stable et manque de robustesse vis-à-vis des semis de nœuds [LB00]. De plus, l'interprétation énergétique de la formulation de Galerkin est très appréciée de nombreux scientifiques (mécanique du solide par exemple). Comme cette technique permet de construire des matrices de raideur symétriques et de moindre largeur de bande, c'est dans ce cadre que se situe notre travail.

Une alternative à la technique de Galerkin est la technique de Petrov–Galerkin où l'on n'utilise pas la transformation de l'énergie par intégration par partie. Avec cette formulation, les propriétés nécessaires pour la convergence sont plus faibles (voir [Kro96, Lao96]). Cependant, il faut dans le cas de l'équation de la chaleur ou de l'élasticité linéaire calculer des dérivées secondes.

### 1.1.2 Formulation variationnelle de l'équation de la chaleur

Reprenons ici l'exemple de l'équation de la chaleur sous conditions de Dirichlet homogènes dans un ouvert  $\Omega$  en explicitant les espaces de fonctions sous-jacents (voir [Bre83, RT98]).

$$\begin{cases} -\Delta u + \alpha u = f \text{ sur } \Omega \\ u|_{\partial\Omega} = 0 \end{cases} \quad (1.5)$$

où  $f$  est une fonction donnée de  $L^2(\Omega)$ .

La première étape est de transformer le problème (1.5) pour diminuer le degré de dérivation (théorème des travaux virtuels). Pour cela, on cherche  $u \in \mathcal{H}_0^1$ , telle que

$$\forall v \in \mathcal{H}_0^1(\Omega), \int_{\Omega} \nabla(u) \cdot \nabla(v) \, dx + \alpha \int_{\Omega} uv \, dx = \int_{\Omega} fv \, dx \quad (1.6)$$

On montre par intégration par partie (formule de Green) qu'une solution de ce problème est bien solution de l'équation initiale.

Le problème variationnel admet une unique solution comme le démontre le Lemme de Lax-Milgram. Pour cela on pose

$$a(u, v) = \int_{\Omega} \nabla(u) \cdot \nabla(v) \, dx + \alpha \int_{\Omega} uv \, dx$$

et

$$l(v) = \int_{\Omega} fv \, dx$$

Comme  $a$  est bilinéaire, continue et elliptique dans  $\mathcal{H}_0^1$ , et  $l$  est linéaire continue, le lemme s'applique bien. De plus, la solution est aussi solution du problème de minimisation

$$\min_{v \in \mathcal{H}_0^1} \left\{ \frac{1}{2} a(v, v) - l(v) \right\}$$

Cependant, les espaces mis en jeu ne sont pas de dimension finie en général. Nous devons donc chercher une solution approchée de  $u$  dans un sous-espace vectoriel  $V_n$  de  $\mathcal{H}_0^1$ . Si l'on est capable de construire une suite de sous-espaces telle que la solution  $u$  du problème (1.5) y soit « bien approchée », c'est-à-dire

$$\lim_{n \rightarrow +\infty} \inf \left\{ \|u - v\|_{\mathcal{H}_0^1}, v \in V_n \right\} = 0$$

alors on montre que la solution  $u_n$  du problème variationnel dans  $V_n$  converge vers la solution réelle  $u$  dans  $\mathcal{H}_0^1$  de la façon suivante ( $C$  est une constante)

$$\|u_n - u\|_{\mathcal{H}_0^1} \leq C \inf \left\{ \|u - v\|_{\mathcal{H}_0^1}, v \in V_n \right\} \quad (1.7)$$

Ceci suffit pour construire des solutions approchées du problème de Dirichlet homogène en faisant abstraction de l'intégration numérique.

Le cas non homogène se traite en minimisant la même énergie, mais cette fois-ci dans l'espace des fonctions vérifiant les conditions aux limites imposées.

Explicitons pour finir les équations à résoudre si l'espace  $V_n$  admet les fonctions  $N_1, \dots, N_n$  comme base. La solution du problème variationnel dans  $V_n$ ,  $u_n$  peut

s'écrire  $u_n = \sum_{i=1}^n u_i N_i$ , et le vecteur de ses coordonnées  $U = \langle u_1, \dots, u_n \rangle^\top$  est solution du système

$$\begin{bmatrix} K & P \\ P^\top & 0 \end{bmatrix} \begin{Bmatrix} U \\ \Lambda \end{Bmatrix} = \begin{Bmatrix} F \\ S \end{Bmatrix}$$

où

$$\begin{cases} K_{ij} = \int_I \nabla(N_i) \cdot \nabla(N_j) \, d\omega + \alpha \int_I N_i N_j \, dx \\ F_i = \int_\Omega f N_i \, d\omega \\ P \text{ et } S \text{ représentent les conditions de Lagrange} \end{cases}$$

Nous avons choisi ici de prendre en compte les conditions aux limites via des conditions de Lagrange, mais il existe d'autres techniques décrites dans [DT84] par exemple.

### 1.1.3 Crimes variationnels

Le terme de *crime variationnel* a été proposé par Strang et Fix dans [SF73]. C'est pour mieux comprendre cette notion que nous avons repris la formulation variationnelle de l'équation de la chaleur dans son cadre mathématique.

La section précédente montre que les ingrédients essentiels dans la formulation variationnelle sont

- les espaces fonctionnels mis en jeu (régularité, norme),
- le calcul de la fonctionnelle  $\Pi_0(v) = \frac{1}{2}a(v, v) - l(v)$  (qui remplace l'équation différentielle).

En pratique, pour la formulation de Galerkin, ceci signifie que

1. Les fonctions de forme devraient être  $C^{m-1}$  pour un problème d'ordre  $2m$ .
2. Les conditions aux limites devraient être vérifiées exactement.
3. La fonctionnelle  $I$  devrait être calculée exactement.

Cependant, en pratique, on viole ces recommandations: ce sont les crimes variationnels. Les éléments de coques non conformes ne respectent pas la première recommandation, et les éléments sous intégrés ne respectent pas la condition 3 par exemple. Cependant, on sait par expérience que ces techniques sont convergentes. Leur analyse n'a été possible que grâce au *patch test* proposé par Irons (voir par exemple [IR72]).

### 1.1.4 Le patch test

La définition première du patch test a été donnée par Irons et se veut une caractérisation d'ingénieur de la convergence d'éléments finis particuliers en mécanique (voir [IL83] par exemple). Le test consiste dans ce cadre à vérifier que *toutes* les solutions à contraintes constantes sont calculées exactement. Pour l'équation de la chaleur, ceci revient à calculer exactement les solutions de flux  $\nabla(u)$  constant.

Bien que le patch test ait été contreversé (voir [Stu80]), on peut considérer qu'une preuve de sa validité a été donnée dès l'ouvrage de Strang et Fix [SF73]. Le « dernier mot » revenant à Zienkiewicz et Taylor [TSZC86, ZT94] pour qui le patch test est

- (a) Une condition nécessaire de convergence d'un code éléments finis,

- (b) Une condition suffisante de convergence,
- (c) Un test des algorithmes utilisés,
- (d) Un moyen de tester des algorithmes violant certaines propriétés ou certaines lois,
- (e) Enfin, un moyen de tester la programmation.

Dans l'intervalle, le patch test a été très étudié ou utilisé pour le développement de nouveaux éléments, citons pour mémoire [Veu74, SB77] et de façon plus récente [FHM95].

Depuis, le patch test a connu de nouveaux développements comme le montrent [ZT97] et son utilisation pour l'analyse des méthodes sans maillage [Lao96, Kro96]. Nous verrons plus loin qu'il est à la base des techniques d'intégration spécifiques mises au point par différentes équipes [BK98, BT V99c].

## 1.2 Choix d'une méthode d'approximation

Il existe plusieurs types d'approximation sans maillage comme nous l'avons vu dans la première partie. Dans la littérature, on trouve les techniques suivantes pour la résolution d'équations aux dérivées partielles:

- *Smooth particle hydrodynamics* [Mon92, BK98],
- L'approximation diffuse et ses variantes [LO80, NTV92, BLG94, OIZT96],
- L'approximation diffuse continue et ses variantes [LJZ95],
- hp-clouds [DO96],
- Les *radial basis functions* [Kan90, Wen99],
- La techniques des voisins naturels [Suk98].

On se place dans la suite dans un ouvert  $\Omega$  connexe de  $\mathbb{R}^n$ . On y approche une fonction  $u$  connue sur un semis de nœuds  $(x_i)_{i \in I}$  de  $\Omega$ .

### 1.2.1 Smooth particle hydrodynamics (SPH)

Le principe de base de SPH est d'approcher une fonction  $u$  par son produit de convolution par un noyau  $w_h$  défini par  $w_h = c(h) * w_{ref}(\|x\|/h)$  où  $c(h)$  est un coefficient de normalisation ( $1/c(h) = \int_{\Omega} w_h(x) d\Omega$ ):

$$\tilde{u}(x) = \int_{\Omega} w_h(x-y)u(y) d\Omega \quad (1.8)$$

Cependant, cette formulation intégrale n'est pas utilisable et on lui substitue une forme discrète

$$\tilde{u}(x) = \sum_{i \in I(x)} w_h(x-x_i)u_i V_i. \quad (1.9)$$

Le coefficient  $V_i$  est le volume associé au nœuds  $i$ . Dans la formulation originale, ce coefficient s'écrit  $\frac{m_i}{\rho_i}$  où  $m_i$  est la masse du nœud  $i$  et  $\rho_i$  est sa masse volumique. Cette approximation conduit à une formulation particulière des équations; elle a été développée à l'origine pour résoudre des problèmes d'astrophysique avec conditions aux limites à l'infini [Luc77].



La forme discrète 1.9 fait apparaître des fonctions de forme

$$N_i(x) = w_h(x - x_i)V_i.$$

Cependant SPH n'est que consistante d'ordre 0 si on ne fait pas d'hypothèse supplémentaire sur le noyau  $w_h$ , ceci peut poser des difficultés de convergence. Le cas 1d est étudié dans [BKO<sup>+</sup>96].

### 1.2.2 Moindres carrés glissants

**Diffus continu, *reproducing kernel particle* ou RKPM** Une technique pour rendre consistante une approximation du type SPH est l'approximation diffuse continue (voir le chapitre 2 de la première partie). Une alternative est proposée dans [LJZ95] avec l'approximation *reproducing kernel particle*. Rappelons que l'approximation RKPM s'écrit

$$\tilde{u}(x) = \int_{\Omega} K_{\rho}(y - x, x)u(y) dy$$

où  $K_{\rho}$  est une fonction de pondération assurant la consistance de l'approximation. Ce noyau s'écrit comme le produit de  $w_{\rho}$  et d'une fonction de correction  $C(y - x, x)$  (voir équation 2.8, page 16 ou [LJZ95]). Le passage du continu au discret se fait pour RKPM de la même façon que pour SPH :

$$\tilde{u}(x) = \sum_{i \in I(x)} K_{\rho}(x_i - x, x)u_i V_i \quad (1.10)$$

et les fonctions de forme associées sont  $N_i(x) = K_{\rho}(x_i - x, x)V_i$ .

La version continue de l'approximation respecte bien les conditions de consistance, et les résultats numériques présentés avec la méthode RKPM semblent le confirmer. Cependant, le passage à la forme discrète peut dégrader cette propriété suivant le choix des volumes d'intégration  $V_i$  (voir à ce sujet [LHCG97], page 297). Dans [LL99a], les auteurs évaluent la correction pour la forme discrétisée de l'approximation. Dans ce cas, seuls les poids d'intégration  $V_i$  distinguent RKPM de l'approximation diffuse ; en particulier, pour des poids égaux entre eux, les fonctions de forme des deux approximations sont identiques (voir page 224).

**Diffus discret (MLS)** Nous avons décrit en détail l'approximation diffuse dans la première partie. Il existe deux variantes dans son application aux équations aux dérivées partielles :

1. avec pseudo-dérivation (*Diffuse Element Method* ou DEM),
2. avec dérivation complète (*Element Free Galekin* ou EFG).

Notons que la première technique a été la première utilisée pour de la collocation dans [Jen72, LO80]. L'approximation proposée n'avait cependant pas encore atteint sa forme finale puisqu'on ne trouve pas encore la notion de fonction d'influence  $w_i$ . Avec cette technique, la consistance est exacte avec une forme discrète, elle est donc aujourd'hui la plus utilisée.

### 1.2.3 hp-clouds et éléments finis généralisés ou PUFEM

Les principes de hp-clouds et PUFEM sont proches et l'idée est d'améliorer l'approximation de la solution de l'équation différentielle *a priori* pour améliorer

la convergence de la méthode (inégalité 1.7, page 141). Pour améliorer la capacité d'approximation de l'espace de recherche de la solution, deux techniques sont possibles :

- On augmente le degré de l'approximation (méthode  $p$ , hp-clouds)
- On utilise une connaissance a priori de l'équation à résoudre (méthode PUFEM, [Mel95]).

La technique s'appuie sur la notion de partition de l'unité.

**Définition 19** Soit  $(\Omega_i)_{i \in I}$  un ensemble d'ouvert recouvrant  $\Omega$  et un ensemble de fonctions  $(\varphi_i)_{i \in I}$  de régularité donnée tels que

- (i)  $\text{supp}(\varphi_i) = \Omega_i$
- (ii)  $\forall x \in \Omega, \sum_{i \in I} \varphi_i(x) = 1.$

Les  $\varphi_i$  forment une partition de l'unité de  $\Omega$  subordonnée aux  $(\Omega_i)_{i \in I}$ .

Notons que les fonctions de forme diffuses respectent ces conditions. On considère aussi des espaces de fonctions  $V_i^p$  sur  $\Omega_i$  (dans  $\mathcal{H}^1(\Omega \cap \Omega_i)$  pour les équations elliptiques usuelles) et l'espace de recherche des solutions de l'équation différentielle est

$$V_n = \text{span} \{ \varphi_i v_i^p, i \in I, v_i^p \in V_i^p \} \quad (1.11)$$

Dans le cas de hp-clouds, les espaces  $V_i^p$  sont des monômes, et la partition de l'unité initiale des fonctions de forme diffuse. Chaque nœud porte ainsi plusieurs degré de liberté (un par fonction de  $V_i^p$ ) et l'approximation d'une fonction s'écrit au point  $x$

$$\tilde{u}(x) = \sum_{i \in I(x)} \sum_{v_i^p \in V_i^p} a_{i,p} \varphi_i v_i^p(x)$$

où les  $a_{i,p}$  sont des constantes. L'intérêt de cette technique réside dans deux points [DO96] :

- La convergence de l'approximation bénéficie des propriétés des fonctions  $\varphi_i$  et des fonctions  $v_i^p$ .
- Les fonctions de forme sont simples à calculer par rapport aux autres méthodes de type  $p$ .

Dans le cas de PUFEM, les espaces  $V_i^p$  sont construits à partir d'une connaissance *a priori* de la solution (par exemple à partir de la solution homogène).

Ces techniques favorisent la « multiplication externe » pour enrichir l'espace d'approximation. Cependant, il est possible d'utiliser la construction de l'approximation diffuse pour augmenter la consistance de  $V_n$  en intégrant dans la fonction  $\tilde{u}$  des termes supplémentaires comme le montre [FCMB97] (*enriched basis*).

### 1.2.4 Autres techniques

**Radial basis functions** Les interpolateurs radiaux (RBF) décrit au chapitre 3 de la première partie ont aussi été utilisés pour le traitement des équations différentielles. On trouve deux types d'applications :

1. Les RBF sont utilisées comme technique de fabrication de fonctions de forme ([ZPC98, Wen99] par exemple).

2. On utilise le modèle sous-jacent à l'interpolation [Kan92].

Dans le premier cas, le type d'interpolant utilisé est crucial pour la localisation des fonctions de forme et leur consistance. Par exemple, si on force la consistance par filtrage (voir le système 1.7 page 56), alors les fonctions de forme ne sont pas locales. On peut utiliser leur décroissance pour tronquer leur influence, mais il est alors nécessaire d'avoir une estimation de l'erreur ainsi commise. Le deuxième type d'approche est beaucoup plus porteur : nous avons vu que les RBF ont un modèle sous-jacent (la construction de l'approximation myope le montre en détail pour les splines plaques minces), l'objectif est d'utiliser ce modèle pour avoir de bonnes propriétés au niveau de l'interpolation. On retrouve ici l'idée des *generalized finite elements* de Babuska et Melenk [Mel95].

**Voisins naturels** Nous avons cité l'interpolant de Sibson dans la première partie (chapitre 1, page 9). Comme cette interpolation est consistante d'ordre 1, elle peut être utilisée pour résoudre des équations aux dérivées partielles avec formulation de Galerkin [Suk98]. Cependant, la forme très complexe des supports de fonctions de forme rend l'intégration difficile.

## 1.3 Raffinement de la méthode d'interpolation

Nous avons passé en revue les différentes techniques d'approximation (ou d'interpolation) utilisées dans la littérature. Un certain nombre d'articles proposent des raffinements de ces méthodes d'approximation.

### 1.3.1 Moindres carrés glissants

**Calcul des fonctions de forme diffuses** Le calcul des fonctions de forme diffuses ou de leurs dérivées (au sens usuel ou diffus) est lourd (voir par exemple [KB95]). La caractérisation des fonctions de forme et le système associé (théorème 3, page 27) permet de mettre en œuvre un calcul rapide [BTV99a]. Les détails sont donnés en annexe A.

**Double grille** En élasticité linéaire, la collocation demande de calculer des dérivées secondes. Une idée pour limiter l'ordre de consistance des fonctions de forme et ainsi la largeur de bande de la matrice de raideur est d'utiliser deux semis de points distincts [BTV99b]. La technique s'inspire des éléments finis où les déplacements sont portés par les nœuds et les contraintes-déformations par les points de Gauss. Cette technique amène à une formulation de type  $B_d H B$  comme en éléments finis mais où les deux matrices  $B_d^\top$ ,  $B$  sont différentes. En pratique, on considère deux ensembles de points : les nœuds  $\mathbf{x}_i$  et les points d'évaluation  $\mathbf{y}_j$  et les fonctions de forme diffuses associées  $N_i$ ,  $M_j$ . Des déplacements aux nœuds, on calcule les déformations aux points d'évaluation

$$\begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{Bmatrix} = [B] \begin{Bmatrix} \vdots \\ u_i^n \\ v_i^n \\ \vdots \end{Bmatrix}, [B] = \begin{bmatrix} \dots & \frac{\partial N_i^n}{\partial x} & 0 & \dots \\ \dots & 0 & \frac{\partial N_i^n}{\partial y} & \dots \\ \dots & \frac{\partial N_i^n}{\partial y} & \frac{\partial N_i^n}{\partial x} & \dots \end{bmatrix}$$

puis les contraintes au points d'évaluation par la loi de l'élasticité linéaire  $\sigma = [D]\epsilon$ . On revient aux nœuds grâce aux fonctions de forme  $M_i$  par

$$\left\{ \begin{array}{l} \frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} \\ \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_y}{\partial y} \end{array} \right\} = [B_d]^\top \left\{ \begin{array}{l} \vdots \\ \sigma_x^{e_i} \\ \sigma_y^{e_i} \\ \tau_{xy}^{e_i} \\ \vdots \end{array} \right\}, [B_d]^\top = \left[ \begin{array}{ccc} \dots & \frac{\partial M_i^n}{\partial x} & 0 \\ \dots & 0 & \frac{\partial M_i^n}{\partial y} \\ \dots & \frac{\partial M_i^n}{\partial y} & \frac{\partial M_i^n}{\partial x} \\ \dots & \dots & \dots \end{array} \right]$$

Les relations précédentes permettent d'écrire les équations d'équilibre aux nœuds, soit de façon élémentaire

$$[B_d][D][B] \left\{ \begin{array}{l} \vdots \\ u_i^n \\ v_i^n \\ \vdots \end{array} \right\} = [k_i] \left\{ \begin{array}{l} \vdots \\ u_i^n \\ v_i^n \\ \vdots \end{array} \right\} = \{F_i\}$$

où  $\{F_i\}$  représente les efforts évalués au nœud  $i$ . On retrouve ici une forme qui ressemble aux éléments finis, mais où l'on perd certaines propriétés :

- $B_d^\top$  et  $B$  sont différentes,
- les « matrices de raideur »  $[k_i]$  ne sont pas carrées,
- la matrice de raideur globale n'est pas symétrique.

D'après l'auteur, la collocation double grille souffre aussi de problèmes de stabilité, bien que moindres que ceux de la collocation classique [LB00].

**Dérivée réelle de l'approximation diffuse** L'utilisation de pseudo-dérivées demande une intégration adaptée comme nous le montrons dans le chapitre suivant. Un premier pas (mais pas suffisant) est de calculer la dérivée au sens usuel des fonctions de forme de l'approximation diffuse. Cependant, ce calcul est très couteux si l'on utilise le système de calcul direct des fonctions de forme (voir page 26)

$$\langle N(x) \rangle = \langle p(x) \rangle [A(x)]^{-1} P^\top W(x)$$

car cela demande de dériver  $[A(x)]^{-1}$ . Pour éviter ce calcul inutile, Krongauz [Kro96] propose de calculer la dérivée de  $\alpha(x)$  en dérivant le système (2.14, page 21) centré, soit (en 1d)

$$A(x)\alpha'(x) + A'(x)\alpha(x) = b'(x)$$

Comme la matrice  $A(x)$  est déjà triangularisée pour le calcul de  $\alpha(x)$ , on effectue alors une « économie d'échelle ».

**Discontinuité** Le développement des méthodes sans maillage est motivé en particulier par la régularité des approximations utilisées. Cependant, il existe des cas où la solution contient des discontinuités. Par exemple une barre composée de deux matériaux, de caractéristiques  $E_1, \nu_1$  sur  $[0, l/2]$  et  $E_2, \nu_2$  sur  $[l/2, l]$  en traction (cet exemple est donné dans [Kro96]). La dérivée des déplacements est discontinue à la soudure. Dans ce cas, les auteurs tirent avantage de la dérivation complète de l'approximation et utilisent à la jonction une fonction d'influence de dérivée discontinue qui permet de construire une fonction de forme de dérivée discontinue au nœud de soudure (figure 1.3).

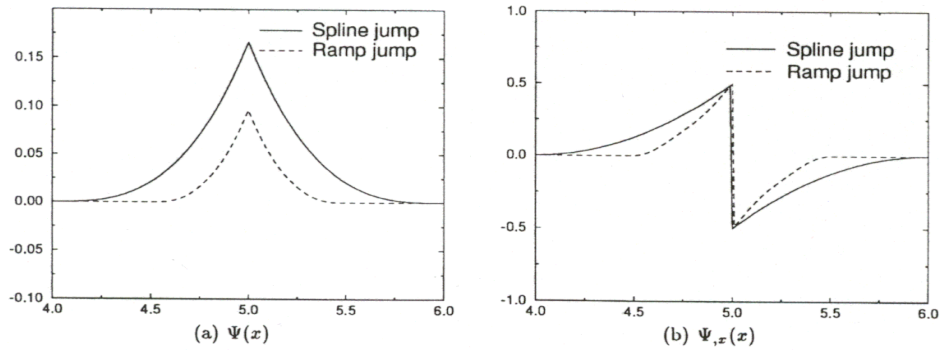


FIG. 1.3: Fonctions de forme avec dérivée discontinue (d'après [KB98]).

Cette technique prend bien en compte la discontinuité, mais nous préférons dans ce cas utiliser deux poutres soudées par le nœud d'interface. En effet les fonctions de forme à dérivée(s) discontinue(s) cachent le fait que l'on a deux matériaux différents. Or cette information doit être portée par le modèle géométrique et permet donc de gérer facilement deux approximations couplées. Par contre ces fonctions de forme peuvent être utilisées lorsqu'on applique une force ponctuelle (figure 1.4).

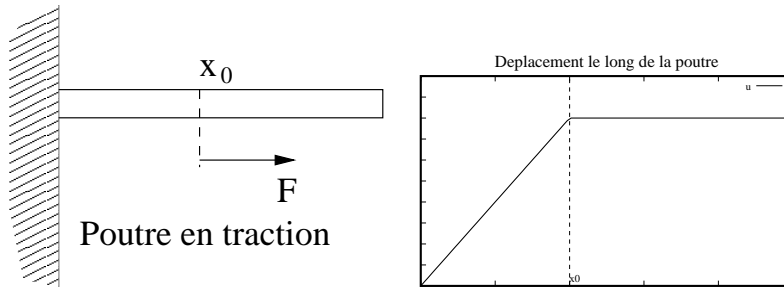


FIG. 1.4: Discontinuité de la dérivée avec force ponctuelle.

### 1.3.2 Consistance

**Dérivées consistantes** En suivant la même idée, Krongauz propose aussi de rendre les *dérivées* de l'approximation de Shepard (diffus de degré 0) consistantes d'ordre 1. Dans le cas de l'élasticité linéaire, ce sont les seules conditions de consistances nécessaires.

En 2d, les conditions de consistances pour les dérivées des fonctions de forme s'écrivent (on utilise la notation usuelle  $f_{,x} = \frac{\partial f}{\partial x}$ )

$$\left\{ \begin{array}{l} \sum_{i \in I(x)} N_{i,x}(\mathbf{x}) = 0 \\ \sum_{i \in I(x)} N_{i,y}(\mathbf{x}) = 0 \\ \sum_{i \in I(x)} N_{i,x}(\mathbf{x}) x_i = 1 \\ \sum_{i \in I(x)} N_{i,y}(\mathbf{x}) x_i = 0 \\ \sum_{i \in I(x)} N_{i,x}(\mathbf{x}) y_i = 0 \\ \sum_{i \in I(x)} N_{i,y}(\mathbf{x}) y_i = 1 \end{array} \right.$$

Les dérivées partielles seront calculées comme un mélange des dérivées partielles

des fonctions de forme de Shepard  $\phi_i$

$$\begin{cases} N_{i,x}(\mathbf{x}) &= \alpha_1(\mathbf{x})\phi_{i,x}(\mathbf{x}) + \alpha_2(\mathbf{x})\phi_{i,y}(\mathbf{x}) \\ N_{i,y}(\mathbf{x}) &= \beta_1(\mathbf{x})\phi_{i,x}(\mathbf{x}) + \beta_2(\mathbf{x})\phi_{i,y}(\mathbf{x}) \end{cases}$$

Les deux premières conditions sont respectées car on utilise les fonctions de forme de Shepard, les quatres autres conditions permettent le calcul des coefficients  $\alpha$  et  $\beta$ . Cependant, ces dérivées ne passent pas le patch test et il est nécessaire d'utiliser la formulation de Petrov–Galerkin (voir partie 1.4).

**Consistance de SPH** Le schéma d'approximation SPH n'est lui aussi que consistant d'ordre 0. Il est possible d'utiliser la technique précédente pour calculer des dérivées consistantes, mais cela nous restreint à la formulation de Petrov–Galerkin et à certaines classes de problèmes.

Bonet propose une alternative pour la formulation de Galerkin elle-même [BK98] en utilisant une fonction de correction selon le principe de RKPM, mais cette fonction de correction est calculée après la discrétisation de l'approximation (1.10). On modifie le noyau d'approximation grâce aux fonctions  $\alpha(\mathbf{x}), \beta(\mathbf{x})$  (nous restons ici à deux dimensions)

$$\hat{w}_i(\mathbf{x}) = w_i(\mathbf{x})\alpha(\mathbf{x}) (1 + \beta(\mathbf{x}) \cdot (\mathbf{x} - \mathbf{x}_i))$$

qui sont calculées pour respecter les conditions de consistance (la seconde égalité est vectorielle)

$$\begin{cases} \sum_{i \in I(\mathbf{x})} \hat{w}_i(\mathbf{x})V_i &= 1 \\ \sum_{i \in I(\mathbf{x})} (\mathbf{x} - \mathbf{x}_i)\hat{w}_i(\mathbf{x})V_i &= 0 \end{cases}$$

et les auteurs donnent l'expression de  $\alpha, \beta$

$$\begin{cases} \alpha(\mathbf{x}) = 1 / \left( \sum_{i \in I} (1 + \beta(\mathbf{x}) \cdot (\mathbf{x} - \mathbf{x}_i)) \right) w_i(\mathbf{x})V_i \\ \beta(\mathbf{x}) = \left[ \sum_{i \in I(\mathbf{x})} (\mathbf{x} - \mathbf{x}_i) \otimes (\mathbf{x} - \mathbf{x}_i)w_i(\mathbf{x})V_i \right]^{-1} \left( \sum_{i \in I(\mathbf{x})} (\mathbf{x} - \mathbf{x}_i)w_i(\mathbf{x})V_i \right) \end{cases}.$$

### 1.3.3 Analyse multi-échelle

**Multiple scale RKPM** Dans le cadre de RKPM, Liu *et al.* se sont intéressés à l'influence de la fonction de référence  $w_{ref}$ . Dans leur article initial [LJZ95], ils montrent même des taux exponentiels de convergence avec un noyau Gaussien corrigé.

L'aspect le plus intéressant est cependant l'analyse multi-échelle qu'ils proposent en se basant sur des classes de noyaux particuliers issus de la théorie des ondelettes. Le principe est d'utiliser comme fonction de référence  $w_{ref}$  une « ondelette père »  $\phi$  (*scaling function*). Décrivons les idées de bases pour la construction de cette fonction (pour plus de détails consulter [Mey90, Dau92, WA94]).

Une ondelette père est définie à partir de la notion d'analyse multirésolution qui est une suite d'espaces emboîtés

$$\{0\} = V_\infty \subset \dots \subset V_1 \subset V_0 \subset V_{-1} \subset \dots \subset V_{-\infty} = \mathcal{L}^2(\mathbb{R})$$

et est caractérisée par « l'équation de dilatation » (*dilation equation*) suivante

$$\exists (a_k)_{k \in \mathbb{Z}} \in l^2(\mathbb{R}), \phi(x) = \sum_{k=-\infty}^{\infty} a_k \phi(2x - k).$$

Cette équation signifie que les fonctions  $\phi_{nm}(x) = 2^{-m/2}\phi(\frac{x}{2^m} - n)$ ,  $n \in Z$  génèrent les espaces  $V_m$ . On définit l'espace  $W_m$  comme le complément de  $V_m$  dans  $V_{m-1}$  (somme *a priori* non-orthogonal)

$$V_{m-1} = V_m + W_m$$

Et de la même façon que précédemment, les espaces  $W_m$  sont générés par les fonctions  $\psi_{mn}(x) = 2^{-m/2}\psi(\frac{x}{2^m} - n)$ ,  $n \in Z$  où  $\psi$  est l'ondelette associée à  $\phi$ .  $\mathcal{L}^2(\mathbb{R})$  est la somme des espaces  $W_m$  qui représentent les fonctions avec une précision croissante

$$\begin{cases} \mathcal{L}^2(\mathbb{R}) = \dots + W_{-1} + W_0 + W_1 + \dots \\ f \in \mathcal{L}^2(\mathbb{R}), f = \dots + w_1 + w_0 + w_{-1} + \dots \end{cases}$$

On utilise la technique usuelle de RKPM pour construire l'ondelette père corrigée  $\bar{\phi}_\rho$  et l'ondelette mère corrigée  $\bar{\psi}_{2\rho}$  définie par

$$\bar{\psi}_{2\rho}(x - x_i) = \bar{\phi}_\rho(x - x_i) - \bar{\phi}_{2\rho}(x - x_i).$$

Par analogie aux ondelettes, l'approximation d'une fonction  $u$  par RKPM peut se décomposer en échelles successives. Pour cela, on définit l'opérateur d'approximation  $P_\rho$  par

$$P_\rho(u)(x) = \sum_{i \in I(x)} \bar{\phi}_\rho(x - x_i) V_i u_i$$

et l'opérateur complémentaire  $Q_{2\rho}$  par

$$\begin{aligned} Q_{2\rho}(u)(x) &= \sum_{i \in I(x)} \bar{\psi}_{2\rho}(x - x_i) V_i u_i \\ &= P_\rho(u)(x) - P_{2\rho}(u)(x) \end{aligned}$$

L'approximation à l'échelle  $\rho$  est donc

$$P_\rho(u) = P_{2\rho}(u) + Q_{2\rho}(u)$$

et l'on peut répéter le processus sur  $P_{2\rho}$  pour avoir une représentation multi-résolution. A partir de ce schéma, il est possible de construire un schéma d'adaptation  $h$ - $p$  où  $P_{2\rho}$  joue le rôle d'indicateur d'erreur [LC95].

Un exemple de famille de fonctions  $\phi$  est donnée par les splines, qui sont les convoluées successives de la fonction de Haar (voir figure 1.5).

**Mise en œuvre de RKPM** La mise en œuvre de RKPM suppose quatre choix :

1. celui de la consistance,
2. celui du noyau,
3. celui du rayon d'influence  $\rho$ ,
4. celui du semis de nœuds.

La consistance de l'approximation est en général choisie en fonction de l'équation à résoudre et du taux de convergence souhaité. En pratique on assure une consistance d'ordre  $m$  pour une équation d'ordre  $2m$  (patch test). Le choix du noyau, de son rayon d'influence et du semis de nœuds est rarement abordé dans la littérature, mais Liu *et al.* propose une analyse dans l'espace des fréquences pour effectuer ces choix en s'appuyant sur des techniques de traitement du signal (échantillonnage) [LC95].

La figure 1.6 montre les transformées de Fourier des fonctions d'atténuation splines (la spline d'ordre  $k$  admet pour transformée  $\hat{f}(\xi) = \left(\frac{\sin(\xi/2)}{\xi/2}\right)^{k+1}$ ). On voit ici que le contenu fréquentiel de ces fonctions est différent : plus l'ordre de la spline est élevé plus la fonction  $\phi$  est passe-bas. A l'extrême, la Gaussienne  $\phi(x) = \exp(-x^2)$  de transformée  $\hat{\phi}(\xi) = \sqrt{\pi} \exp(-\xi^2)$  ne présente pas de « lobes » moyenne et haute fréquence. Ceci explique les taux de convergence très élevés notés dans [LJZ95].

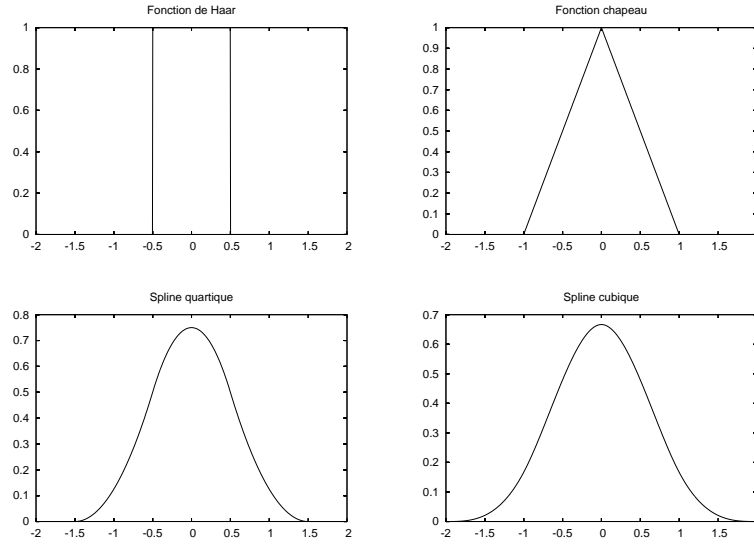


FIG. 1.5: Fonction d'atténuation splines.

### 1.3.4 Base hiérarchique, *hierarchical partition of unity*

On retrouve ici l'idée de pseudo-dérivée dans le cadre RKPM, mais son exploitation est différente. Elle est utilisée pour construire une base de fonctions de forme hiérarchique qui admet de bonnes propriétés d'approximation et permet de résoudre des problèmes difficiles [LL99a, LL99b]. On se place dans  $\mathbb{R}$  dans la suite par souci de simplicité.

Soit  $\bar{u}_x(t) = \sum_{i=0}^n d_i(x) p_i(t-x)$  la fonction ajustée à  $u$  par moindres carrés continus (diffus continu). Nous avons vu dans la première partie que

$$\bar{u}_x(t) = P^\top \left( \frac{t-x}{\rho} \right) M^{-1}(x) \int_{\Omega} P \left( \frac{y-t}{\rho} \right) u(y) \phi_{\rho}(t-x) dy$$

La technique habituelle en moindres carrés glissants pose  $\tilde{u}(x) = \bar{u}_x(x)$ . Pour définir une *partition de l'unité hiérarchique*, les auteurs définissent la fonction

$$u_x^l(t) = \bar{u}_x(t) + \sum_{\alpha=1}^m \frac{C_{\alpha}(x) \rho^{\alpha}}{\alpha!} \frac{d^{\alpha}}{dt^{\alpha}} \left( P^\top \left( \frac{t-x}{\rho} \right) \right) M^{-1}(x) \int_{\Omega} P \left( \frac{y-x}{\rho} \right) u(y) \phi_{\rho}(y-x) dy$$

Cette approximation « peut être vue comme un développement de Taylor tronqué lorsque  $C_{\alpha} = 1, \forall \alpha$  » [LL99a]. L'approximation  $\tilde{u}$  de  $u$  se définit par

$$\tilde{u}(x) = u_x^l(x)$$

Dans la version continue, il vient

$$\left\{ \begin{array}{l} \tilde{u}(x) = \sum_{\alpha=0}^m C_{\alpha}(x) \int_{\Omega} \mathcal{K}^{[\alpha]}(t-x, x) u(t) dt \\ \mathcal{K}^{[\alpha]}(t-x, x) = P^\top \left( \frac{t-x}{\rho} \right) b^{\alpha}(x) \phi_{\rho}(t-x) \\ M(x) b^{\alpha}(x) = P^{(\alpha)}(0) = \left. \left\{ \frac{\rho^{\alpha}}{\alpha!} \frac{d^{\alpha}}{dx^{\alpha}} \left( P \left( \frac{x}{\rho} \right) \right) \right\} \right|_{x=0} \end{array} \right\}$$



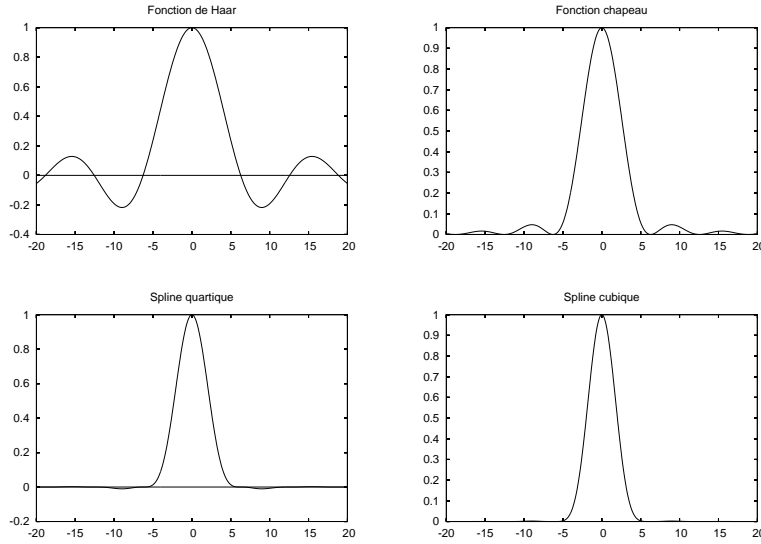


FIG. 1.6: Transformées de Fourier des fonctions d'atténuation splines.

où  $M(x)$  est la matrice des moments de  $\phi_\rho$ , soit

$$M(x) = \int_{\Omega} P^\top \left( \frac{t-x}{\rho} \right) P \left( \frac{t-x}{\rho} \right) \phi_\rho(t-x) dt$$

Les intégrales sont discrétisées comme précédemment, ainsi que le calcul de  $M(x)$  ce qui assure la consistance numérique de l'approximation [LL99a]. L'approximation  $\tilde{u}$  s'écrit

$$\begin{cases} \tilde{u}(x) = \sum_{\alpha=0}^m \sum_{i \in I(x)} \psi_i^{[\alpha]}(x) u_i \\ \psi_i^{[\alpha]}(x) = \mathcal{K}^{[\alpha]}(x_i - x, x) V_i \end{cases}$$

Les fonctions  $\psi_i^{[\alpha]}$  ne sont pas libres dans  $\text{span}(\psi_i^{[\alpha]})_{i\alpha}$ . Une base hiérarchique est une sous-famille libre de la famille  $(\psi_i^{[\alpha]})_{i\alpha}$  [LL99a].

Si la base  $P$  est  $\langle 1, x, \dots, x^m \rangle$ , si les fonctions  $C_\alpha(x)$  sont constantes et égales à 1, et les poids d'intégration  $V_i$  sont tous égaux, alors les fonctions de forme hiérarchiques  $\psi_i^{[\alpha]}$  sont les fonctions de forme diffuses et leur pseudo-dérivées:

$$\forall i, \forall \alpha, \psi_i^{[\alpha]} = \frac{\delta^\alpha N_i}{\delta x^\alpha}.$$

Les fonctions de forme  $\psi_i^{[\alpha]}$  admettent dans le cas général les propriétés de consistance des fonctions de forme diffuse et de leurs pseudo-dérivées.

Cette approximation permet avec un choix particulier des constantes  $C_\alpha$  de synchroniser la convergence dans les espaces  $\mathcal{H}^i(\Omega)$ . C'est-à-dire qu'on a le même taux de convergence pour les dérivées de  $u$  jusqu'à un certain ordre  $p$ . Muni de cette propriété, les auteurs montrent dans un article compagnon [LL99b] des applications aux équations aux dérivées partielles. Les résultats pour l'équation d'Helmoltz par exemple semblent très encourageants.

Cependant, la famille des  $(\psi_i^{[\alpha]})_{i\alpha}$  n'est pas libre, ceci pose des problèmes pour le choix des fonctions à conserver et aucun résultat général n'est donné à ce sujet. Cette difficulté est contournée lorsqu'on utilise les bases hiérarchiques pour des

techniques d'adaptation de type  $p$  (on ajoute les « ondelettes »  $\psi_i^{[\alpha]}$  là où c'est nécessaire et par  $\alpha$  croissant). Notons que cette technique d'adaptation peut être particulièrement simple [LL99b].

## 1.4 Formulation

### 1.4.1 Formulations pour méthodes sans maillage

Les spécificités des méthodes d'approximation que nous avons rencontrées dans la littérature (continuité et consistance élevées, utilisation à 1, 2 ou 3 dimension(s), ... ) permettent d'envisager la plupart des formulations issues des résidus pondérés.

**Collocation** Certains auteurs prétendent que la collocation mène à une « vraie » méthode sans maillage car on n'a pas de découpage sous-jacent pour l'intégration [Alu00]. Cette technique a été la première implémentée [Jen72, LO80], mais elle souffre d'instabilité comme nous l'avons déjà vu.

La technique de collocation double grille permet de limiter la consistance des fonctions de forme prises en compte (on ne dérive chaque famille qu'une seule fois), mais la matrice de raideur obtenue n'est pas symétrique, et le problème d'instabilité n'est pas parfaitement résolu.

**Galerkin, Petrov–Galerkin** La formulation de Galerkin est aujourd'hui la plus employée pour les méthodes sans maillage. Cependant, cette technique n'est pas sans poser quelques problèmes liés à l'intégration et aux conditions aux limites.

L'intégration numérique est nécessaire car les fonctions de forme n'ont pas une forme algébrique simple et connue à l'avance [BTV99a]. Lorsque l'on écrit la formulation variationnelle, les travaux virtuels sont intégrés par parties (voir l'introduction de ce chapitre). Or la pseudo-dérivée ne vérifie pas la formule de Green et il est nécessaire d'utiliser la formulation de Petrov–Galerkin [Lao96, Kro96]. C'est dans ce cadre que l'on peut utiliser les dérivées consistantes présentées précédemment.

**Autres formulations** Il existe aussi d'autres formulations utilisées avec une approximation sans maillage. Nous citons simplement pour référence deux techniques de type éléments frontières :

- *Boundary Node Method* (BNM) [MM97, CMM99],
- *Meshless Local Boundary Integration Equation* (MLBIE) [AZ00],

car BNM n'est pas adaptée aux problèmes non-linéaires et l'intégration nécessaire pour MLBIE est très complexe (intégration sur des sphères).

### 1.4.2 Conditions aux limites

La prise en compte des conditions aux limites de type Dirichlet ne pose pas de problème avec les techniques de collocation car on sait construire des fonctions de forme interpolantes ou utiliser des multiplicateurs de Lagrange. Au contraire, lorsqu'on écrit la formulation variationnelle, les fonctions de test sont de trace nulle sur la frontière. Ce point est important pour l'intégration par parties, car il fait « disparaître » la partie surfacique de la formule. On doit donc autant que possible utiliser des fonctions de forme qui vérifient cette propriété. On trouve quatre techniques de prise en compte des conditions aux limites

1. Par multiplicateurs de Lagrange sans forcer la condition  $N_i|_{\partial\Omega} = 0$  [TBB94].

2. Par couplage avec une couche d'éléments finis [Kro96].
3. Par modification des fonctions de forme diffuses [BV99].
4. Avec interpolation à la frontière [KS97].

La première technique consiste à conserver les fonctions de forme et à imposer les conditions aux limites par des multiplicateurs de Lagrange définis sur la frontière par une interpolation de Lagrange. Cette technique conduit à une formulation modifiée pour prendre en compte les conditions aux limites et à un système avec multiplicateurs de Lagrange, donc plus lourd à résoudre. Une formulation différente permet de construire un système qui conserve une structure bande [LBG94], mais qui reste plus lourd que le système éléments finis usuel.

La seconde technique consiste à utiliser une couche d'éléments finis le long de la frontière  $\Gamma_u$  où l'on impose des conditions aux limites. On profite ainsi des bonnes propriétés des fonctions de forme de l'interpolation par sous-domaines pour prendre en compte les conditions aux limites (elles sont bien dans  $\mathcal{H}_0^1$ ). Le couplage entre fonctions de forme diffuses  $N_i^{dif}$  et fonctions de forme éléments finis  $N_i^{ef}$  est défini par Krongauz [Kro96] et les nouvelles fonctions de forme  $N_i^{cpl}$  sont

$$N_i^{cpl}(\mathbf{x}) = \begin{cases} (1 - v(\mathbf{x}))N_i^{dif}(\mathbf{x}) + v(\mathbf{x})N_i^{ef}(\mathbf{x}) & \mathbf{x} \in \Omega_B \\ N_i^{dif}(\mathbf{x}) & \mathbf{x} \in \Omega_E \end{cases}$$

où  $\Omega_E$  est la région EFG pur, et  $\Omega_B$  est la région de couplage couverte par les éléments finis. La fonction de couplage  $v(\mathbf{x})$  est définie par  $v(\mathbf{x}) = \sum_{i \in I_\Gamma(\mathbf{x})} N_i^{ef}(\mathbf{x})$  avec  $I_\Gamma(\mathbf{x})$  l'ensemble des indices de nœuds de frontière de l'élément dans lequel on se trouve. Ainsi, l'approximation sur la frontière est l'approximation par sous-domaine.

Une troisième technique est proposée par Villon et Breitkopf [BV99]. L'idée est encore de forcer les fonctions de forme à s'annuler sur la frontière de  $\Omega$ . Ici, la technique utilise une modification du critère de construction de l'approximation diffuse le long de la frontière. Nous ne décrivons pas cette méthode car la construction est très complexe.

Enfin on peut aussi utiliser une interpolation diffuse sur la frontière pour imposer aux nœuds uniquement les conditions aux limites [KS97]. Cette technique ne cherche pas à respecter la condition  $v \in \mathcal{H}_0^1$  exactement, mais uniquement à la limite lorsque le nombre de nœuds sur la frontière augmente.

La prise en compte des conditions aux limites reste une question ouverte, car les techniques employées demandent une gestion spécifique de la frontière et le maintien d'une structure compliquée pour le calcul de l'approximation. Dans le cas du couplage éléments finis-approximation diffuse, les fonctions de forme sont difficilement intégrables.

On note ici clairement que les approximations sans maillage doivent utiliser une géométrie de l'objet qui n'était pas nécessaire en éléments finis où les éléments définissent la géométrie par bijection à l'élément de référence. Ce point est à la fois une force (on évite les erreurs de discrétisation à la frontière) et une faiblesse (on sait mal prendre en compte les conditions aux limites) des méthodes sans maillage. Une technique efficace de prise en compte des conditions de Dirichlet reste encore à développer.

## 1.5 Intégration

Le point crucial pour la mise en œuvre de la formulation de Galerkin est l'intégration de la matrice de raideur, de la matrice de masse (dynamique, analyse modale) et du second membre du système discret. En éléments finis, ce problème est

« pré-résolu » par l'approximation mise en œuvre: les sous-domaines sur lesquels est calculée l'interpolation, servent naturellement de pavés d'intégration. Ainsi, les supports d'intégration sont cohérents avec les supports de l'approximation. Ce point est l'une des clefs du succès des éléments finis.

Lorsque l'on utilise une technique d'approximation sans maillage, on n'a pas de support naturel d'intégration car les intersections de supports de fonctions de forme sont parfois très complexes<sup>1</sup>. Ce point fait donc l'objet de plusieurs travaux qui utilisent deux types de techniques :

1. définition d'un pavage et intégration de Gauss sur chaque pavé,
2. intégration nodale.

Nous allons décrire dans la suite les variantes de ces deux techniques. Puis nous décrirons l'analyse du patch test proposée par [Lao96, Kro96, BK98] sur laquelle s'appuiera notre technique d'intégration.

### 1.5.1 Pavage d'intégration

**Pavage aveugle** Dans sa version initiale, EFG s'appuie sur un pavage d'intégration indépendant du semis de nœuds et des supports des fonctions de forme (figure 1.7).

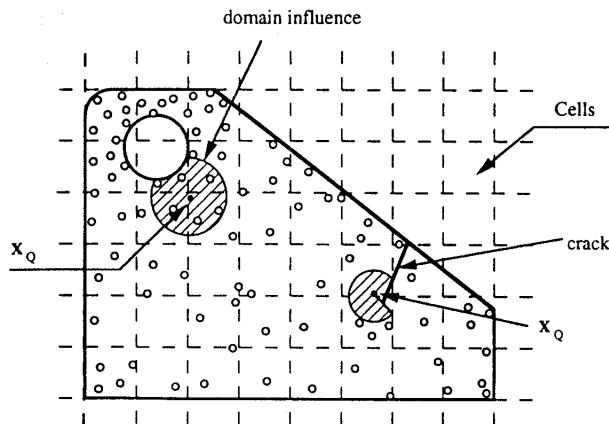


FIG. 1.7: Cellules d'intégration d'après [BLG94]

Cette technique conduit à un algorithme de construction de la matrice de raideur différent des éléments finis puisqu'elle se base sur les cellules d'intégration. Ceci montre bien que le cœur de la formulation variationnelle est l'intégration (voir figure 1.8).

L'intégration d'EFG pose cependant de nombreux problèmes comme le montre [DB99]. Ceci s'explique de la façon suivante: lorsqu'on évalue une intégrale avec la technique de Gauss, on écrit

$$\int_{\Omega_e} f(\mathbf{x}) d\mathbf{x} = \sum_{g=1}^{n_g} \omega_g \bar{f}(\xi_g)$$

où les poids et points d'intégration  $\omega_g, \xi_g$  sont calculés pour un domaine de forme spécifié (et éventuellement de référence, ce qui fait intervenir une bijection entre le

<sup>1</sup>. Le cas extrême est celui des voisins naturels [Suk98] où l'on ne contrôle pas les supports d'approximation qui sont des réunions de disques (sphères) !

1. boucle sur les cellules d'intégration, C
  - (i). boucle sur les points d'intégration (points de Gauss),  $x_g$ .
    - a. si le point d'intégration est à l'extérieur du domaine, aller à e.
    - b. calculer  $I(x_g)$
    - c. calculer les fonctions de forme, leurs dérivées.
    - d. calculer les contributions aux matrices, vecteurs du système linéaire.
    - e. fin si.
  - (ii). fin boucle  $x_g$ .
2. fin boucle C

FIG. 1.8: Séquencement de EFG d'après [BLG94, LBG94]

domaine réel et le domaine de référence). L'erreur d'intégration est donc due à l'écart de la fonction intégrée à sa partie contenue dans l'ensemble des fonctions intégrées exactement (en général des polynômes jusqu'à un certain degré). L'expérience des éléments finis montre que cette erreur est suffisamment faible pour être négligée (éléments isoparamétriques). Or dans le cas de EFG, on fait aussi une erreur sur le domaine d'intégration, et l'expérience prouve que cette erreur est beaucoup plus préjudiciable à la qualité de l'intégration.

Pour montrer l'effet du domaine sur le calcul numérique d'une intégrale, nous avons calculé l'intégrale de l'indicatrice de forme géométrique simple dans le carré  $[-1, 1] \times [-1, 1]$  par la méthode de Gauss-Legendre (figure 1.10), la lenteur de la convergence montre qu'il est nécessaire d'adapter le pavage d'intégration aux supports des fonctions de forme.

Le test de la figure 1.10 est un cas extrême car la fonction intégrée admet une forte discontinuité. Pour illustrer ce résultat de façon intuitive, prenons le cas de l'indicatrice  $I$  de  $[-1, 0]$  sur  $[-1, 1]$  intégrée par la méthode de Gauss-Legendre

$$\mathcal{I}(I) \approx \int_{-1}^1 I(\xi) d\xi = \sum_{g=1}^n \omega_g I(\xi_g)$$

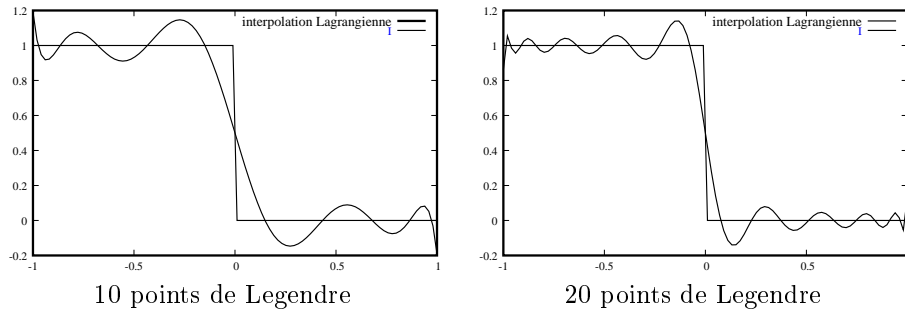
Par construction, la valeur de  $\mathcal{I}(I)$  est l'intégrale de l'approximation Lagrangienne de  $I$  appuyée sur les points de Legendre  $(\xi_g)$ , soit

$$\int_{-1}^1 I(\xi) d\xi = \int_{-1}^1 \sum_{g=1}^n L_g(\xi) f(\xi_g) d\xi, \quad L_g(\xi) = \frac{\prod_{i \neq g} (\xi - \xi_i)}{\prod_{i \neq g} (\xi_g - \xi_i)}$$

et la discontinuité de  $I$  fait apparaître un phénomène de type Gibbs (figure 1.9).

Nous avons donc calculé l'approximation d'une fonction continue dont le support est contenu dans  $[-1, 1] \times [-1, 1]$ . La convergence est ici meilleure, mais le nombre de points de Legendre pour obtenir une bonne approximation est toujours très élevé (400 points pour une erreur relative de  $10^{-5}$ !). Ceci montre comme précédemment que pavage d'intégration et support d'interpolation doivent être cohérents, c'est-à-dire que l'ensemble des nœuds d'influence  $I(x)$  doit être constant sur chaque pavé d'intégration.

**Pavage adapté** Le constat précédent pousse donc à intégrer sur les zones où la liste des nœuds d'influence  $I(x)$  est constante. La forme de ces zones est complexe si

FIG. 1.9: *Approximation Lagrangienne d'un escalier.*

on utilise la distance usuelle pour les fonctions d'influence  $w_i(\mathbf{x}) = w_{ref} \left( \frac{\|\mathbf{x} - \mathbf{x}_i\|_2}{r_i} \right)$ : ce sont des intersections de disques (2d)! Les auteurs proposent donc d'utiliser les fonctions d'influence définies par produit tensoriel [DB99]

$$w_i(\mathbf{x}) = w_{ref} \left( \frac{x - x_i}{r_i^x} \right) w_{ref} \left( \frac{y - y_i}{r_i^y} \right).$$

Ainsi les zones de liste d'influence constante peuvent se découper en rectangle (technique des boîtes d'encombrement).

L'utilisation du pavage d'intégration adapté permet une nette amélioration des résultats de EFG. Cependant, le patch test n'est passé que pour un très grand nombre de points de Gauss. Ce résultat n'est pas surprenant si l'on s'intéresse à la forme des fonctions de forme MLS: dans le cas 1d avec consistance linéaire, on a pour  $n$  nœuds [BTV99a]

$$N_i(x) = \frac{\sum_{j \neq i} w_i(x) w_j(x) (x - x_j) (x_i - x_j)}{d}, \quad d = \sum_{j=1}^{n-1} \sum_{j=i}^n w_i(x) w_j(x) (x_i - x_j)^2$$

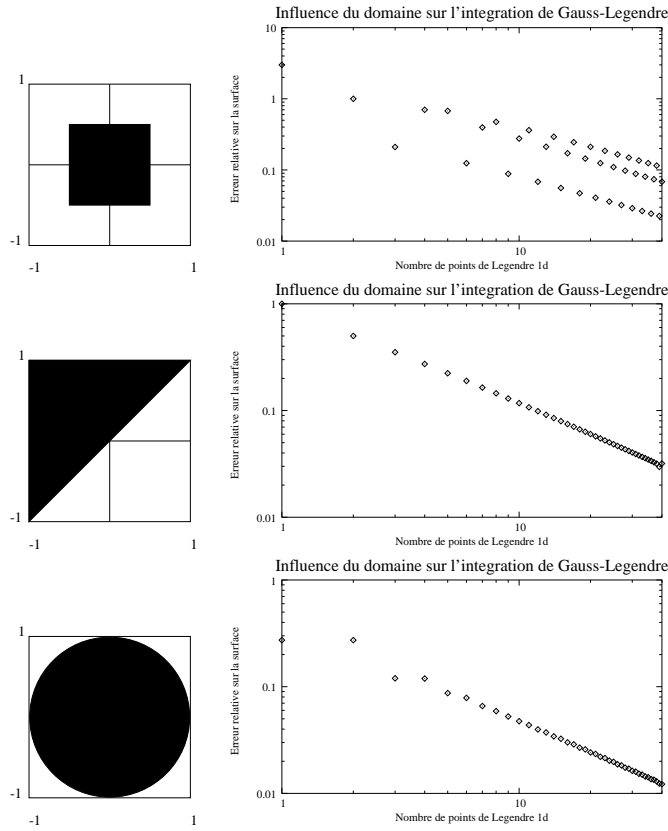
La fonction de forme n'est pas un polynôme mais une fonction rationnelle avec  $w_{ref}$  polynomiale. Intégrons sur  $[-1, 1] \times [-1, 1]$  deux fonctions rationnelles simples à titre d'exemple (figure 1.12):

$$\begin{cases} f(\xi, \eta) &= \frac{1}{(\xi+2)(\eta+2)} \\ g(\xi, \eta) &= \frac{1}{(\xi+2)^2(\eta+2)^2} \end{cases}$$

On note sur ce simple test une remarquable corrélation avec les résultats présentés dans [DB99] pour le patch test. Un schéma d'intégration efficace pour les méthodes sans maillage doit être capable d'intégrer efficacement les fonctions rationnelles. Nous verrons au chapitre suivant comment construire une telle intégration. De plus, la construction des pavés d'intégration proposée est lourde, nous définirons aussi une technique inverse et plus simple (on ajuste les supports des fonctions de forme aux pavés d'intégration).

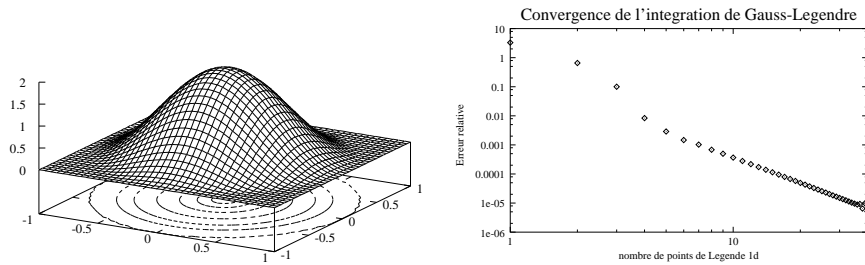
### 1.5.2 Intégration nodale

Un schéma d'intégration basé sur un pavage peut engendrer des difficultés pour les grands déplacements, car les cellules d'intégration dégénèrent. On retrouve ici les difficultés des éléments finis lorsqu'il est nécessaire de remailler. Pour contourner ce problème, une intégration nodale de la formulation variationnelle discrète a été proposée par Beissel et Belytschko [BB96].



$$\begin{cases} S(\text{forme}) = \int_{[-1,1] \times [-1,1]} I_{\text{forme}}(x) dx = \sum_{(\xi_i, \xi_j) \in \text{forme}} \omega_i \omega_j \\ \xi_i \text{ points de Legendre, } \omega_i \text{ poids de Gauss associés} \end{cases}$$

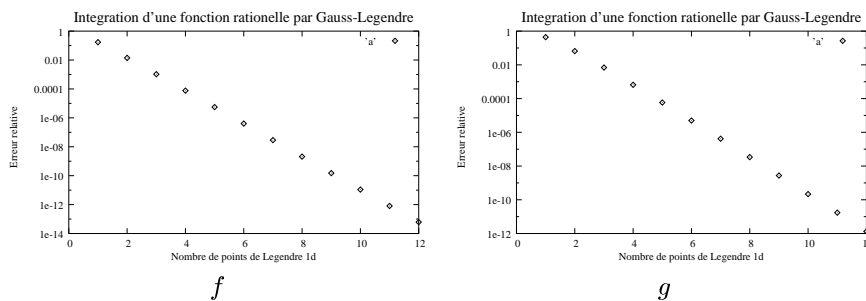
FIG. 1.10: Intégrale de l'indicatrice d'un cercle, d'un carrée et d'un triangle, étude de la convergence du schéma de Gauss-Legendre tensoriel. En abscisse, le nombre de points de Legendre 1d.



$$f(\xi, \eta) = 1 + \cos(\pi \sqrt{\xi^2 + \eta^2})$$

$$\int_{[-1,1] \times [-1,1]} f(\xi, \eta) dS = \pi - \frac{4}{\pi}$$

FIG. 1.11: Convergence de l'intégration de Gauss-Legendre tensorielle – Cas d'une fonction continue de support le disque unité.

FIG. 1.12: *Intégration de fonctions rationnelles par la méthode de Gauss–Legendre*

**Schéma de base** L'idée de base pour une intégration nodale est simple: on utilise la définition Riemannienne de l'intégrale. La discrétisation de la fonction inconnue  $u$  s'appuie sur un ensemble de nœuds  $(x_i)_{i \in I}$  dans  $\Omega$ ; à chaque nœud, on associe un volume élémentaire  $q_i^\Omega$  tel que la somme de ces volumes soit celui de  $\Omega$ . Une intégrale sur  $\Omega$  est approchée par

$$\int_{\Omega} f(\mathbf{x}) \, d\Omega = \sum_{i \in I} q_i^\Omega f(\mathbf{x}_i).$$

En pratique, les auteurs choisissent

$$q_i^\Omega = \frac{f_\Omega^i \rho_i^n}{\sum_{i \in I} f_\Omega^i \rho_i^n} A_\Omega$$

avec  $\rho_i$  le rayon d'influence du nœud  $i$ ,  $f_\Omega^i$  la fraction de la zone d'influence dans  $\Omega$  et  $n$  la dimension de l'espace. D'autres volumes élémentaires peuvent être proposés qui sont mieux adaptés à de grands maillages, mais cela ne change pas le principe de l'intégration nodale [BB96].

L'intégration nodale est particulièrement bien adaptée aux méthodes sans maillage. En particulier, les problèmes avec très grandes déformations comme le forgeage sont traités sans remaillage ou équivalent pour les cellules d'intégration [BK98]. Le but semble atteint, mais une difficulté apparaît: l'intégration nodale ne filtre pas certains modes parasites.

**Modes parasites (*spurious modes*)** Le phénomène des modes parasites est bien connu en différences finies ou pour des éléments sous-intégrés. Ce sont des solutions d'énergie nulle, leur présence signifie que la formulation discrète du problème variationnel n'est pas définie (matrice de raideur  $K$ ). Bonet et Kulasegaram mettent en évidence le même phénomène pour la méthode SPH corrigée décrite précédemment [BK98] et la figure 1.13 montre un exemple avec EFG et intégration nodale.

On élimine les modes parasites en discriminant les différentes solutions d'énergie nulle par l'ajout d'un terme de stabilisation. Ce terme de stabilisation n'est autre que la fonctionnelle utilisée dans la méthode des moindres carrés, soit pour l'équation de la chaleur dans  $\mathbb{R}^n$  (problème 1.5, page 141)

$$\begin{cases} \Pi(u) = \Pi_0(u) + \Pi_{st}(u) \\ \Pi_{st}(u) = \frac{1}{2} \eta \rho^n \int_{\Omega} (-\Delta u + \alpha u + f)^2 \, d\Omega \end{cases}$$

où  $\eta$  est un facteur adimensionnel et  $\rho^n$  assure l'homogénéité. Le minimum de  $\Pi$  est celui de  $\Pi_0$ . La figure 1.13 montre que l'on stabilise bien la solution. Il est nécessaire de calculer des dérivées secondes et donc d'assurer une consistance quadratique de la base d'approximation. Ceci entraîne un élargissement de la matrice de raideur



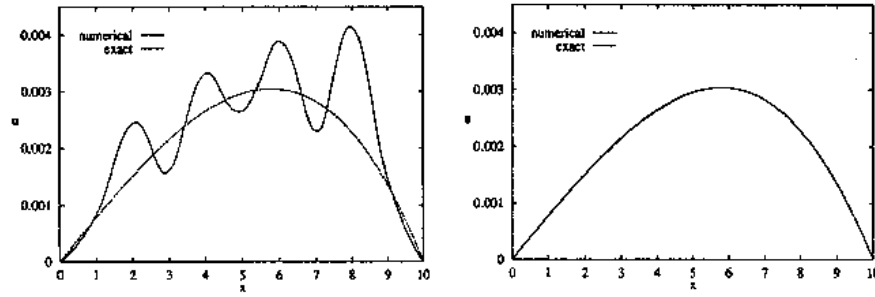


FIG. 1.13: Modes parasites et stabilisation d'après [BB96]

$K$ . Une correction du type dérivée constante est proposée dans [BK98] pour le Laplacien, ce qui évite ce problème.

L'intégration nodale se heurte à une autre difficulté: le patch test n'est pas passé exactement mais seulement asymptotiquement. Il faut donc construire une correction de l'intégration ou des gradients calculés pour passer le patch test. Bonet et Kulasegaram construisent un gradient modifié

$$\tilde{\nabla}u(\mathbf{x}_i) = \nabla u(\mathbf{x}_i) + \gamma_i \llbracket u \rrbracket_i$$

avec  $\llbracket u \rrbracket_i$  le défaut d'interpolation de SPH, soit

$$\llbracket u \rrbracket_i = u_i - \tilde{u}(\mathbf{x}_i).$$

Les coefficients  $\gamma_i$  sont calculés de façon à satisfaire les conditions du patch test que nous allons décrire dans la section suivante. Ils sont solution d'un système d'équations linéaires. Finalement, la méthode SPH avec correction du noyau et intégration nodale corrigée permet de simuler le forgeage.

Notons que cette correction de l'intégration nodale ne peut être mise en œuvre avec une interpolation. Il faut donc trouver une autre technique pour bénéficier de l'égalité entre dérivée diffuse et dérivée usuelle aux nœuds d'une interpolation diffuse.

### 1.5.3 Le patch test

Une analyse du patch test est proposée dans [Lao96, Kro96, BK98] pour l'équation de la chaleur ou l'élasticité linéaire. Les différents auteurs montrent des résultats similaires; nous en proposons ici une argumentation générale.

Soit un problème différentiel (voir [ZT97])

$$S^\top(HSu) = Au = f$$

où  $S$  est un opérateur différentiel et  $H$  une matrice de paramètres. Sa formulation variationnelle s'écrit: trouver  $u \in \mathcal{H}$  telle que

$$\forall v \in \mathcal{H}, a(u, v) = l(v).$$

Après discrétisation et intégration numérique, on doit résoudre le problème approché (de solution  $u^*$ ): trouver  $u \in V_n$  tel que

$$\forall v \in V_n, a^*(u^*, v) = l^*(v)$$

Les formes approchées bilinéaire  $a^*$  et linéaire  $l^*$  sont obtenues à partir de  $a$  et  $l$  en remplaçant la vraie intégration  $\int$  par une intégration numérique  $\mathcal{f}$  et (éventuellement) la vraie dérivation  $\frac{\partial}{\partial x_i}$  par une pseudo-dérivation  $\frac{\delta}{\delta x_i}$ , sachant que les deux

opérateurs d'intégration et de dérivation approchés restent linéaires. La forme bilinéaire symétrique définie positive  $a$  étant fixée, à chaque choix de la forme linéaire  $l$  correspond un problème et une solution unique  $u_{ex}$ .

**Condition nécessaire et suffisante pour passer le patch test** Soit  $P$  l'ensemble des fonctions  $u$  telles que  $Su$  soit constant. Le patch test vérifie que tout problème admettant une fonction de  $P$  comme solution est résolu exactement avec les opérateurs approchés  $a^*, l^*$ , i.e.  $u^* = u_{ex}$  si  $u_{ex} \in P$ . L'espace de recherche  $V_n$  doit donc toujours contenir  $P$  pour passer le patch test.

**Hypothèse 6** Soit  $N_i$  une base de  $V_n$ , et  $I_{int}$  l'ensemble des indices des nœuds intérieurs, nous supposons que :

**H1** Le problème discret est bien posé (i.e.  $K = [a^*(N_i, N_j)]_{i,j \in I_{int}}$  est inversible).

**H2** Si  $A^{-1}f \in P$ , alors  $l(N_i) = l^*(N_i)$ .

Or  $u^* = u_{ex}$  implique que pour tout  $i$  dans  $I$ ,  $a^*(u_{ex}, N_i) = l^*(N_i)$  grâce à H1. Ceci démontre que sous H1, H2, on passe le patch test si et seulement si pour tout  $p$  dans  $P$ , pour tout  $i$  dans  $I$ ,  $a^*(p, N_i) = a(p, N_i)$ . Cependant, pour les problèmes usuels (élasticité, équation de la chaleur) si  $p$  est dans  $P$ , alors  $Ap = 0$ . Ainsi l'hypothèse H2 est vérifié avec une intégration de type Gauss ou une intégration nodale.

**Théorème 12** Si pour tout  $p$  dans  $P$ ,  $Ap = 0$ , alors le problème discrétisé passe le patch test si et seulement si

$$\forall p \in P, \forall i \in I, a^*(p, N_i) = a(p, N_i)$$

**Conséquences** Si  $AP = \{0\}$ , pour une base  $e_i$  de  $P$  et une base  $N_j$  de  $V_n$ , il vient

$$\forall i, \forall j, a^*(e_i, N_j) = a(e_i, N_j) \quad (1.12)$$

Par ailleurs, pour que le problème admette toujours une solution unique dans  $V_n$ , il faut et il suffit que  $a^*$  (qui est symétrique) soit définie (H1). Cette condition est équivalente à imposer l'inversibilité de la matrice  $K^*$  telle que  $K_{ij}^* = a^*(N_i, N_j)$  et dans laquelle ont été prises en compte les conditions aux limites (cas d'une interpolation et d'une base Lagrangienne).

Pour l'équation de la chaleur ( $\alpha = 0$ ), le patch test minimal est la reproduction exacte des solutions linéaires. Ceci impose donc aux fonctions de forme une consistance linéaire ainsi qu'à leurs dérivées premières. Les conditions (1.12) sont des tautologies pour les constantes. Pour les fonctions linéaires, il vient aux nœuds intérieurs (le cas des nœuds extérieurs est réglés par les conditions aux limites) :

$$\forall j, \int_{\Omega} \nabla(N_j) \cdot \mathbf{c} \, d\Omega = \int_{\Omega} \nabla(N_j) \cdot \mathbf{c} \, d\Omega$$

où  $\mathbf{c}$  est un vecteur constant et quelconque dans  $\mathbb{R}^n$ . D'après la formule de Green,

$$\int_{\Omega} \nabla(u) \, d\Omega = \int_{\partial\Omega} nu \, dS$$

et en se souvenant que les fonctions de forme  $N_i$  sont locales, il vient finalement la seconde condition pour passer le patch test

$$\forall i, \mathbf{x}_i \text{ nœuds intérieur, } \int_{\Omega} \nabla(N_i) \, d\Omega = \int_{\partial\Omega} nN_i \, dS \quad (1.13)$$

Ceci nous permet d'écrire le théorème suivant :

**Théorème 13** *Soit la discrétisation de la formulation variationnelle de l'équation de la chaleur (1.5) avec condition de Dirichlet par une interpolation Lagrangienne de fonction de forme  $N_i$ , et  $I_{int}$  l'ensemble des indices des nœuds intérieurs. Deux conditions nécessaires pour passer le patch test sont :*

1. *La matrice  $K$  définie par,  $K_{ij} = \int_{\Omega} \nabla(N_i) \cdot \nabla(N_j) d\Omega$ ,  $i, j \in I_{int}$ , est inversible.*
2. *L'intégration numérique  $\mathfrak{f}$  et la dérivation  $\frac{\delta}{\delta x_i}$  sont liées par l'équation (1.13) :*

$$\forall i \in I_{int}, \mathfrak{f}_{\Omega} \nabla(N_i) d\Omega = \mathfrak{f}_{\partial\Omega} n N_i dS$$

Dans le cas 1d ( $\Omega = ]a, b[$ ,  $a < b$ ), l'équation (1.13) s'écrit pour une interpolation

$$\forall i \in I_{int}, \mathfrak{f}_{\text{supp}(N_i) \cap ]a, b[} N'_i dx = 0,$$

et pour une approximation

$$\forall i \in I_{int}, \mathfrak{f}_{\text{supp}(N_i) \cap ]a, b[} N'_i dx = N_i(b) - N_i(a).$$

Nous exploiterons cette condition dans le prochain chapitre pour construire une intégration robuste et qui assure le patch test. Le chapitre suivant généralisera ce résultat à une équation d'ordre 4 qui régit la flexion des poutres.

## 1.6 Conclusion

En conclusion, le nombre de méthodes (techniques?) sans maillage est très important, et le choix de l'une d'entre elles difficile. Un premier constat pousse vers les formulations de Galerkin et Petrov–Galerkin car la collocation n'est pas stable.

En second lieu, on doit choisir une approximation. À la vue des options possibles, il semble que les moindres carrés glissants discrets sont les mieux formulés et les plus flexibles pour les équations différentielles :

- c'est une approximation discrète,
- elle admet une consistance exacte d'ordre quelconque  $m$ ,
- il reste des libertés sur les supports des fonctions de forme qui peuvent être locaux.

Ils permettent aussi divers schémas d'adaptation (bases hiérarchiques, hp–clouds). Mais ce point n'a pas été approfondi car d'autres aspects ne sont pas encore satisfaisants.

Dans la formulation de Galerkin, on doit être attentif à l'approximation sur la frontière. Nous avons vu que les solutions proposées ne sont pas satisfaisantes. De la même façon, l'intégration numérique utilisée en pratique n'assure pas de passer le patch–test. Une première piste est d'adapter le pavage d'intégration aux supports des fonctions de forme. Cependant l'algorithme est complexe.

La mise en œuvre d'une méthode sans maillage suppose donc une avancée significative dans chacun des deux domaines :

- Conditions aux limites.

– Intégration numérique.

Nous avons choisi ici de nous pencher sur l'intégration numérique pour plusieurs raisons :

1. Il est possible de tester l'intégration numérique sans connaître de solution pour les conditions aux limites (1d, domaines de forme simples en 2d, 3d).
2. On ne peut pas tester d'idées sur les conditions aux limites sans intégration performante.
3. Les conditions aux limites font l'objet de travaux à l'UTC (P. Villon, P. Breitskopf).
4. On peut traiter des problèmes industriels avec des modèles 1d (poutre).

Nous allons donc proposer dans la suite une technique d'intégration numérique efficace et robuste basée sur le patch test. De nombreuses variantes de techniques MLS sont testées et montrent les qualités de cette solution pour l'équation de la chaleur en 1d (chapitre suivant). Nous étendons ensuite la définition de l'intégration numérique à des barres (poutre rectiligne, équation du quatrième degré, chapitre 3) ce qui nous permet finalement de traiter un modèle de forage (chapitre 4).



## Chapitre 2

# Intégration numérique

Le patch test a souvent été utilisé pour la construction d'éléments non conformes. Pour les approximations sans maillage, la continuité de l'approximation est assurée, mais la mise au point d'une bonne stratégie d'intégration est difficile. Nous proposons une technique d'intégration qui respecte les conditions nécessaires pour passer le patch test et qui s'appuie sur deux piliers :

1. mise en cohérence du pavage d'intégration et des supports de fonctions de forme,
2. intégration des dérivées des fonctions de forme.

Ces deux fondements sont exactement les points faibles des techniques proposées dans la littérature. Les principes de construction de notre intégration numérique permettent de faire varier l'approximation (approximation ou interpolation, type de dérivation) et les supports des fonctions de forme (en 1d). Ceci permet de construire de nombreuses variantes de méthodes sans maillage. Pour tester ces méthodes, nous commençons par définir deux problèmes type qui sont le problème de Dirichlet et le calcul de modes propres. Les modes propres montrent bien l'approximation de l'opérateur continu par l'opérateur discret issu de l'approximation. La deuxième partie décrit l'intégration numérique spéciale et une méthode de résolution du système résultant qui est singulier. Nous décrivons ensuite la construction des fonctions d'influence pour assurer des pavés d'intégration de liste d'influence constante et ceci permet de répertorier les méthodes testées. Enfin des résultats numériques sont présentés.

On considère  $\Omega = ]0, 1[$  et un semis de nœuds  $(x_i)_{i \in [0, n]}$  tel que pour  $i$  dans  $[1, n - 1]$ ,  $x_i$  appartienne à  $\Omega$  et  $x_0 = 0$ ,  $x_n = 1$ . On suppose la suite  $(x_i)$  croissante.

## 2.1 Problèmes types

### 2.1.1 Problème de Dirichlet

Nous allons résoudre le problème de Dirichlet (1.5) :

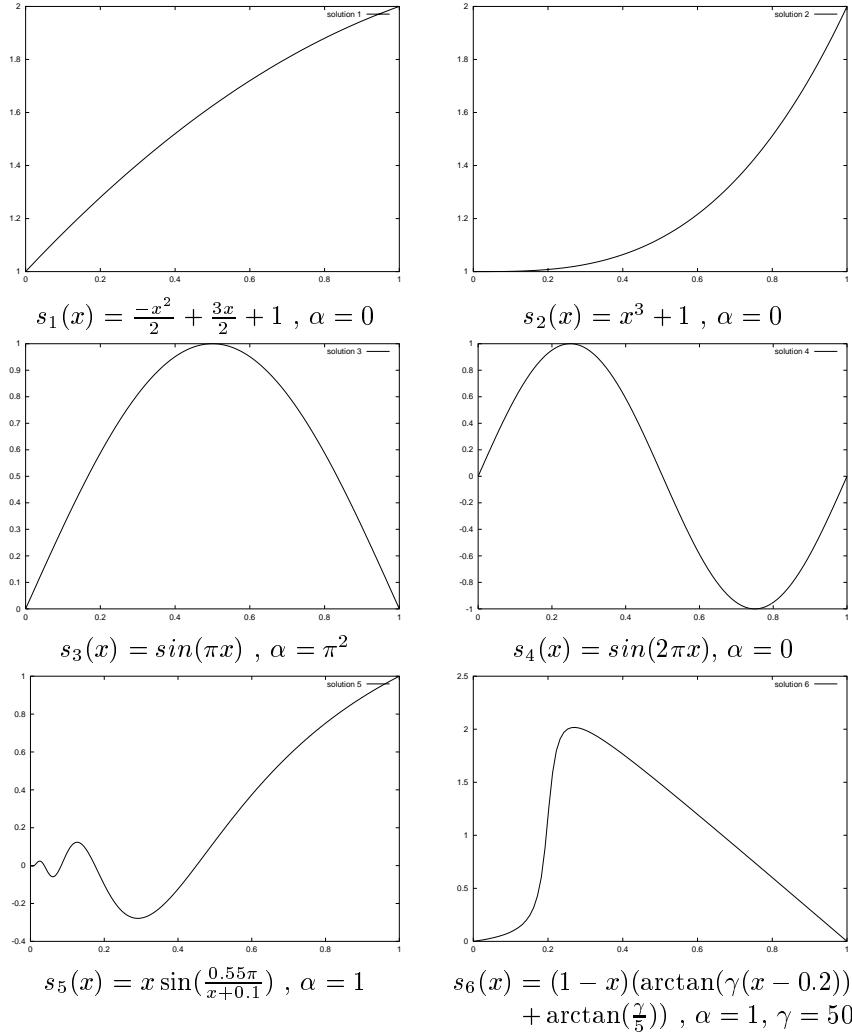
$$\begin{cases} -u'' + \alpha u = f \text{ sur } ]0, 1[ \\ u(0) = u_0 \\ u(1) = u_1 \end{cases}$$

pour différentes valeurs de  $\alpha$  et différents termes sources  $f$ , car cette équation est le prototype des équations elliptiques.

Les six cas testés sont reportés dans la table 2.1. Les cas 1 et 2 admettent comme solutions des polynômes de degrés respectifs 2 et 3. Les solutions des cas 3

et 4 sont des sinus de pulsation  $\pi$  (demi-période) et  $2\pi$  (période complète). Enfin les solutions des cas 5 et 6 correspondent à des cas « complexes ».

Nous avons ajouté à cela le patch test pour  $\alpha = 0$  et  $\alpha = 1$  car il nous permet d'analyser les résultats obtenus pour chaque méthode.



TAB. 2.1: Solutions et paramètres des problèmes traités (le second membre  $f_i$  est calculé pour que  $s_i$  soit solution).

### 2.1.2 Modes propres

Rappelons pour commencer l'origine des problèmes spectraux (en nous inspirant largement de [Cia98, RT98]). Pour cela, on considère l'équation des ondes dans une corde de longueur unité (le segment  $]0,1[$ )

$$\begin{cases} \frac{\partial^2 u}{\partial t^2} - \frac{\partial^2 u}{\partial x^2} = 0 \\ u(0) = 0 \\ u(1) = 0 \end{cases} \quad (2.1)$$

On peut chercher une solution de ce problème sous la forme d'un produit

$u(x, t) = w(x)\varphi(t)$  (séparation des variables). L'équation devient alors

$$w''(x)\varphi(t) - w(x)\varphi''(t) = 0$$

c'est-à-dire

$$\frac{w''(x)}{w(x)} = \frac{\varphi''(t)}{\varphi(t)} = -\lambda, \quad \lambda > 0 \text{ constant}$$

Le problème est donc ramené à chercher les valeurs propres ( $\lambda_k$ ) et vecteurs propres ( $u_k$ ) de l'opérateur  $-\frac{d^2}{dx^2}$ , et résoudre l'équation

$$\varphi''(t) + \lambda\varphi(t) = 0$$

De la même façon que pour le problème de Dirichlet, on cherche une approximation de ces valeurs/vecteurs propres dans  $V_n$  par le biais d'une formulation variationnelle. Le problème linéaire résultant est (sans les conditions aux limites)

$$\left\{ \begin{array}{l} \text{trouver } U \text{ dans } \mathbb{R}^{n+1} \text{ et } \lambda > 0 \text{ tels que} \\ KU = \lambda MU \\ \text{avec } \left\{ \begin{array}{l} K_{ij} = \int_0^1 \frac{\delta N_i}{\delta x} \frac{\delta N_j}{\delta x} dx \quad (i, j) \in \llbracket 0, n \rrbracket^2 \\ M_{ij} = \int_0^1 N_i N_j dx \quad (i, j) \in \llbracket 0, n \rrbracket^2 \end{array} \right. \end{array} \right. \quad (2.2)$$

La résolution de ce type de problème montre bien l'efficacité de chaque méthode d'interpolation pour la construction de l'opérateur discret  $K$ , car les solutions sont des modes propres de l'opérateur traité (ici le Laplacien 1d).

Dans notre cas particulier, les valeurs propres, vecteurs propres sont les fonctions  $(\sqrt{2} \sin(k\pi x))_{k \in \mathbb{N}}$  associées à  $(k^2\pi^2)_{k \in \mathbb{N}}$ .

## 2.2 Construction d'une intégration numérique

On cherche une formule d'intégration efficace du point de vue numérique, en posant

$$\int_a^b f(x)dx = \frac{b-a}{2} \int_{-1}^1 \bar{f}(\xi)d\xi = \frac{b-a}{2} \sum_{g=1}^{n_g} \omega_g \bar{f}(\xi_g)$$

avec  $\bar{f}(\xi) = f(\frac{b-a}{2}\xi + \frac{a+b}{2})$ .

On mesure la qualité d'une telle intégration par l'estimation de l'intégrale de fonctions simples. En général, ce sont les polynômes : la condition d'intégration des polynômes de degré  $m$  s'écrit matriciellement  $G\omega = \gamma$  et  $e^\top \omega = 2$  avec

$$\left\{ \begin{array}{l} k \in [1, m], g \in [1, n_g], G_{kg} = \xi_g^k, \gamma_k = \int_{-1}^1 \xi^k d\xi \\ e = \langle 1, \dots, 1 \rangle^\top \end{array} \right.$$

Dans la méthode classique de Gauss-Legendre les poids  $\omega_g$  doivent vérifier les conditions avec  $m = 2n_g - 1$ . En revanche dans la méthode que nous proposons ici, par le fait même que nous sommes obligés d'imposer les conditions issues du patch-test, nous nous contenterons de chercher à minimiser  $\|G\omega - \gamma\|$ .

### 2.2.1 Construction

Nous nous plaçons dans la suite sur un intervalle  $]a, b[ \subset ]0, 1[$  de liste d'influence  $I(x)$  constante et relié à  $] -1, 1[$  par transformation affine ( $x = \frac{b-a}{2}\xi + \frac{a+b}{2}$ ). Nous



abandonons la notation  $\bar{f}(\xi)$  car cela ne soulève aucune ambiguïté et  $I(x)$  est renuméroté  $[1, n_i]$ .

Les contraintes du patch test sont respectées dès que l'on impose sur tous les pavés d'intégration (avec  $\frac{\delta}{\delta x}$  la dérivée diffuse ou usuelle)

$$\left\{ \begin{array}{l} \forall i \in [1, n_i], \int_a^b \frac{\delta N_i}{\delta x}(x) dx = N_i(b) - N_i(a) \\ \text{ou de façon équivalente} \\ \forall i \in [1, n_i], \int_{-1}^1 \frac{\delta N_i}{\delta \xi}(\xi) d\xi = N_i(1) - N_i(-1) \end{array} \right.$$

Cela impose la condition  $e^\top \omega = 2$ . En effet, multiplions chaque équation par  $x_i$ , sommions sur  $i$  et explicitons l'intégration numérique, il vient :

$$\frac{b-a}{2} \sum_{g=1}^{n_g} \omega_g \underbrace{\sum_{i=1}^{n_i} x_i \frac{\delta N_i}{\delta x}(x_g)}_1 = \underbrace{\sum_{i=1}^{n_i} x_i N_i(b)}_b - \underbrace{\sum_{i=1}^{n_i} x_i N_i(a)}_a$$

les conditions de consistance permettent de transformer cette équation en

$$\sum_{g=1}^{n_g} \omega_g = 2$$

Nous pouvons imposer séparément cette condition qui signifie que les constantes sont intégrées exactement.

Compte tenu de  $e^\top \omega = 2$ , les conditions du patch test s'écrivent pour tout  $i$ ,  $\sum_{g=1}^{n_g} D_{gi} \omega_g = 0$ , ou  $D^\top \omega = 0$  avec

$$i \in [1, n_i], g \in [1, n_g], D_{gi} = \frac{\delta N_i}{\delta \xi}(\xi_g) - \frac{N_i(1) - N_i(-1)}{2}.$$

Ainsi, en posant  $A = \{\psi \in \mathbb{R}^{n_g}, e^\top \psi = 2 \text{ et } D^\top \psi = 0\}$ , la construction de l'intégration numérique consiste à résoudre le problème de minimisation :  $\omega = \operatorname{argmin}_{\psi \in A} (\|G\psi - \gamma\|^2)$ . Ce qui est équivalent au système linéaire symétrique suivant : Trouver  $\psi \in \mathbb{R}^{n_g}, \lambda \in \mathbb{R}$  et  $\mu \in \mathbb{R}^n$  tels que

$$\begin{bmatrix} G^\top G & e & D \\ e^\top & 0 & 0 \\ D^\top & 0 & 0 \end{bmatrix} \begin{bmatrix} \omega \\ \lambda \\ \mu \end{bmatrix} = \begin{bmatrix} G^\top \gamma \\ 2 \\ 0 \end{bmatrix} \quad (2.3)$$

### 2.2.2 Propriétés du système résultant

Nous allons montrer que la matrice  $D$  n'est pas de rang maximal. A cette fin, il suffit de montrer que les  $n$  contraintes  $\sum_{g=1}^{n_g} D_{gi} \omega_g = 0$  ne sont pas indépendantes. Par définition de  $D$ , ces équations sont

$$\int_{-1}^1 \frac{\delta N_i}{\delta \xi} d\xi = N_i(1) - N_i(-1).$$

Les fonctions de forme  $N_i$  ont une consistance polynômiale jusqu'à un certain degré. Considérons alors un polynôme  $p$  tel que  $p(\xi) = \sum_{i=1}^{n_i} N_i(\xi) p(\xi_i)$  et  $\frac{\delta p}{\delta \xi} = \frac{dp}{d\xi}$  et combinons les  $n_i$  équations ci-dessus avec les coefficients  $p(\xi_i)$ , il vient :

$$\int_{-1}^1 \frac{dp}{d\xi} d\xi = p(1) - p(-1).$$

Or cette égalité est vraie lorsque le degré de  $p$  est 0 (on obtient une évidence  $0 = 0$ ) et aussi lorsque le degré de  $p$  est 1 comme précédemment (car on impose  $e^\top \omega = 2$ ). Le système  $D^\top \omega = 0$  présente une redondance d'ordre au moins 2. Cela nous amène à utiliser une méthode particulière pour résoudre le système linéaire 2.3.

### 2.2.3 Résolution du système d'optimalité

**Pivot de Gauss modifié** En mécanique, les multiplicateurs de Lagrange associés aux déplacements imposés sont les forces de réactions. Ainsi, lorsque les contraintes sont redondantes, la force associée à l'équation supplémentaire est nulle. Ceci nous fournit un algorithme de résolution du système (2.3) : On applique un pivot de Gauss avec recherche du pivot maximal lorsque le pivot est nul. Comme la matrice  $G^T G$  est inversible (car  $G$  contient une matrice de Vandermonde de points distincts), on n'aura un pivot nul que dans les équations associées aux multiplicateurs de Lagrange. Ainsi, lorsque tous les pivots possibles sont nuls, on annule les inconnues restantes, c'est-à-dire les multiplicateurs de Lagrange associés aux contraintes redondantes. Il suffit ensuite de résoudre un système de la forme

$$\begin{bmatrix} T & A \\ 0 & I \end{bmatrix} \begin{Bmatrix} \eta \\ \mu_{nul} \end{Bmatrix} = \begin{Bmatrix} s \\ 0 \end{Bmatrix}$$

où  $\eta = \langle \omega, \mu_{inc} \rangle^T$ ,  $I$  est l'identité, et  $T$  est triangulaire supérieure. Ainsi, on ne résoud que le système  $T\eta = s$  après triangulation.

**Orthogonalisation de la contrainte** Une alternative possible est d'utiliser la forme particulière que nous avons donnée à la contrainte  $D^T \omega = 0$ . On cherche ainsi les solutions  $\omega$  dans le noyau de  $D^T$ . On calcule une base de ce noyau par orthogonalisation grâce à la décomposition  $QR$  de  $D$ .

En effet, il existe pour toute matrice, une matrice de permutation  $P$ , une matrice orthogonale  $Q$  et une matrice triangulaire supérieure  $R$  telle que  $DP = QR$ . Cette décomposition peut se calculer à partir de la transformée de Householder, et un sous-produit de l'algorithme est le rang  $r$  de  $D$  (voir par exemple [GL89], pages 235 et 245). Lorsque la matrice n'est pas de rang maximal, la matrice  $R$  prend la forme suivante ( $T$  est triangulaire supérieure, de taille  $(r, r)$  et de rang  $r$ ).

$$R = \begin{bmatrix} T & R' \\ 0 & 0 \end{bmatrix}$$

Ainsi, la contrainte  $D^T \omega = 0$  se transforme successivement en  $P^T D^T \omega = 0$ , car  $P$  est orthogonale, soit  $R^T Q^T \omega = 0$ . D'après la forme particulière de  $R$ , les contraintes se traduisent par

$$\{Q^T \omega\}_j = 0, \forall j \in \llbracket 1, \dots, r \rrbracket$$

Et donc le noyau de  $D^T$  a la forme suivante,

$$\text{Ker}(D^T) = \left\{ \omega = Q \begin{Bmatrix} 0 \\ \eta \end{Bmatrix}, \eta \in \mathbb{R}^{n_g - r} \right\} = \{ \bar{Q} \eta, \eta \in \mathbb{R}^{n_g - r} \}$$

où  $\bar{Q}$  est la matrice des  $n_g - r$  derniers vecteurs de  $Q$ .

Finalement, le calcul des poids d'intégration se ramène au problème équivalent

$$\begin{cases} \min \|G\bar{Q}\eta - g\|^2 \\ e^T \bar{Q}\eta = 2 \end{cases}$$

qui se résout facilement. Une variante consiste à écrire les équations de reproduction pour  $n_g - r$  monômes, ce qui permet de calculer le vecteur  $\eta$ .

**Pseudo-inverse** On peut aussi utiliser la méthode de la pseudo-inverse (voir *singular value decomposition* dans [GL89] pages 239–243 et 257) pour calculer la solution du système (2.3) directement.

**Choix d'un algorithme de résolution** La seconde méthode a été choisie pour nos tests numériques. Les trois principales raisons de ce choix sont :

1. La pseudo-inverse n'a pas de sens pour le problème traité car les poids  $\omega_g$  sont traités de la même façon que les multiplicateurs de Lagranges  $\lambda, \mu$ .
2. Le pivot de Gauss modifié impose  $m \geq n_g$  car la matrice  $G^\top G$  doit être inversible. Pour la méthode par orthogonalisation, la contrainte est  $m \geq n_g - r$ .
3. La décomposition  $QR$  est robuste (voir l'algorithme proposé dans [GL89], pages 211–221) et on contrôle bien le rang de  $D$ .

## 2.3 Méthodes testées

La construction d'une méthode de résolution des problèmes proposés repose sur le choix d'une technique d'approximation et de l'intégration associée. L'utilisation de l'approximation diffuse laisse donc libre les paramètres suivants :

- La partition de  $\Omega$  et la construction des fonctions d'influence  $w_i$ .
- Le type d'approximation diffuse (approximation/interpolation, type de dérivation).
- L'intégration numérique (Gauss–Legendre classique ou modifiée).

Dans un premier temps, nous décrivons les différentes constructions des fonctions d'influence, puis nous donnerons la liste des méthodes testées. La figure 2.2 fixe le vocabulaire.

### 2.3.1 Construction des fonctions d'influence

Le domaine  $\Omega$  est un segment  $]a, b[$ , dans lequel on connaît un semis de nœuds  $(x_i)_{i \in [1, n-1]}$ , avec  $x_0 = a$  et  $x_n = b$ . La famille d'ouvert pour la construction de la partition de l'unité est indexée sur les nœuds, et l'on suppose que pour tout  $i$  de  $[[0, n]$ ,  $x_i$  appartient à  $\Omega_i$ . On peut proposer deux types de couverture de  $\Omega$  :

- (i) Centrée aux nœuds,  $\Omega_i$  est la boule  $B(x_i, \rho_i)$  de centre  $x_i$  et de rayon  $\rho_i$ .
- (ii)  $\Omega_i$  est un segment  $]x_i^w, x_i^e[$ , avec  $x_i^w < x_i < x_i^e$

C'est le premier type de couverture qui est utilisée dans la littérature. Un raffinement consiste à modifier la norme pour obtenir des boules de formes simples (norme infinie dans  $\mathbb{R}^n$ ). Dans cette formulation, le problème revient à calculer les rayons d'influence  $\rho_i$ . Des algorithmes de construction de ces rayons sont proposés dans [DO96] par exemple.

**Construction** Les fonctions poids  $w_i$  sont définies par  $w_i(x) = w_{ref}(\frac{|x-x_i|}{\rho_i})$ , où  $w_{ref}$  est une « cloche » sur  $[-1, 1]$ . La fonction

$$w_{ref}(\xi) = \begin{cases} 0 & \text{si } |\xi| > 1 \\ 1 - 6|\xi|^2 + 8|\xi|^3 - 3|\xi|^4 & \text{sinon} \end{cases}$$

en est un exemple, elle sera utilisée dans les tests.

Dans le cas de couverture plus générale  $\Omega_i = ]x_i^w, x_i^e[$ , les fonctions  $w_i$  sont asymétriques et définies par

$$w_i(x) = w_{ref}(\xi_i(x))$$

où  $\xi_i$  est une bijection (strictement) croissante et suffisamment continue de  $[x_i^w, x_i^e]$  dans  $[-1, 1]$  tel que

$$\xi_i(x_i^w) = -1, \quad \xi_i(x_i) = 0, \quad \xi_i(x_i^e) = 1$$

Différentes formes de fonction  $\xi_i$  existent, leurs espaces de validité se mesurent avec le rapport  $\frac{|x_i - x_i^w|}{|x_i - x_i^e|}$ . Il est possible de construire des bijections polynomiales.

Les fonctions d'influence doivent assurer l'existence de l'approximation : en 1d, pour une consistance d'ordre  $m$  et pour conserver la continuité de l'approximation, la fonction  $I(x)$  doit vérifier les deux conditions suivantes :

1.  $\forall x \in \Omega, \text{Card}(I(x)) \geq m + 1,$
2.  $\{x, \text{Card}(I(x)) = m + 1\}$  est de mesure nulle.

Il est possible de proposer une solution minimale dans le sens où  $\text{Card}(I(x)) = m + 2$  presque partout, elle est décrite en annexe B. Nous désignerons cette construction : « support de type Voronoï ».

Guyon montre par ailleurs [Guy95b] que l'on peut construire par croissance différents pavages. Les supports de Voronoï peuvent être utilisés pour calculer des rayons d'influence  $\rho_i$  par croissance. Cette idée s'applique aussi dans  $\mathbb{R}^n$  [Guy95b].

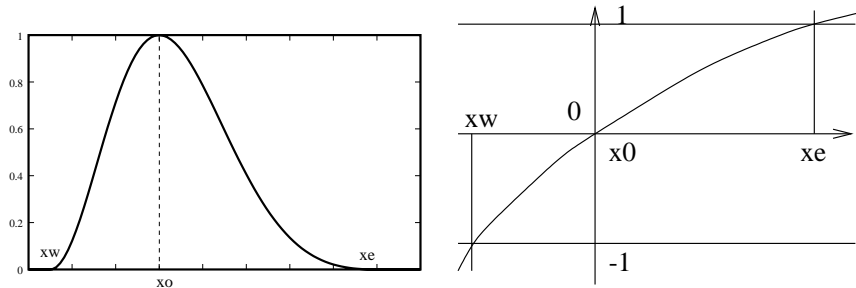


FIG. 2.1: Fonction  $w_i$  asymétrique pour le triplet  $[x_w, x_o, x_e]$  et bijection de  $[x_w, x_o, x_e]$  vers  $[-1, 0, 1]$ .

**Adaptation au pavage d'intégration** Nous avons vu que l'intégration numérique demande d'adapter les supports de fonctions de forme au pavage d'intégration. En général, une grille régulière (en 1d ce sont des segments de longueurs égales) sert de support à l'intégration. Pour adapter  $\Omega_i$  à ce pavage, on tire sur ses extrémités jusqu'à la frontière de pavé la plus proche. Les ouverts  $\Omega_i$  prennent alors la forme  $]x_i^w, x_i^e[$ . La figure 2.3 illustre le cas 2d : la zone d'influence du nœud  $x_i$  est la boîte  $]x_i^w, x_i^e[ \times ]y_i^s, y_i^n[$  et la fonction d'influence est le produit tensoriel des fonctions définies sur les segments  $]x_i^w, x_i^e[$  et  $]x_i^w, y_i^n[$ , soit

$$w_i(\mathbf{x}) = w_i^x(x)w_i^y(y)$$

**Semis de nœuds** La construction de la partition de l'unité s'appuie sur un semis de nœuds connu *a priori*. Pour l'étude de la convergence des méthodes testées dans la suite, ce semis de nœuds est calculé par perturbation d'un maillage régulier. On considère les points  $x_i = ih + r_i$ ,  $h = \frac{1}{n}$ ,  $i \in [1, n - 1]$ , où  $r_i$  est une variable aléatoire uniforme dans  $[-\frac{a}{h}, \frac{a}{h}]$ , et  $a$  est un paramètre de valeur type 0.1.

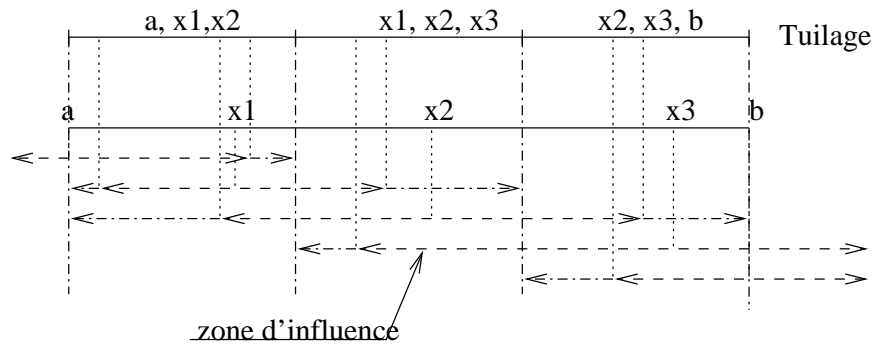


FIG. 2.2: Segment  $]a, b[$  et sa discrétisation à 5 nœuds  $(a, x_1, x_2, x_3, b)$ . Le segment supérieur représente le pavage d'intégration ou **tuilage**. Lorsque la **zone d'influence** (le support de  $w_i$ ) d'un nœud intersecte une tuile, ce nœud est dit **nœud d'influence de la tuile**. Le prolongement des zones d'influence représente le processus d'adaptation au tuilage.

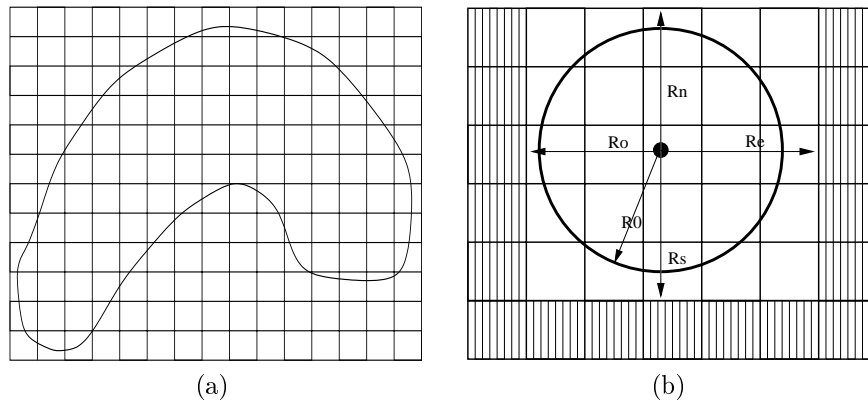


FIG. 2.3: (a) domaine et grille sous-jacente, (b) ajustement du support de la fonction de pondération à la grille (les cellules non-hachurées sont le nouveau support de la fonction de forme).

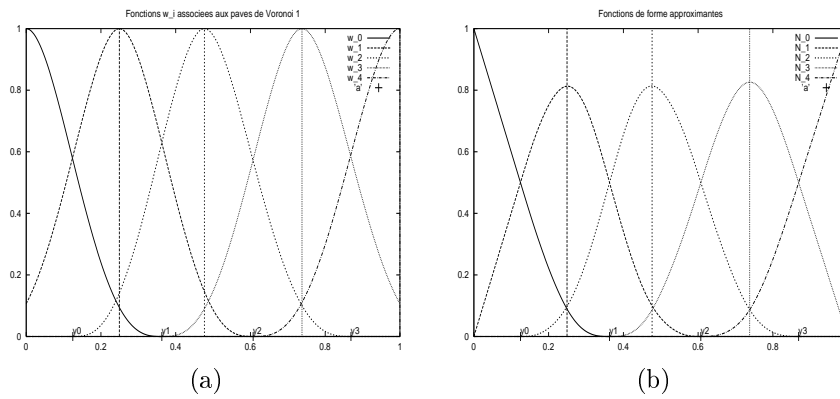


FIG. 2.4: Fonctions  $w_i$  et fonctions de forme  $N_i$  associées à une discrétisation à 5 nœuds de  $[0, 1]$ . Sur les pavés extérieurs, l'interpolation est linéaire car on n'a que deux nœuds d'influence. On assure cependant les propriétés de continuité de l'approximation diffuse.

### 2.3.2 Liste des méthodes testées

Nous venons de voir que de nombreux choix sont possibles pour la mise en œuvre d'une méthode sans maillage dans le cadre variationnel. Ces choix concernent la construction de  $V_n$ , et le pavage d'intégration :

**Construction de  $V_n$**  Dans le cadre des moving least squares, on peut construire une base de fonctions interpolantes ou approximantes. De la même façon, on peut dériver par une dérivée réelle (EFG) , ou utiliser le pseudo-dérivateur (diffus). De la même façon, trois choix de supports de fonctions de forme sont possibles: boules centrées aux nœuds avec calcul des rayons décrit dans [DO96], pavés de Voronoï, et enfin supports adaptés au pavage d'intégration. Cette dernière construction sera appelée tuilage dans la suite.

**Intégration** On distingue deux pavages d'intégration, la grille régulière, et les pavés de Voronoï. Deux méthodes d'intégration sont testées. La méthode de Gauss-Legendre d'une part, et la méthode d'intégration adaptée d'autre part.

Ceci nous amène donc à la liste des méthodes testées du tableau 2.2. Notons que les méthodes 1 à 6 sont généralisables à 2 et 3d. La construction mixte de  $V_n$  est celle de la figure 2.4; c'est-à-dire que l'on utilise des fonctions de formes approximantes à l'intérieur de  $[a, b]$ , et interpolantes pour les nœuds de bords.

|    | $V_n$   | Dérivation | Support         | Pavage | Intégration |
|----|---------|------------|-----------------|--------|-------------|
| 1  | approx. | réelle     | $(x_i, \rho_i)$ | Grille | G.-L.       |
| 2  | interp. | réelle     | $(x_i, \rho_i)$ | Grille | G.-L.       |
| 3  | interp. | réelle     | adaptés         | Tuile  | G.-L.       |
| 4  | interp. | réelle     | adaptés         | Tuile  | adaptée     |
| 5  | interp. | diffuse    | adaptés         | Tuile  | G.-L.       |
| 6  | interp. | diffuse    | adaptés         | Tuile  | adaptée     |
| 7  | interp. | réelle     | Voron.          | Voron. | G.-L.       |
| 8  | interp. | réelle     | Voron.          | Voron. | adaptée     |
| 9  | interp. | diffuse    | Voron.          | Voron. | G.-L.       |
| 10 | interp. | diffuse    | Voron.          | Voron. | adaptée     |
| 11 | mixte   | réelle     | Voron.          | Voron. | G.-L.       |
| 12 | mixte   | réelle     | Voron.          | Voron. | adaptée     |
| 13 | mixte   | diffuse    | Voron.          | Voron. | G.-L.       |
| 14 | mixte   | diffuse    | Voron.          | Voron. | adaptée     |

TAB. 2.2: Méthodes sans maillage testées. La méthode de référence éléments finis linéaires avec intégration de Gauss-Legendre (G.-L.) pour le second membre est ajouté à cette liste. Les conditions aux limites sont prises en comptes des multiplicateurs de Lagrange. La première méthode est la méthode EFG.

## 2.4 Analyse des résultats

Toutes les simulations ont été effectuées en double précision, c'est-à-dire pour une précision relative machine de  $\epsilon_m = 2,220446.10^{-16}$  (arithmétique IEEE). Autrement dit,  $1 + x \neq 1$  dès que  $|x| \geq \epsilon_m$  et  $1 + x = 1$  si  $|x| < \epsilon_m$ . L'enjeu des tests que nous allons décrire est la convergence des méthodes proposées. Pour cela, on trace la courbe qui représente le logarithme d'une mesure de l'erreur en fonction du logarithme du nombre de degrés de liberté du modèle (« dof » pour *degree of freedom*). Les théorèmes de convergence prévoient que la courbe obtenue est une droite, dont la pente est le **taux de convergence** de la méthode.

### 2.4.1 Critères de comparaison

**Problème de Dirichlet** On connaît une expression analytique des solutions des problèmes de Dirichlet que nous avons traités. Ainsi, on peut calculer la norme  $\mathcal{L}^2$  et la semi norme<sup>1</sup>  $\mathcal{H}^1$  de la fonction erreur  $e(x) = u_{ex}(x) - u(x)$ . Ces calculs font intervenir une intégrale, nous avons choisi d'utiliser la même intégration numérique que dans le calcul de la matrice de raideur pour chaque méthode.

Comme ce choix d'intégration rend difficile la comparaison entre les méthodes utilisant différentes intégrations (quelle part de l'erreur donner à l'intégration?), nous avons utilisé la norme  $l^2$ : les nœuds sont des points particuliers de  $[0, 1]$ , une mesure possible de l'erreur  $\epsilon$  est donc la suivante

$$\epsilon^2 = \frac{1}{n} \sum_{i=0}^n (u_{ex}(x_i) - u(x_i))^2$$

Cette norme n'est pas satisfaisante non plus, car les éléments finis reproduisent exactement (au sens de l'intégration de Gauss-Legendre) les valeurs nodales lorsque  $\alpha = 0$  (voir [SF73, Ton69]). Notons que l'on utilise bien  $u(x_i)$  et non pas  $u_i$  pour ne pas pénaliser artificiellement les approximations au profit des interpolations.

Une quatrième norme d'erreur est la norme énergie, c'est-à-dire avec les notations habituelles  $\frac{1}{2}U^T KU - U^T F = \frac{-1}{2}(U^T F + \Lambda^T S)$ . Cependant, cette norme d'erreur souffre du même problème que l'erreur nodale.

Finalement nous avons choisi de n'utiliser que la normes  $\mathcal{L}^2$  et la semi norme  $\mathcal{H}^1$  car

- la norme  $\mathcal{L}^2$  montre bien la convergence de la grandeur calculée (ici les valeurs nodales  $u_i$ );
- la norme  $\mathcal{H}^1$  est la norme naturelle pour la formulation variationnelle et est la somme de la norme  $\mathcal{L}^2$  et de la semi norme  $\mathcal{H}^1$ :  $\|e\|_{\mathcal{H}^1}^2 = \|e\|_{\mathcal{L}^2}^2 + \|e'\|_{\mathcal{L}^2}^2$ .

**Modes propres** Dans les problèmes dynamiques, l'information importante est constituée par les valeurs propres, car elles vont donner les fréquences propres des structures. Par ailleurs, ce sont les basses fréquences qui sont les plus énergétiques, et donc ce sont celles qui nous intéressent le plus.

Nous considérerons donc les erreurs  $\frac{|\lambda_k - (k\pi)^2|}{(k\pi)^2}$  pour les premières valeurs propres ( $\lambda_k^{ex} = k\pi$ ). Comme les vecteurs propres et valeurs propres jouent un rôle très particulier pour l'opérateur mis en jeu, on estime souvent que la convergence des valeurs propres mesure la convergence de  $K$  vers l'opérateur continu.

---

1. Nous utiliserons dans la suite l'expression « norme  $\mathcal{H}^1$  » par abus de langage



### 2.4.2 Intégration, première analyse

Le calcul de l'intégration fait intervenir plusieurs paramètres, dont

- Les points d'intégration.
- Le nombre de points d'intégration.
- Le nombre de tuiles (ou pavés).

Dans un premier temps, nous avons utilisé des points réguliers sur le pavé d'intégration. L'expérimentation a vite montré que l'emploi des points de Legendre est meilleur.

On veut intégrer exactement les fonctions de forme sur les pavés d'intégration, c'est-à-dire que l'on impose au moins  $n_i - 1$  conditions aux poids de Gauss (car il y a au moins une équation redondante). A ceci s'ajoute la condition  $\sum \omega_g = 2$ . Il faut donc au moins autant de points d'intégration que de nœuds. Comme on veut minimiser l'écart à l'intégration des polynômes, on a nécessairement  $n_g > n_i$  pour préserver des degrés de liberté.

La procédure de calcul des poids de Gauss, avec contrainte d'intégration des dérivées des fonctions de forme est efficace comme nous allons le voir dans la suite. Cependant, dans certains cas, les poids de Gauss sont négatifs, on n'assure plus la définie positivité de l'intégration. Cette situation apparaît lorsqu'on utilise la pseudo-dérivation. L'expérience montre qu'il suffit d'avoir  $K$  inversible pour assurer la convergence de la méthode.

La figure 2.5 montre pour commencer que l'intégration adaptée permet bien de construire une intégration numérique qui vérifie la première condition pour passer le patch test. Tous paramètres égaux par ailleurs, cette figure montre que l'intégration adaptée est beaucoup plus performante que l'intégration de Gauss-Legendre du point de vue de ce critère.

Dans tous nos tests numériques, nous avons ajouté à la matrice  $D$  les lignes qui correspondent à l'intégration exacte des monômes pour lesquels l'approximation est consistante. En pratique, nous utilisons des approximations de consistance linéaire, donc la matrice  $D$  a la ligne supplémentaire  $\sum \omega_g \xi_g = 0$ . Cet ajout n'est pas nécessaire, mais nous semblait naturel pour assurer la cohérence des propriétés de l'intégration et de l'approximation.

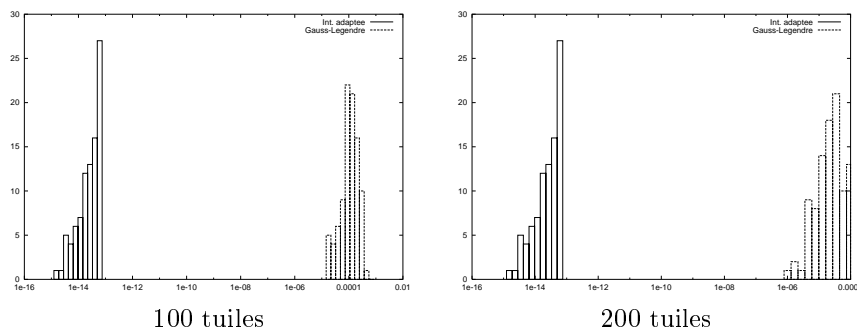
### 2.4.3 Sensibilité aux différents paramètres

**Nombre de monômes ( $m$ )** Nous avons vu que le nombre de monômes utilisés pour le calcul de l'intégration adaptée n'est que minoré par  $n_g - r$ . Ceci laisse une grande liberté de choix sur ce paramètre. Comme on cherche à rester efficace du point de vue CPU, il semble naturel dans un premier temps de choisir  $m = n_g - r + 1$ . Un autre choix possible est  $m = n_g + 1$  pour avoir une matrice  $G$  régulière bien que cela ne soit pas nécessaire.

La figure 2.6 montre la convergence pour une intégration par tuiles avec interpolation diffuse et pseudo-dérivation. La technique d'adaptation de  $m$  est moins performante que le choix systématique  $m = n_g + 1$  de deux points de vue :

1. régularité de la convergence,
2. qualité de la simulation pour de faibles discrétisations.

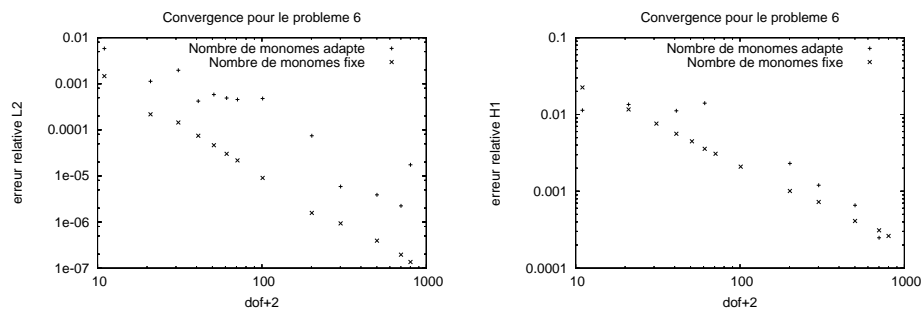
Nous choisirons donc dans la suite  $m = n_g + 1$ .



Description des paramètres :

- Les fonctions  $N_i$  sont de consistance linéaire.
- 6 points de Gauss pour l'intégration de Gauss-Legendre.
- $n_g = n_i + 2$  points de Gauss par tuile pour l'intégration adaptée.
- $n_i$  moyen sur chaque tuile :
  - 100 tuiles : 4 ( $\max=5, \min=3$ ).
  - 200 tuiles : 3 ou 4 ( $\max=5, \min=2$ ).
- nombre de monômes  $m = n_g + 1$ .

FIG. 2.5: Histogramme des valeurs de  $\int_{\Omega} N_i^j(x) dx$ , avec tuilage et dérivée réelle. Ce test s'appuie sur 100 nœuds quasi-aléatoires dans  $[0, 1]$  (perturbation 10%).



Description des paramètres :

- Interpolation diffuse et pseudo-dérivation.
- Tuilage, même nombre de tuile que de nœuds.
- $n_g = n_i + 2$ .
- $m = n_g - r + 1$  ou  $m = n_g + 1$ .

FIG. 2.6: Convergence comparée pour les deux choix de construction de la matrice  $G$  (perturbation 10%).

**Nombre de tuiles ( $n_t$ ) et nombre de points de Gauss ( $n_g$ )** Nous avons étudié la sensibilité à ces deux paramètres pour les méthodes généralisables à 2 et 3d. Les résultats des trois méthodes suivantes sont représentatifs :

- intégration de Gauss–Legendre avec dérivée réelle (EFG),
- intégration adaptée avec dérivée réelle (tuiles),
- intégration adaptée avec dérivée diffuse (tuiles diffuses).

Pour chacun de ces cas, nous avons fait varier le nombre de pavés d'intégration et le nombre de points d'intégration. Les résultats sont présentés sur les figures 2.7, 2.8 et 2.9, le semis de 52 nœuds utilisé a été généré par perturbation (10%) comme décrit précédemment. L'approximation est l'interpolation diffuse.

Ces tests montrent que le nombre de tuiles a une forte influence sur la qualité du résultat ; ceci est particulièrement vrai lorsqu'on utilise directement une intégration de Gauss–Legendre. Le nombre de points d'intégration a une influence moins importante, particulièrement dans le cas de l'intégration adaptée.

Nous devons noter que ces deux paramètres ne sont pas indépendants. En effet, le nombre total de points de Legendre est le produit du nombre de tuiles par le nombre de points d'intégration par tuile. Or nous avons vu que le support d'intégration doit être choisi très attentivement, c'est cet effet que l'on retrouve : plus le nombre de tuiles augmente, plus leur liste d'influence est homogène et plus l'effet de support diminue. Cependant, le support n'est pas la seule origine de l'erreur. Raffiner le pavage d'intégration ou adapter pavage d'intégration et supports des fonctions de forme n'est pas suffisant. L'intégration adaptée est aussi nécessaire.

Notons enfin que les tests menés avec les pavés de Voronoï montrent de la même manière que le nombre de points de Gauss par tuile a une faible influence. Ce paramètre est délicat à ajuster, car nous avons remarqué expérimentalement deux effets inverses :

- Lorsque le nombre de points supplémentaires augmente, le second membre est mieux intégré,
- Les conditions du patch test sont mieux vérifiées avec peu de points supplémentaires.

La première constatation signifie simplement que l'on a plus de degrés de liberté pour minimiser  $\|G\Omega - \gamma\|$  et donc mieux intégrer la partie polynomiale des fonctions (développement de Taylor). La seconde constatation est plus délicate. Il semble qu'elle soit simplement due à un bruit numérique dans les algorithmes que nous avons utilisés. La figure 2.10 montre les poids d'intégration pour  $n_g - n_i = 3$  pour la pseudo-dérivée et la dérivée réelle. L'écart aux poids de Gauss–Legendre diminue lorsque l'on utilise la dérivée au sens usuel.

Dans la suite, nous allons utiliser les paramètres suivants

- autant de tuiles que de nœuds,
- 6 points de Legendre par tuile pour l'intégration de Gauss–Legendre,
- 3 points supplémentaires pour l'intégration adaptée,
- un monôme de plus que de points de Legendre.

Le nombre de points d'intégration total de toutes les méthodes testées est ainsi presque identique d'une méthode à l'autre.

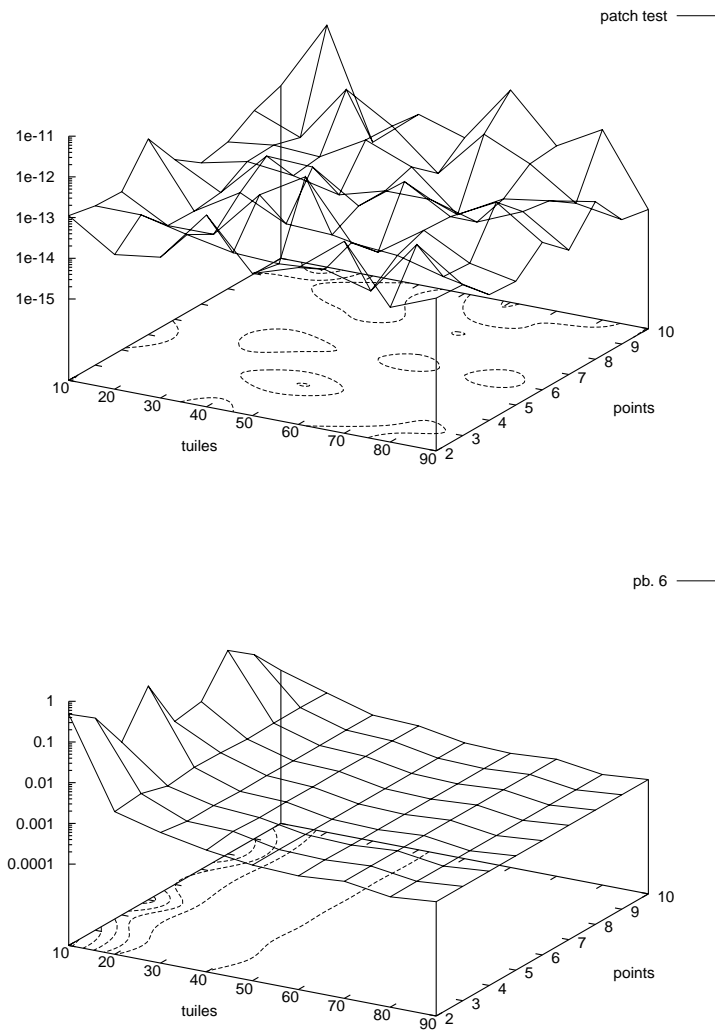


FIG. 2.7: Influence du nombre de tuiles et du nombre de points d'intégration supplémentaires par tuile ( $n_i - n_g$ ). Cas de la dérivée réelle sur le patch test et le problème 6. L'erreur utilisée est l'erreur nodale, car elle ne fait pas intervenir l'intégration numérique.

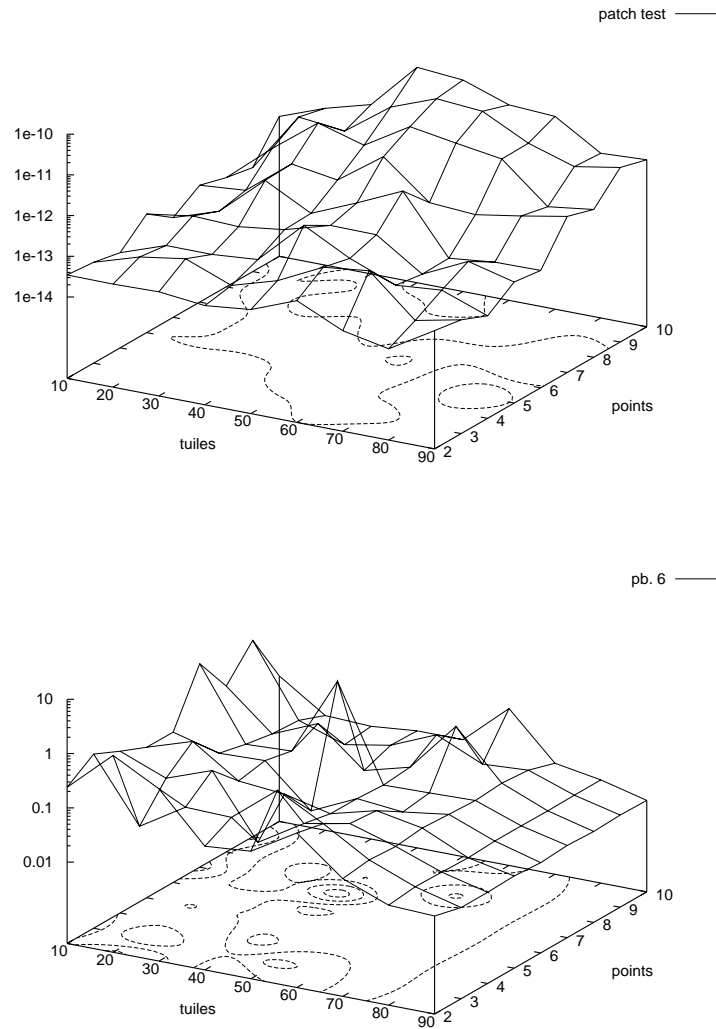


FIG. 2.8: Influence du nombre de tuiles et du nombre de points d'intégration supplémentaires par tuile ( $n_i - n_g$ ). Cas de la dérivée diffuse sur le patch test et le problème 6. L'erreur utilisée est l'erreur nodale, car elle ne fait pas intervenir l'intégration numérique.

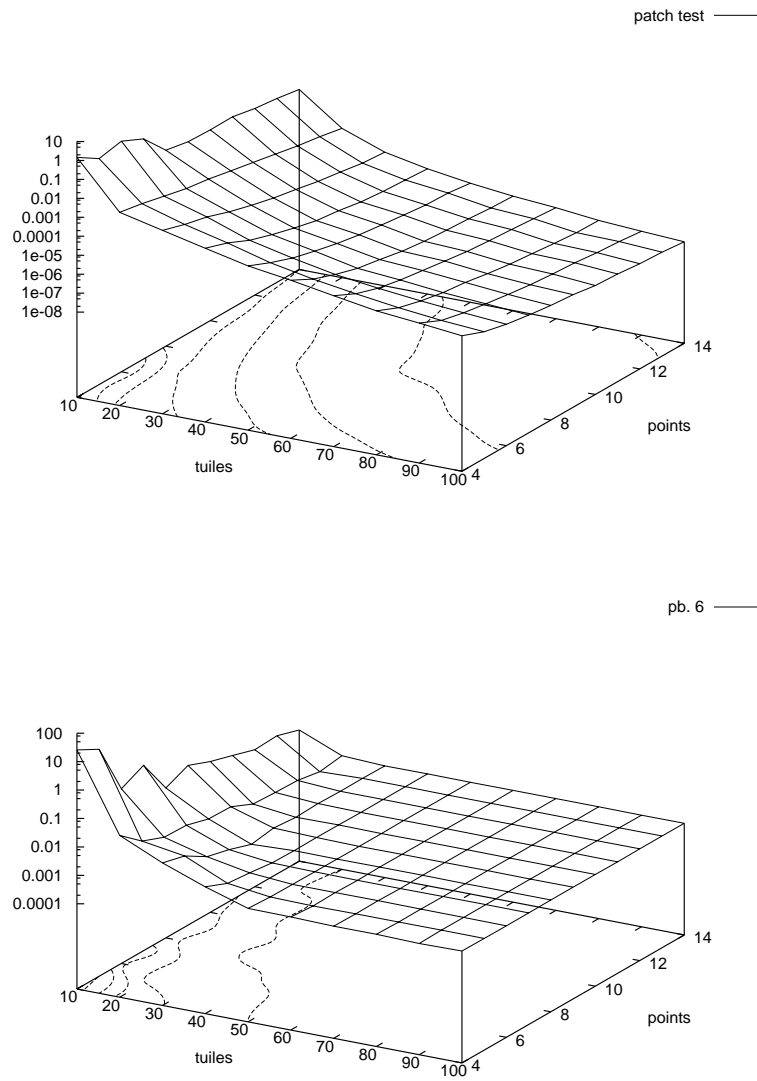


FIG. 2.9: Influence du nombre de tuiles et du nombre de points d'intégration par tuile. Cas de la méthode EFG interpolante sur le patch test et le problème 6. L'erreur utilisée est l'erreur nodale.

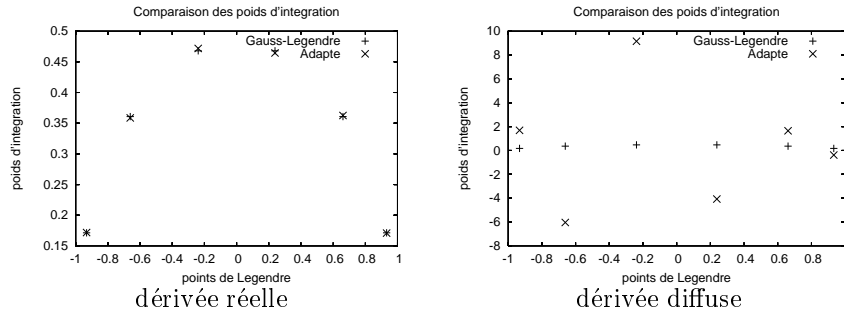


FIG. 2.10: Poids d'intégration pour les mêmes fonctions de forme avec dérivée réelle ou diffuse.

**Semis de nœuds** La répartition des nœuds dans  $\Omega$  a une influence sur les fonctions de forme, et en particulier leurs gradients. La figure 2.11 montre un tel cas.

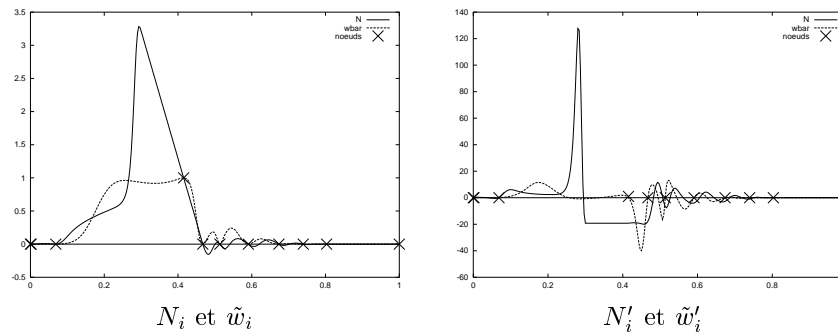


FIG. 2.11: Les fonctions de forme présentent de forte variations lorsque le semis de nœuds est trop irrégulier ( $\tilde{w}$  est définie en annexe B).

L'intégration de telles fonctions de forme est délicate, c'est pourquoi certains spécialistes préfèrent utiliser des semis de nœuds réguliers. De plus les semis réguliers ont tendance à filtrer les erreurs d'intégration comme le montrent les tests suivants. Nous avons tracé la valeur de l'erreur  $\mathcal{L}^2$  à la solution 3 en fonction d'un indicateur  $\sigma$  de répartition des nœuds dans  $[0,1]$  :  $\sigma^2 = \frac{1}{n-1} \sum_{i=0}^{n-1} \left( \frac{1}{n-1} - (x_{i+1} - x_i) \right)^2$  (voir annexe E). Le semis régulier correspond à  $\sigma = 0$ , et  $\sigma$  croît avec l'irrégularité.

Les résultats<sup>2</sup> regroupés dans les figures E.2, E.3, E.4 et E.5 montrent que l'influence du semis de nœuds n'est pas la même sur les différentes méthodes de résolution. En particulier, les méthodes basées sur la dérivée diffuse sont plus sensibles à des semis irréguliers. Ceci est particulièrement vrai pour l'intégration avec tuiles.

L'intégration adaptée diminue fortement l'influence du semis de nœuds sur l'erreur. Les résultats sont spectaculaires pour l'intégration adaptée sur pavés de Voronoï (figure E.3 et E.4). On pouvait s'attendre à ce type de résultat pour les raisons suivantes : l'erreur augmente avec  $\sigma$  pour l'intégration de Gauss-Legendre car les dérivées des fonctions de forme sont très éloignées de polynômes dans ce cas. L'intégration adaptée prend en compte cet écart, et gomme ainsi la difficulté.

<sup>2</sup>. Nous n'avons testé que les méthodes présentant des propriétés de convergence « raisonnables ».

Les courbes montrent que l'intégration adaptée a tendance à filtrer certains termes comme le font les semis réguliers ( $\sigma = 0$ ), et assure ainsi une bonne convergence.

#### 2.4.4 Convergence pour le problème de Dirichlet

**Première analyse : les cas 1 et 3** Nous allons commencer par analyser les résultats obtenus pour les problèmes 1 et 3, car il sont simples et représentent les deux opérateurs que nous avons traités ( $\alpha = 0$  pour le cas 1, et  $\alpha = \pi^2$  pour le cas 3).

Ces cas mettent en évidence une classification des méthodes que nous justifierons dans la suite sur les autres tests. On distingue trois classes de méthodes

1. Les méthodes non convergentes;
2. Les méthodes semi-convergentes;
3. Les méthodes convergentes.

Les méthodes non convergentes sont celles qui utilisent la dérivée diffuse sans intégration adaptée (méthodes 5, 9 et 13 du tableau 2.2), comme le montre la figure E.6. On retrouve ici un résultat connu (voir [Lao96] par exemple).

Les méthodes s'appuyant sur la dérivée réelle et une intégration de Gauss-Legendre semblent converger, puis bloquent sur une erreur minimale qui n'est pas un zéro machine comme nous le verrons plus tard. Ces méthodes (1, 2, 3, 7, 11 dans le tableau 2.2) forment le groupe des méthodes semi-convergentes. On peut distinguer deux sous-groupes dans cette catégorie, formés des méthodes 7, 11 d'une part, et 1, 2 et 3 d'autre part comme le montre les figures E.7 et E.8. Les méthodes du premier sous-groupe ont un comportement semblable à celui des méthodes non convergentes, alors que le changement de pente est plus tardif et moins marqué pour celles du second sous-groupe. La méthode EFG ne présente pas ce changement de pente caractéristique, mais ses taux de convergence sont faibles ( $-1.49$  selon  $\mathcal{L}^2$  et  $-0.48$  selon  $\mathcal{H}^1$  pour le problème 1).

Enfin les autres méthodes, qui utilisent toutes l'intégration adaptée décrite précédemment sont convergentes comme le montre la figure E.9. Les courbes changent de pente lorsque le zéro numérique de la méthode est atteint (nous y reviendrons dans la partie suivante).

Les tableaux 2.3 et 2.4 montrent les taux de convergence expérimentaux pour les méthodes convergentes. Notons que le patch test assure au mieux un taux de convergence de 2 selon  $\mathcal{L}^2$  et 1 selon  $\mathcal{H}^1$ .

| méthodes | 4     | 6     | 8     | 10    | 12    | 14    | EFL   |
|----------|-------|-------|-------|-------|-------|-------|-------|
| $a$      | -2.05 | -2.15 | -2.02 | -2.04 | -2.05 | -2.03 | -2.02 |
| $b$      | -0.77 | -0.46 | -0.85 | -1.30 | -1.52 | -1.28 | -0.96 |
| $a'$     | -1.02 | -1.01 | -1.01 | -1.01 | -1.02 | -1.01 | -1.01 |
| $b'$     | -0.43 | -0.51 | -0.46 | -0.74 | -0.97 | -0.83 | -0.51 |

TAB. 2.3: Taux de convergence des méthodes convergentes pour le problème 1. On ajuste  $\log(e) = a * \log(n) + b$  aux résultats issus des différentes simulations par moindres carrés pour la norme  $\mathcal{L}^2$  et  $\log(e) = a' * \log(n) + b'$  pour la semi norme  $\mathcal{H}^1$ .



| méthodes | 4     | 6     | 8     | 10    | 12    | 14    | eff   |
|----------|-------|-------|-------|-------|-------|-------|-------|
| $a$      | -2.02 | -2.03 | -2.01 | -2.02 | -2.01 | -2.01 | -2.02 |
| $b$      | -0.22 | -0.12 | -0.24 | -0.51 | -0.83 | -0.71 | -0.34 |
| $a'$     | -1.00 | -1.00 | -1.00 | -1.00 | -1.00 | -1.00 | -1.00 |
| $b'$     | 0.38  | 0.30  | 0.39  | 0.08  | -0.18 | -0.19 | -0.32 |

TAB. 2.4: Taux de convergence des méthodes convergentes pour le problème 3.

**Le patch test** Les résultats du patch test sont regroupés sur les figures E.22 à E.25.

**Méthodes non convergentes** Le patch test montre une très faible convergence pour la norme  $\mathcal{L}^2$  uniquement. Ceci confirme la non-convergence pour les différents problèmes de Dirichlet que nous avons traités. On ne peut donc pas utiliser la pseudo-dérivation sans intégration adaptée (à l'exception des maillages réguliers).

**Méthodes semi-convergentes** Le patch test montre une convergence selon la norme  $\mathcal{L}^2$  avec des taux de  $-1$  environ. Les méthodes du premier groupe ont un taux inférieur à  $-1$ , alors que celles du second groupe ont un taux supérieur à  $-1$ . En revanche les dérivées ne convergent pas. Ceci justifie le terme « semi-convergent ». Nous classons la méthode EFG dans ce groupe bien qu'elle montre une convergence lente selon la semi norme  $\mathcal{H}^1$ . Elle passe en effet le patch test au sens faible, avec des taux de convergence de  $-1$  et  $-0.5$  respectivement.

**Méthodes convergentes** Toutes les méthodes que nous avons classées convergentes selon les cas 1 et 3, passent le patch test au sens fort. On remarque que l'erreur d'approximation de la solution linéaire augmente lentement avec le nombre de degrés de liberté. Nous pensons que cette augmentation est liée à l'accumulation des erreurs d'arrondi dans les procédures numériques, qui augmente avec le nombre d'opérations. Nous avons donc une détermination du zéro numérique pour une procédure de calcul donnée, qui est l'erreur sur le patch test avec le même semis de nœuds (le résidu varie beaucoup avec le semis de nœuds).

**Définition 20** *Le zéro numérique d'une méthode pour une norme donnée et un semis de nœuds donné est l'erreur à la solution du patch test.*

Dans nos expériences, les méthodes avec tuiles ont un zéro numérique plus élevé que les autres méthodes. Nous ne savons pas expliquer ceci pour l'instant.

Ainsi, le patch test confirme les impressions que nous avons eues dans l'analyse des cas 1 et 3. Il montre l'importance d'une intégration numérique performante, et par là le succès de l'intégration numérique adaptée pour les problèmes monodimensionnels.

**Autres problèmes (2, 4, 5, et 6)** On retrouve les mêmes résultats que précédemment sur les problèmes 2 et 4 comme le montrent les figures en annexes.

Les problèmes 5 et 6 montrent des comportements singuliers : les méthodes semi-convergentes du second groupe semblent convergentes sur ces problèmes. Le cas de la méthode EFG est très particulier.

Les taux de convergence observés pour les méthodes convergentes sont environ  $-2$  pour la norme  $\mathcal{L}^2$  et  $-1$  pour la semi-norme  $\mathcal{H}^1$ .

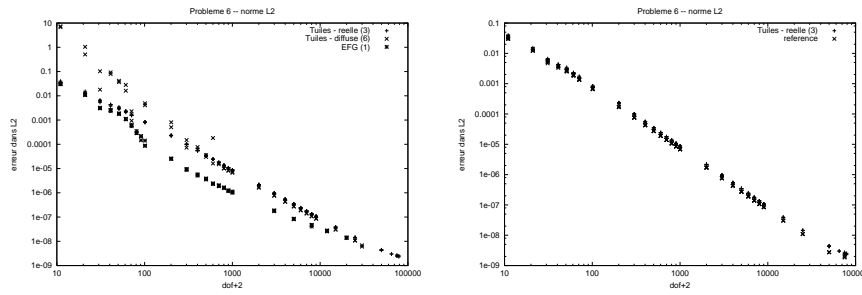


FIG. 2.12: Le problème de Dirichlet 6 fait apparaître des comportements particuliers (1 et 6) de certaines méthodes.

### 2.4.5 Convergence pour le calcul des modes propres

Le calcul des modes propres et des fréquences propres par les différentes méthodes sans maillages nous amène à des conclusions proches des précédentes.

Les méthodes avec dérivées diffuses et sans intégration adaptée ne convergent pas. Le groupe des méthodes semi-convergentes se sépare en deux sous-groupes : des méthodes qui ne convergent pas comme précédemment, et des méthodes convergentes comme c'était le cas sur les problèmes 5 et 6 (voir les méthodes 2 et 3 en particulier).

|       | 2     | 3     | 4     | 6     | 8     | 10    | 12    | 14    | ref.  |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| $a_1$ | -1.98 | -2.06 | -2.02 | -2.00 | -2.02 | -2.02 | -2.02 | -2.02 | -2.02 |
| $b_1$ | 0.03  | 0.15  | 0.07  | 0.19  | 0.12  | -1.24 | -1.13 | -0.40 | -0.03 |
| $a_2$ | -2.18 | -2.04 | -2.03 | -2.00 | -2.02 | -2.01 | -2.03 | -2.02 | -2.02 |
| $b_2$ | 1.05  | 0.71  | 0.70  | 0.79  | 0.72  | -0.68 | -0.50 | 0.21  | 0.58  |
| $a_3$ | -2.13 | -2.04 | -2.03 | -2.00 | -2.02 | -1.98 | -2.04 | -2.03 | -2.02 |
| $b_3$ | 1.33  | 1.07  | 1.06  | 1.13  | 1.07  | -0.39 | -0.11 | 0.57  | 0.93  |
| $a_4$ | -2.10 | -2.04 | -2.04 | -2.00 | -2.02 | -1.95 | -2.06 | -2.03 | -2.02 |
| $b_4$ | 1.51  | 1.32  | 1.32  | 1.39  | 1.32  | -0.23 | 0.18  | 0.83  | 1.18  |
| $a_5$ | -2.06 | -2.03 | -2.03 | -2.01 | -2.02 | -1.84 | -2.08 | -2.04 | -2.02 |
| $b_5$ | 1.61  | 1.50  | 1.50  | 1.61  | 1.51  | -0.34 | 0.43  | 1.05  | 1.38  |

TAB. 2.5: Taux de convergence pour le problème de valeurs propres ( $\log(e) = a_i \log(n) + b_i$  pour  $(i\pi)^2$ ,  $i$  dans  $[1, 5]$ ).

On trouve encore des taux de convergence très voisins, mais les ordonnées à l'origine montre jusqu'à un ordre de grandeur d'écart entre les méthodes sur la valeur de l'erreur (par exemple entre la méthode 10 et les éléments finis linéaires). Bien que les éléments finis linéaires donnent une moins bonne estimation des valeurs propres, ils calculent mieux le vecteur propre associé.

Pour finir, nous avons remarqué que les méthodes 7 et 11<sup>3</sup> sont convergentes lorsqu'on intègre avec un unique point de Legendre par pavé de Voronoï comme le montre la figure 2.13. Ceci peut ouvrir une piste d'investigation sur la meilleure façon d'intégrer la dérivée réelle.

3. pavage Voronoï, dérivée réelle et intégration de Gauss-Legendre

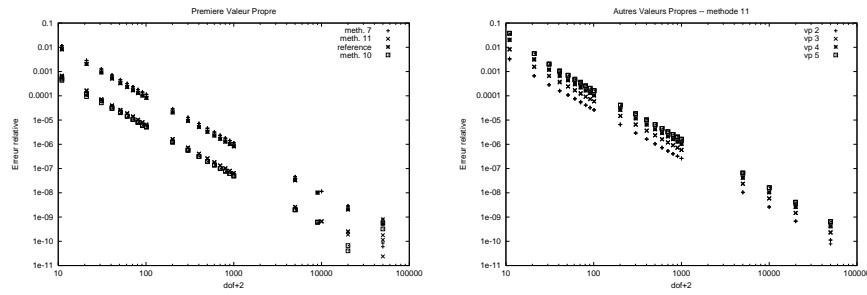


FIG. 2.13: Convergence des méthodes 7 et 11 avec un unique point d'intégration par pavé de Voronoï.

## 2.5 Conclusion

Nous avons testé quatorze méthodes sans maillage s'appuyant sur les moindres carrés glissants. Ces tests montrent que ces méthodes se séparent en trois groupes

- (i) Méthodes non-convergentes
- (ii) Méthodes semi-convergentes
- (iii) Méthodes convergentes

Toutes les méthodes convergentes utilisent l'intégration numérique spéciale que nous avons développée. Les méthodes semi-convergentes peuvent peut-être mieux se comporter si l'on sait mettre au point une bonne stratégie d'intégration comme le montre la figure 2.13.

L'importance de la méthode d'intégration pour la construction d'une méthode de simulation est ainsi démontrée. L'intégration numérique spéciale répond bien à ce besoin, et son effet sur la sensibilité au semis de nœuds lui ouvre aussi d'autres applications.

Bien sûr, certaines difficultés subsistent avec la méthode proposée :

- surcoût dû au calcul de l'intégration,
- génération et maintien du pavage d'intégration en grande déformation.

Cependant, la méthode est très générale et peut prendre en charge tout type d'approximation (*enriched basis* par exemple [FCMB97]) dès que l'on sait écrire les contraintes issues du patch test. Nous montrons aussi que les conditions nécessaires du théorème 13 sont expérimentalement suffisantes. Enfin, le patch test trouve encore ici une place centrale pour la construction d'une méthode numérique efficace pour la formulation variationnelle et confirme l'intuition de Irons. Nous verrons de la même façon son rôle pour la prédiction de taux de convergence.

Dans le chapitre suivant nous allons étendre l'intégration numérique adaptée pour l'appliquer à des structures élancées avec hypothèse de Kirchhoff (équation d'ordre 4).

## Chapitre 3

# Poutre diffuse

Nous allons développer un élément poutre Kirchhoff diffus pour résoudre un problème de forage (chapitre suivant). L'objectif est d'obtenir une meilleure description du contact entre la structure et la roche grâce à l'approximation diffuse tout en conservant une description nodale du contact.

Les deux principales difficultés dans le traitement des corps élancés (poutre, plaques, coques) sont le phénomène de blocage et l'ordre des dérivées mises en jeu avec certaines hypothèses complémentaires sur la physique de ces corps (Kirchhoff par exemple). Les méthodes sans maillage ont été utilisées pour résoudre ces deux classes de problème:

- Krysl et Belytschko pour les plaques et les coques avec EFG [KB95, KB96],
- Donning et Liu pour les poutres droites et les plaques avec des splines [DL98], Li *et. al.* pour les coques [LHL00].
- Atluri *et. al.* pour les poutres droites avec l'approximation diffuse Hermite [ACK99].

Dans chacun de ces exemples, la régularité de l'approximation est mise à profit. Les auteurs règlent le problème de blocage de façon différente: Krysl joue sur le support des fonctions de forme et Donning utilise des propriétés des splines. Cependant, nous avons remarqué que l'intégration numérique de EFG n'est pas satisfaisante et les splines utilisées s'appuient sur des semis de nœuds réguliers.

Nous proposons de remédier à ces deux problèmes en étendant la construction de l'intégration adaptée à la **flexion** des poutres et en utilisant l'hypothèse dite des « sections droites ». La méthode s'applique aussi bien à une approximation diffuse Lagrangienne qu'à l'approximation diffuse Hermite que nous avons définie. Ceci nous permet finalement de définir un modèle de poutre qui passe le patch test dans le cas des barres.

Le problème de poutre en trois dimensions est d'abord décrit avec son cadre naturel qui est la géométrie différentielle des courbes. La formulation variationnelle est donnée dans le repère de Frenet, ainsi que sa forme avec l'hypothèse supplémentaire de Kirchhoff. La seconde partie décrit le patch test et l'approximation utilisée. L'intégration numérique est construite en suivant le schéma de l'équation de la chaleur. Comme la flexion correspond à une équation d'ordre 4, nous proposons une extension de la méthode aux équations d'ordre  $2m$ , puis à des interpolations de type Hermite. Enfin, des résultats numériques sont présentés.

### 3.1 Description du problème

#### 3.1.1 Rappel de géométrie différentielle des courbes

Rappelons les bases de l'analyse différentielle des courbes gauches (dans  $\mathbb{R}^3$ ) qui seront utiles pour décrire l'équilibre des poutres. On considère une courbe paramétrique  $\mathcal{C}^3$ ,  $\vec{x}(u) = \langle x(u), y(u), z(u) \rangle^\top$ .

**Généralités** Nous utiliserons les propriétés et définitions suivantes :

1. La longueur du segment de courbe défini par l'intervalle  $[u_0, u_1]$  est

$$l(u_0, u_1) = \int_{u_0}^{u_1} \sqrt{(x'(u))^2 + (y'(u))^2 + (z'(u))^2} du.$$

L'abscisse curviligne sur cette courbe, en fixant l'origine au point 0 par convention est  $s(u) = l(0, u)$ . La fonction  $s(u)$  est une bijection et  $s$  est le paramétrage canonique de la courbe. On note  $'$  la dérivation par rapport à  $s$  lorsqu'il n'y a pas d'ambiguïté.

2. La tangente au point  $\vec{x}(s)$  est

$$\vec{t}(s) = \vec{x}'(s) = \frac{d\vec{x}}{ds} = \frac{1}{\sqrt{(x'(u))^2 + (y'(u))^2 + (z'(u))^2}} \frac{d\vec{x}}{du}.$$

Elle est normée, on a  $\frac{d\vec{t}}{ds} \cdot \vec{t} = 0$  pour tout  $s$  donc  $\vec{t}'$  et  $\vec{t}$  sont orthogonaux pour tout  $s$ .

3. La courbure  $\rho(s)$  est la norme de  $\vec{t}'$  et le vecteur  $\vec{n}$  est défini par

$$\vec{n}(s) = \frac{\vec{t}'(s)}{\rho(s)} \text{ si } \rho(s) \neq 0$$

4. Le repère de Frenet au point  $\vec{x}(s)$  est  $(\vec{x}(s), \vec{t}(s), \vec{n}(s), \vec{b}(s))$  où  $\vec{b} = \vec{t} \wedge \vec{n}$  est le vecteur binormal.

5. On démontre alors que  $\vec{b}'$  est colinéaire à  $\vec{n}$  et la torsion est définie par

$$\vec{b}'(s) = -\tau \vec{n}(s)$$

6. La base de Frenet vérifie le système d'équations différentielles

$$\begin{pmatrix} \vec{t}' \\ \vec{n}' \\ \vec{b}' \end{pmatrix} = \begin{bmatrix} 0 & \rho & 0 \\ -\rho & 0 & \tau \\ 0 & -\tau & 0 \end{bmatrix} \begin{pmatrix} \vec{t} \\ \vec{n} \\ \vec{b} \end{pmatrix}$$

Nous noterons  $C^\top(s)$  la matrice antisymétrique des courbures de ce système.

**Grandeur vectorielle sur une courbe gauche** Soit une grandeur vectorielle  $\vec{u}$  connue sur une courbe gauche. Il est naturel de projeter  $\vec{u}$  sur la base de Frenet, soit  $\vec{u}(s) = u_t(s)\vec{t}(s) + u_n(s)\vec{n}(s) + u_b(s)\vec{b}(s)$ ;  $U(s) = \langle u_t(s), u_n(s), u_b(s) \rangle^\top$  est le vecteur des coordonnées de  $\vec{u}$  dans le repère de Frenet.

**Propriété 36** Les coordonnées  $V(s)$  du vecteur  $\vec{u}'$  sont alors  $V(s) = U'(s) + C(s)U(s)$  et en posant

$$D(s) = \begin{bmatrix} -\rho^2 & -\rho' & \rho\tau \\ \rho' & -(\rho^2 + \tau^2) & -\tau' \\ \rho\tau & \tau' & -\tau^2 \end{bmatrix}$$

le vecteur des coordonnées de  $\vec{u}''$  est  $U''(s) + 2C(s)U'(s) + D(s)U(s)$ .

Ces formules nous seront très utiles tout le long de ce chapitre.

### 3.1.2 Équilibre d'une poutre

La cinématique d'une poutre de longueur  $l$  est décrite par le déplacement de sa fibre neutre  $\vec{u}$  et la rotation de la section droite  $\vec{\omega}$  (figure 3.1). Ces vecteurs sont projetés sur le repère de Frenet de la fibre neutre de l'état de référence.

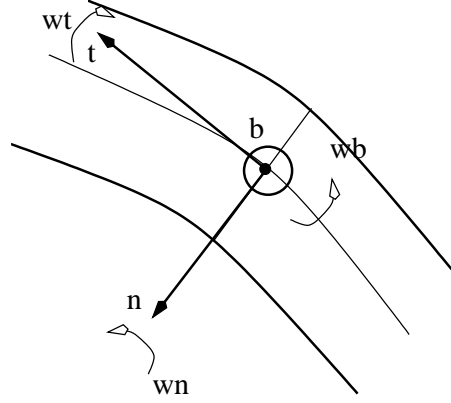


FIG. 3.1: Cinématique d'une poutre.

L'équilibre de la structure est décrite par les relations suivantes sur les contraintes généralisées  $\vec{T}, \vec{M}$  (éléments de réduction des efforts sur une section droite)

$$\begin{cases} \vec{T}'(s) + \vec{f} + \vec{F}_i \delta_i = \vec{0} \\ \vec{M}'(s) + \vec{t}(s) \wedge \vec{T}(s) + \vec{C}_j \delta_j = \vec{0} \end{cases}, s \in [0, l]$$

où  $\vec{f}$  représente une force répartie exercée sur la structure par unité de longueur (charge linéique). Le chargement est complété par des forces ponctuelles  $\vec{F}_i$  et des couples ponctuels  $\vec{C}_j$ .

Les déformations généralisées de la structure sont définies par

$$\begin{cases} \vec{\epsilon}_T(s) = \vec{u}'(s) + \vec{t}(s) \wedge \vec{\omega}(s) \\ \vec{\epsilon}_M(s) = \vec{\omega}'(s) \end{cases}$$

L'élasticité linéaire lie les contraintes et les déformations généralisées par les relations suivantes :

$$\vec{T} = \begin{Bmatrix} ES \vec{\epsilon}_T \cdot \vec{t} \\ k_n GS \vec{\epsilon}_T \cdot \vec{n} \\ k_b GS \vec{\epsilon}_T \cdot \vec{b} \end{Bmatrix}, \vec{M} = \begin{Bmatrix} k_0 I_0 GS \vec{\epsilon}_M \cdot \vec{t} \\ EI_n \vec{\epsilon}_M \cdot \vec{n} \\ EI_b \vec{\epsilon}_M \cdot \vec{b} \end{Bmatrix} \quad (3.1)$$

où

- $E$  et  $G$  sont le module d'Young et le module de cisaillement ( $E$  et  $G$  sont liés par  $G = \frac{E}{2(1+\nu)}$  avec  $\nu$  le coefficient de Poisson),
- $S, k_0, k_n, k_b, I_0, I_n, I_b$  dépendent de la géométrie de la section de la poutre.

Dans la suite un matériau est défini par ses caractéristiques physiques, mais aussi par les caractéristiques géométriques de la section droite. Pour plus de détails sur ce modèle, on pourra consulter [AC69].

Dans les relations (3.1) nous avons négligé les termes croisés (l'influence d'un moment de torsion  $\vec{M} \cdot \vec{t}$  sur la variation relative de longueur  $\vec{\epsilon}_T \cdot \vec{t}$  par exemple). Ainsi, dans le repère de Frenet la matrice de Hooke généralisée est diagonale. Cependant, la méthode numérique proposée est parfaitement applicable au cas général d'une matrice de Hooke symétrique définie positive.

### 3.1.3 Formulation variationnelle

Les équations d'équilibre associées à l'élasticité linéaire permettent d'écrire la formulation variationnelle de l'équilibre d'une poutre  $[0, l]$ . Les grandeurs sont calculées à partir d'une géométrie de référence à laquelle correspondent d'éventuelles contraintes initiales.

**Théorème 14** *L'équilibre d'une poutre  $[0, l]$  soumise à*

- une force linéique  $\vec{f}$ ,
- des forces et couples ponctuels  $\vec{F}_i, \vec{C}_j$ ,
- des contraintes initiales  $\vec{T}_0, \vec{M}_0$ ,

est décrit par les vecteurs  $\vec{u}, \vec{\omega}$  vérifiant pour tout déplacement virtuel  $\vec{u}^*, \vec{\omega}^*$

$$\int_0^l (\vec{T} \cdot \vec{\epsilon}_T^* + \vec{M} \cdot \vec{\epsilon}_M^*) ds = \int_0^l \vec{f} \cdot \vec{u}^* ds + \sum_i \vec{F}_i \cdot \vec{u}_i^* + \sum_j \vec{C}_j \cdot \vec{\omega}_j^* - \int_0^l (\vec{T}_0 \cdot \vec{\epsilon}_T^* + \vec{M}_0 \cdot \vec{\epsilon}_M^*) \quad (3.2)$$

### 3.1.4 Hypothèse de Kirchhoff

L'hypothèse de Kirchhoff signifie que les sections droites restent droites. Pour le modèle que nous utilisons, cela s'écrit  $\vec{T} \cdot \vec{n} = 0$  et  $\vec{T} \cdot \vec{b} = 0$ . Par définition de  $\vec{T}$ , le couplage entre les vecteurs  $\vec{u}$  et  $\vec{\omega}$  s'écrit alors explicitement

$$\begin{cases} \omega_n = -\vec{u}' \cdot \vec{b} \\ \omega_b = \vec{u}' \cdot \vec{n} \end{cases}$$

Tous calculs faits (en utilisant la propriété 36), cela permet d'écrire les coordonnées de  $\vec{\omega}$ .

**Propriété 37** *Sous les hypothèses de Kirchhoff, les coordonnées des vecteurs  $\vec{\omega}$  et  $\vec{\omega}'$  dans le repère de Frenet sont :*

$$\vec{\omega} = \begin{cases} \omega_t \\ -u'_b - \tau u_n \\ u'_n + \rho u_t - \tau u_b \end{cases} \quad \vec{\omega}' = \begin{cases} \omega'_t + \rho \tau u_n + \rho u'_b \\ \rho \omega_t - \rho \tau u_t - 2\tau u'_n - \tau' u_n + \tau^2 u_b - u''_b \\ \rho u'_t + \rho' u_t + u''_n - \tau^2 u_n - 2\tau u'_b - \tau' u_b \end{cases} \quad (3.3)$$

L'énergie de déformation de la poutre ne compte plus que quatre termes :

- $ES(\vec{u} \cdot \vec{t})^2$  pour la traction,
- $k_0 I_0 GS(\vec{\omega}' \cdot \vec{t})^2$  pour la torsion,
- $EI_n(\vec{\omega}' \cdot \vec{n})^2$  et  $EI_b(\vec{\omega}' \cdot \vec{b})^2$  pour la flexion.

et la cinématique est complètement décrite par les fonctions  $u_t, \omega_t, u_n, u_b$ . La forme bilinéaire qui découle du premier membre de l'équation 3.2 s'écrit

$$\mathcal{E}(\epsilon, \epsilon^*) = \int_0^l E^\top(s) H E(s) ds.$$

avec

- $E(s) = \langle \vec{u}' \cdot \vec{t}, \vec{\omega}' \cdot \vec{t}, \vec{\omega}' \cdot \vec{n}, \vec{\omega}' \cdot \vec{b} \rangle^\top$
- $H = \text{diag}(ES, k_0 I_0 GS, EI_n, EI_b)$ .

## 3.2 Patch test, approximation

### 3.2.1 Le patch test

L'intégration numérique adaptée aux méthodes sans maillage s'appuie sur le patch test. Nous utilisons ici la définition première du patch test : les états de contraintes constantes sont reproduits exactement à la précision numérique près. Avec cette définition, nous sommes contraints de distinguer deux cas de figures :

1. les poutres rectilignes ou barres ( $\rho = \tau = 0$ ),
2. les poutres courbes ( $\rho \neq 0$ ).

En effet, l'état de contraintes constantes correspond aux conditions  $\vec{T}' = 0$ ,  $\vec{M}' = 0$  tout le long de la poutre. Et pour une poutre de section annulaire uniforme (les paramètres géométriques sont constants le long de la poutre, il vient d'après la seconde équation d'équilibre  $\vec{t} \wedge \vec{T} = \vec{0}$ . Soit  $\vec{T} = T_t \vec{t}$  et comme  $\vec{T}' = T_t' \vec{t} + \rho T_t \vec{n} = \vec{0}$ , la coordonnée  $T_t$  est définie par

$$\begin{cases} T_t = \text{constante} \\ \rho T_t = 0 \end{cases}$$

Si  $\rho$  est non nul, alors  $\vec{T} = \vec{0}$  et l'hypothèse de Kirchhoff est automatiquement vérifiée.

**Propriété 38** *Une poutre courbe ( $\rho \neq 0$ ) sous contraintes constantes respecte l'hypothèse de Kirchhoff.*

L'analyse du patch test pour les poutres courbes peut être menée dans le cas général (sans hypothèse de Kirchhoff). Pour passer le patch test il est nécessaire que :

- l'espace d'approximation contienne les solutions du patch test,
- l'intégration respecte certaines conditions (voir le chapitre précédent).

Nous commençons par étudier la première condition.

**Cas d'une barre** Dans le cas d'une barre, les quatre fonctions  $u_t, \omega_t, u_n, u_b$  sont découplées et sont solutions d'équations différentielles du type  $u^{(2m)} + f = 0$  avec  $m = 1$  pour  $u_t, \omega_t$  et  $m = 2$  pour  $u_n, u_b$ . L'état de contraintes constantes correspond à une déformation polynomiale de degré  $m$ . L'approximation des fonctions  $u_t$  et  $\omega_t$  doit donc être consistante d'ordre 1 et celle des fonctions  $u_n$  et  $u_b$  consistante d'ordre 2.

**Cas d'une poutre courbe** On ne sait que rarement calculer la cinématique associée à un état de contrainte constante dans le cas d'une poutre courbe. Les contraintes sont définies par  $\vec{M}$  constant et  $\vec{T}$  nul. Pour une section de symétrie circulaire (disque, anneau), l'élasticité linéaire implique

$$\vec{\omega}' = \frac{1}{EI} (\vec{M} + \nu (\vec{M} \cdot \vec{t}) \vec{t})$$

et la rotation de la section droite est

$$\vec{\omega} = \frac{1}{EI} (sI + \nu a(s)) \vec{M} + \vec{\omega}_0$$

avec le tenseur  $a(s)$  défini par (on note  $\vec{v} \otimes \vec{v} = [v_i v_j]_{i,j}$ )

$$a(s) = \int_0^s \vec{t}(\alpha) \otimes \vec{t}(\alpha) d\alpha$$



Comme  $\vec{T} = \vec{0}$ , le déplacement de la fibre neutre est calculé à partir de l'équation

$$\vec{u}' = \vec{\omega} \wedge \vec{t}$$

Nous avons explicité les formules dans le cas d'une poutre circulaire dans le cas général et pour une poutre hélicoïdale avec coefficient de Poisson  $\nu$  nul (voir annexe G). L'espace d'approximation doit contenir les combinaisons linéaires des fonctions :

$$1, s, s^2, \sin(\rho s), \cos(\rho s), s \sin(\rho s), s \cos(\rho s)$$

pour une poutre circulaire (rayon  $r = \frac{1}{\rho}$ ) et

$$1, s, s^2, \sin(\gamma s), \cos(\gamma s), s \sin(\gamma s), s \cos(\gamma s), s^2 \sin(\gamma s), s^2 \cos(\gamma s)$$

pour une poutre hélicoïdale (courbure  $\rho$ , torsion  $\tau$  et  $\gamma = \sqrt{\rho^2 + \tau^2}$ ).

Il est possible de construire une approximation diffuse qui soit consistante pour les fonctions que nous venons de citer comme le montre l'annexe H. Cependant nous ne savons pas encore écrire les contraintes sur l'intégration dans le cas de poutre courbe. Nous allons donc procéder comme dans le cas de la méthode des éléments finis où l'on se contente de passer le patch test au sens faible pour les poutres courbes.

### 3.2.2 Approximation utilisée

**Type d'approximation** Les fonctions  $u_t$  et  $\omega_t$  peuvent admettre des dérivées discontinues (voir la figure 1.4, page 148). De la même façon, les fonctions  $u_n$  et  $u_b$  peuvent avoir des dérivées secondes discontinues (cas d'une force ponctuelle par exemple). Nous avons donc choisi d'approximer les fonctions  $u_t$  et  $\omega_t$  par une interpolation diffuse Lagrangienne et  $u_n$  et  $u_b$  par une interpolation diffuse Hermitienne. L'interpolation Hermitienne simplifie la prise en compte des couples ponctuels.

**Consistance** Le degré de consistance minimale pour les fonctions  $u_t$  et  $\omega_t$  est 1, cependant on doit approcher des fonctions trigonométriques. Nous utiliserons une approximation diffuse Lagrangienne quadratique (c'est-à-dire consistante pour  $1, s, s^2$ ). De la même façon, on utilisera une consistance cubique pour l'approximation Hermite.

**Dérivation** On peut utiliser la dérivation usuelle ou la dérivée au sens diffus pour l'approximation Lagrangienne. Au contraire, il est nécessaire d'utiliser la dérivation complète pour l'approximation Hermite pour assurer la stabilité de l'intégration numérique comme nous le verrons plus loin.

### 3.2.3 Modèle de poutre discret

Nous utilisons des nœuds  $s_i$  dans  $[0, l]$  avec  $s_0 = 0$  et  $s_n = l$ . Les fonctions d'influence sont les fonctions « de Voronoï » d'ordre 2 (annexe B). Une fonction de forme diffuse Lagrangienne  $L_i$  et deux fonctions de forme Hermitiennes  $N_i, M_i$  sont attachées à chaque nœud  $s_i$  pour approcher la cinétique de la poutre par

$$\begin{cases} u_t(s) &= \sum_{i \in I(s)} L_i(s) U_t^i \\ \omega_t(s) &= \sum_{i \in I(s)} L_i(s) \Omega_t^i \\ u_n(s) &= \sum_{i \in I(s)} N_i(s) U_n^i + M_i(s) V_n^i \\ u_b(s) &= \sum_{i \in I(s)} N_i(s) U_b^i + M_i(s) V_b^i \end{cases}$$

Chaque nœud porte un vecteur de six degrés de liberté

$$U^i = \langle U_t^i, \Omega_t^i, U_n^i, V_n^i, U_b^i, V_b^i \rangle^\top$$

et le vecteur  $E(s)$  s'écrit

$$\left\{ \begin{array}{l} E(s) = \sum_{i \in I(s)} B_i(s) U^i \\ \text{avec} \\ B_i^\top(s) = \begin{bmatrix} L'_i & 0 & -\rho\tau L_i & \rho L'_i \\ 0 & L'_i & \rho L_i & \rho' L_i \\ -\rho N_i & \rho\tau N_i & -2\tau N'_i - \tau' N_i & N''_i - \tau^2 N_i \\ -\rho M_i & \rho\tau M_i & -2\tau M'_i - \tau' M_i & M''_i - \tau^2 M_i \\ 0 & \rho N'_i & \tau^2 N_i - N''_i & -\tau' N_i \\ 0 & \rho M'_i & \tau^2 M_i - M''_i & -\tau' M_i \end{bmatrix} \end{array} \right.$$

On utilise ce modèle discret dans la formulation variationnelle pour obtenir un système linéaire  $KU = F$ . La contribution du couple  $i, j$  à la matrice de raideur est

$$K^{ij} = \int_0^l B_i^\top(s) H B_j(s) ds.$$

**Gestion des matériaux** En pratique, il est courant d'avoir une poutre multimatériaux. Or nous savons que la solution peut être discontinue à la soudure de deux matériaux. Pour bien capter cette discontinuité, nous considérons deux poutres, c'est-à-dire que leur seul point commun est le nœud d'interface (voir figure 3.2). Ceci revient simplement à modifier les fonctions d'influence des nœuds proches de la frontière. Le cas de force ponctuelle est traité de la même manière, car on n'utilise pas systématiquement la dérivée complète de l'approximation Lagrangienne. Une alternative pour ce cas est l'utilisation des fonctions d'influences modifiées de Krongauz (voir l'étude bibliographique, page 148).

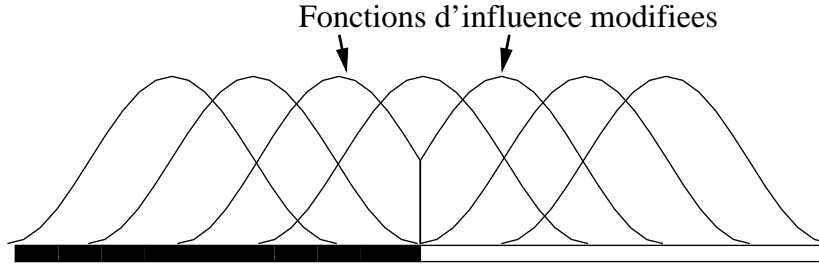


FIG. 3.2: Fonctions d'influence de deux poutres.

**Discontinuités géométriques** Le modèle géométrique peut faire apparaître des discontinuités artificielles du repère de Frenet, de la courbure ou de la torsion (comme dans le chapitre suivant par exemple). Dans ce cas, les valeurs nodales  $U^j$  du nœud commun  $s_j$  n'assurent pas la continuité de la cinématique  $\vec{u}, \vec{\omega}$ . Nous devons donc dédoubler les nœuds en  $s_j^-$  et  $s_j^+$  et souder les deux poutres en écrivant **vectériellement dans le repère Cartésien de référence**  $\vec{u}^- = \vec{u}^+$  et  $\vec{\omega}^- = \vec{\omega}^+$ . Ces contraintes linéaires sont prises en compte par multiplicateurs de Lagrange et s'écrivent  $SU = 0$ . Cependant, la matrice de raideur  $K$  prend une structure bloc

$$\begin{bmatrix} K_1 & 0 \\ 0 & K_2 \end{bmatrix}$$

et chacune des matrices  $K_1$ ,  $K_2$  ont un défaut de rang de 6. Ainsi celui de la matrice de raideur globale est 12.

En pratique, nous bloquons les mouvements de corps solide par modification de la matrice de raideur en annulant six degrés de liberté indépendants. La matrice de raideur ainsi modifiée est alors inversible, ce qui rend possible la résolution du système avec multiplicateurs de Lagrange en conservant un stockage *skyline*. Dans le cas d'une soudure, la matrice  $K$  modifiée présente toujours un défaut de rang de 6. Pour obtenir une matrice inversible, il suffit d'utiliser la matrice modifiée  $K_m = K + S^T S$ , qui ne change pas la solution du système global, ni la largeur de bande de  $K$  (car  $S$  n'agit que sur les nœuds consécutifs  $s_j^-$  et  $s_j^+$  et on impose  $SU = 0$ ). Nous appelons le terme  $S^T S$  « raideur élémentaire de soudure » (cette méthode est inspirée de [Tij78]).

### 3.3 Intégration numérique

#### 3.3.1 Patch test pour une équation d'ordre $2m$

Nous avons vu page 160 que le patch test peut s'écrire (nous sommes dans le cadre d'application du théorème 12)

$$\forall p \in P, \forall v \in V_n, a^*(p, v) = a(p, v)$$

Pour l'équation

$$(-1)^m u^{(2m)}(x) = f(x), \forall x \in ]a, b[$$

la forme bilinéaire  $a$  est

$$a(u, v) = \int_a^b u^{(m)} v^{(m)} dx,$$

l'espace  $P$  est l'ensemble des polynômes de degré  $m$ ,  $\mathcal{P}_m$ . Si l'on considère l'ensemble  $\mathcal{P}_d$ ,  $d \geq m$ , alors les taux de convergence seront supérieurs (voir [ZT97]), on se restreint à  $d < 2m$ .

Soit  $N_i$  une base de l'espace  $V_n$  et la base canonique de  $\mathcal{P}_d$ ,  $(\frac{x^k}{k!})_{k \in [0, d]}$ . Les conditions pour  $k < m$  sont évidemment vérifiées par une méthode de type Gauss car  $(\frac{x^k}{k!})^{(m)} = 0$  dans ce cas. Les conditions restantes sont donc

$$\forall i, \forall k \in [m, d], a^*(\frac{x^k}{k!}, N_i) = a(\frac{x^k}{k!}, N_i).$$

Par ailleurs, l'identité suivante se démontre facilement avec  $k - m$  intégrations par parties successives

$$\forall f, \forall k \in [m, 2m[, a(\frac{x^k}{k!}, f) = \sum_{l=0}^{k-m} \left[ \frac{x^l}{l!} f^{(m-l-1)}(x) \right]_a^b$$

Ces considérations démontrent le théorème suivant.

**Théorème 15** *Soit  $m$  et  $d$  deux entiers tels que  $d \in [m, 2m[$ . Deux conditions nécessaires pour passer le patch test d'ordre  $d$  pour une équation d'ordre  $2m$  sont*

1. La matrice  $K^*$ ,  $K_{ij}^* = a^*(N_i, N_j)$  est de rang  $n - m$ .

2. Pour toute fonction de base  $N_i$ , l'intégration numérique  $\int$  et la dérivation  $\frac{\delta^k}{\delta x^k}$  sont liées par

$$\forall k \in [m, d], \int \frac{x^{k-m}}{(k-m)!} \frac{\delta^m N_i}{\delta x^m} dx = \left[ \sum_{l=0}^{k-m} \frac{x^l}{l!} \frac{\delta^{m-l-1} N_l}{\delta x^{m-l-1}}(x) \right]_a^b$$

On pose dans la suite

$$M_i^k(x) = \sum_{l=0}^{k-m} \frac{x^l}{l!} \frac{\delta^{m-l-1} N_i}{\delta x^{m-l-1}}(x)$$

### 3.3.2 Analyse des contraintes pour un schéma de Gauss modifié

Le théorème 15 est la généralisation des conditions connues pour l'équation de la chaleur ou l'élasticité linéaire [Lao96, Kro96]. Nous utilisons la technique décrite au chapitre précédent pour construire un schéma d'intégration de Gauss modifié avec points de Legendre  $\xi_g$  et poids  $\omega_g$ . Chaque pavé  $] \alpha, \beta [$  a une liste d'influence constante renumérotée  $1, \dots, n_i$  et les contraintes peuvent s'écrire  $e^\top \omega = 2$  et  $D\omega = 0$  comme précédemment, avec

$$k \in [0, d-m], i \in [1, n_i], g \in [1, n_g], j = (d-m)(i-1) + k + 1$$

$$D_{gj} = \frac{\xi_g^k}{k!} \frac{\delta^m N_i}{\delta \xi^m}(\xi_g) - \frac{1}{2} [M_i^k(\xi)]_{-1}^1$$

**Propriété 39** Les contraintes sur les poids  $\omega$  ne forment pas une famille libre. Elles assurent l'intégration exacte des polynômes jusqu'au degré  $2(d-m)$ .

Soit un polynôme de degré inférieur ou égal à  $d$ , et  $a_i$  tels que  $\sum_{i=1}^{n_i} a_i N_i(x) = p(x)$ , il vient en faisant la combinaison linéaire des contraintes avec les  $a_i$  pour un degré  $k \in [0, d-m]$  fixé :

– d'une part (membre de gauche)

$$\begin{aligned} \sum_{i=1}^{n_i} a_i \sum_{g=1}^{n_g} \frac{\xi_g^k}{k!} \frac{\delta^m N_i}{\delta \xi^m}(\xi_g) \omega_g &= \sum_{g=1}^{n_g} \frac{\xi_g^k}{k!} \sum_{i=1}^{n_i} a_i \frac{\delta^m N_i}{\delta \xi^m}(\xi_g) \omega_g \\ &= \sum_{g=1}^{n_g} \frac{\xi_g^k}{k!} \frac{\delta^m p}{\delta \xi}(\xi_g) \omega_g \\ &= \sum_{g=1}^{n_g} \frac{\xi_g^k}{k!} p^{(m)}(\xi_g) \omega_g \end{aligned}$$

– d'autre part (membre de droite),

$$\begin{aligned} \sum_{i=1}^{n_i} a_i M_i^k(\xi) &= \sum_{i=1}^{n_i} a_i \sum_{l=0}^k \frac{\xi^l}{l!} \frac{\delta^{m-l-1} N_i}{\delta \xi^{m-l-1}}(\xi) \\ &= \sum_{l=0}^k \frac{\xi^l}{l!} \sum_{i=1}^{n_i} a_i \frac{\delta^{m-l-1} N_i}{\delta \xi^{m-l-1}}(\xi) \\ &= \sum_{l=0}^k \frac{\xi^l}{l!} \frac{\delta^{m-l-1} p}{\delta \xi^{m-l-1}}(\xi) \\ &= \sum_{l=0}^k \frac{\xi^l}{l!} p^{(m-l-1)}(\xi) \end{aligned}$$

Et finalement, pour chaque degré  $k \in [0, d - m]$ , les conditions de consistance impliquent que

$$\sum_{g=1}^{n_g} \frac{\xi_g^k}{k!} p^{(m)}(\xi_g) \omega_g = \int_{-1}^1 \frac{\xi^k}{k!} p^{(m)}(\xi) d\xi$$

Ceci démontre que les polynômes jusqu'au degré  $2(d - m)$  sont intégrés exactement (car le degré de  $\xi p^{(m)}(\xi)$  varie de 0 à  $2(d - m)$ ). Par ailleurs, en prenant des polynômes de degré inférieur à  $m$ , la condition est  $0 = 0$  : il existe donc des contraintes redondantes dans  $D$ .

### 3.3.3 Test pour la flexion

La flexion d'une barre sous hypothèses de Kirchhoff est régie par une équation du quatrième ordre, soit  $u^{(4)} + f = 0$ . Nous considérons ici deux cas de chargement d'une poutre unitaire  $[0, 1]$ :

- origine bloquée ( $u(0) = 0, u'(0) = 0$ ) et poids (solution polynomiale de degré 4).
- extrémités fixée ( $u(0) = 0, u(1) = 0$ ) et force ponctuelle en  $x_0$ .

La solution numérique est construite avec l'approximation diffuse Lagrangienne avec consistance cubique ou quadratique. L'intégration associée est calculée avec des patch tests respectivement quadratique et cubique. La première méthode correspond au patch test minimum. Les taux de convergence observés pour les deux méthodes sont donnés dans le tableau 3.1. Les taux observés sont ceux attendus, mais nous avons remarqué que la technique d'intégration issue du patch test cubique n'est pas stable numériquement avec un nombre raisonnable de points de Gauss.

|       | A. D. quad. |            | A. D. cub. |            |
|-------|-------------|------------|------------|------------|
|       | problème 1  | problème 2 | problème 1 | problème 2 |
| $u$   | -1.83       | -1.91      | -3.98      | -3.00      |
| $u'$  | -2.04       | -1.94      | -3.39      | -3.58      |
| $u''$ | -1.39       | -1.47      | -2.08      | -1.56      |

TAB. 3.1: Taux de convergence pour la flexion avec l'approximation diffuse Lagrangienne (dérivée diffuse). Pour l'intégration, nous utilisons 3 points de Legendre supplémentaires et  $m = n_g + 1$ .

### 3.3.4 Mise en œuvre pour les poutres

Dans le cas général, nous nous contentons de vérifier les contraintes issues des patch tests minimaux : c'est-à-dire de degré 1 pour  $u_t$  et  $\omega_t$  et degré 2 pour  $u_n$  et  $u_b$ . Ainsi sur chaque pavé d'intégration  $]\alpha, \beta[$ , nous imposons pour tous les nœuds d'influence

$$\begin{cases} \int_{\alpha}^{\beta} \frac{\delta L_i}{\delta x} = [L_i]_{\alpha}^{\beta} \\ \int_{\alpha}^{\beta} \frac{\delta^2 N_i}{\delta x^2} = [\frac{\delta N_i}{\delta x}]_{\alpha}^{\beta} \\ \int_{\alpha}^{\beta} \frac{\delta^2 M_i}{\delta x^2} = [\frac{\delta M_i}{\delta x}]_{\alpha}^{\beta} \end{cases}$$

Ces contraintes s'écrivent comme précédemment et le même algorithme peut être utilisé pour calculer les poids d'intégration. Dans notre cas, la matrice des contraintes

s'écrit

$$g \in [1, n_g], i \in [1, n_i], \begin{cases} D_{g,3(i-1)+1} = \frac{\delta L_i}{\delta \xi}(\xi_g) - \frac{L_i(1) - L_i(-1)}{2} \\ D_{g,3(i-1)+2} = \frac{\delta^2 N_i}{\delta \xi^2}(\xi_g) - \frac{\frac{\delta N_i}{\delta \xi}(1) - \frac{\delta N_i}{\delta \xi}(-1)}{2} \\ D_{g,3(i-1)+3} = \frac{\delta^2 M_i}{\delta \xi^2}(\xi_g) - \frac{\frac{\delta M_i}{\delta \xi}(1) - \frac{\delta M_i}{\delta \xi}(-1)}{2} \end{cases}$$

Dans ce cas, le nombre de points d'intégration est très élevé car nous avons  $3n_i$  contraintes par pavé: ceci augmente le temps de calcul de l'intégration spéciale, mais nous avons vu dans le chapitre précédent qu'elle est nécessaire. De plus, dans l'application que nous présentons, la construction de la matrice de raideur n'est pas limitante. En pratique, nous utiliserons  $n_g = 3n_i + 3$  points de Gauss et  $m = n_g + 1$  monômes pour la minimisation de  $\|G\omega - \gamma\|$ .

### 3.4 Tests numériques

#### 3.4.1 Intégration (flexion)

Nous avons calculé les poids d'intégration adaptés à la pseudo-dérivée de l'approximation diffuse Hermite cubique seule. Deux exemples sont représentés figures 3.3. Les valeurs sont très éloignées des poids de Gauss-Legendre usuels! Ceci s'explique car la pseudo-dérivée des fonctions de forme  $N_i$  n'est pas toujours de bonne qualité: la forme de la fonction est correcte, mais pas l'amplitude comme le montre la figure 3.4. Nous devons donc utiliser la dérivée complète de l'approximation diffuse Hermite.

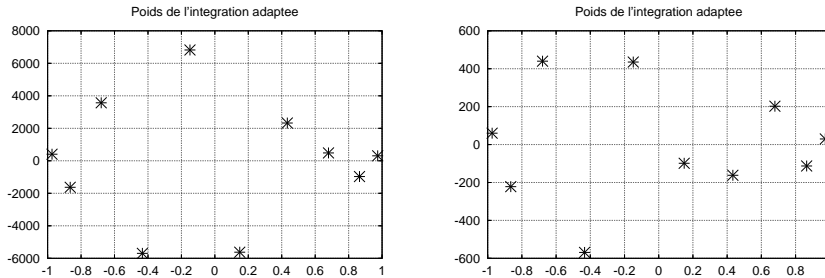


FIG. 3.3: Poids de l'intégration adaptée pour un patch test quadratique et l'approximation diffuse Hermite cubique.

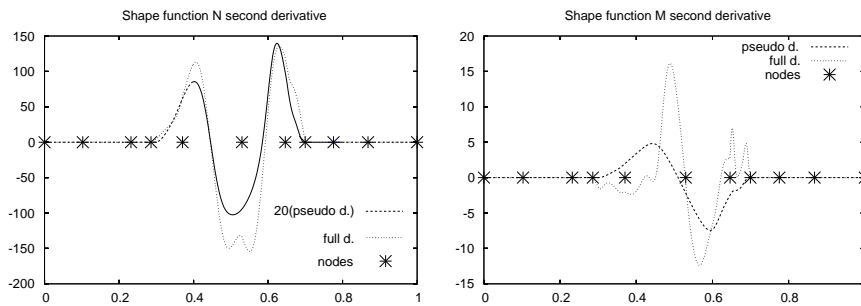


FIG. 3.4: Dérivée seconde de l'approximation diffuse Hermite.

Les études de sensibilité de l'intégration aux différents paramètres (nombre de points de Gauss, nombre de monômes adapté au défaut de rang de  $D$  ou non, ...) montrent des comportements identiques à ceux détaillés dans le chapitre précédent.

### 3.4.2 Cas tests (flexion)

Nous allons ici étudier la convergence de l'approximation diffuse Hermite pour l'étude de la flexion seule. L'équation différentielle de la flexion est

$$EIu^{(4)} + f = 0.$$

et l'intégration correspond au patch test d'ordre 2 (i.e. le patch test minimal).

Nous avons défini quatre cas tests qui représentent les différentes situations offertes par les problèmes de poutres.

**Poutre encastrée sous son propre poids** Le terme source  $f$  est simplement le poids  $-\rho g S$ , l'équation d'équilibre et l'encastrement nous mènent à

$$u(x) = -\frac{\rho g S}{24EI}x^4 + ax^3 + bx^2.$$

Les inconnues  $a$  et  $b$  se calculent en écrivant que  $M$  et  $T$  sont nuls en  $l$ , soit  $u^{(3)}(l) = 0$  et  $u''(l) = 0$ , et la solution s'écrit finalement (problème 1) :

$$u(x) = -u_0\xi^2 [\xi^2 - 4\xi + 6] \quad (3.4)$$

avec  $u_0 = \frac{\rho g S l^4}{24EI}$  et  $\xi = \frac{x}{l}$ .

**Poutre doublement encastrée** Le calcul est le même, mais les conditions aux limites s'écrivent maintenant  $u(l) = u_l$  et  $u'(l) = 0$ . Lorsque le poids est pris en compte, la déformée s'écrit (problème 3) :

$$u(x) = -u_0\xi^2 \left[ \xi^2 - 2 \left( 1 - \frac{u_l}{u_0} \right) \xi + \left( 1 - 3 \frac{u_l}{u_0} \right) \right] \quad (3.5)$$

Et en négligeant le poids (problème 2) :

$$u(x) = u_l \xi^2 [-2\xi + 3] \quad (3.6)$$

**Appuis simples et force ponctuelle** Le dernier cas est celui d'une poutre en appuis simple sur ses extrémités et supportant une force ponctuelle  $F$  au point  $y$ . La déformée de cette poutre est calculée dans [Lel90], et s'écrit (on pose  $u_1 = \frac{F l^3}{6EI}$ , et  $\eta = \frac{y}{l}$ , problème 4) :

$$\left\{ \begin{array}{l} \text{si } 0 < x < y \\ \quad u(x) = u_1 (1 - \eta) \xi [\eta (2 - \eta) - \xi^2] \\ \text{si } y < x < l \\ \quad u(x) = u_1 \eta (1 - \xi) [\xi (2 - \xi) - \eta^2] \end{array} \right. \quad (3.7)$$

Chacune de ces solutions analytiques s'exprime facilement en fonction des paramètres adimensionnels  $\xi$ ,  $\frac{u_l}{u_0}$  et  $\eta$ , et le facteur multiplicatif ne joue pas de rôle dans la suite.

Nous avons testé l'approximation diffuse Hermite avec consistance quadratique et cubique avec fonction d'influence voronoï d'ordre 2. Le nombre de points de Gauss supplémentaires est 4 et le nombre de monômes  $m$  est  $n_g + 1$ . Les taux de convergence sont donnés dans le tableau 3.2 et les courbes en annexe I. La référence est donnée par les éléments finis Hermite cubiques avec intégration analytique. Les résultats montrent que l'approximation diffuse Hermite est efficace pour résoudre ce genre de problème. Les taux de convergence dans le cas de l'approximation Hermite cubique sont parfois très élevés (problème 1 par exemple). Nous ne savons pas expliquer ces valeurs (super-convergence ou influence du calcul d'erreur?)

### 3.4.3 Poutres courbes

Nous considérons maintenant le modèle de poutre complet (i.e. avec les trois phénomènes: traction, torsion et flexion). On considère une poutre hélicoïdale encastrée à l'origine et soumise à des couples en son extrémité. L'état de contraintes de cette poutre correspond à un patch test. La figure 3.5 montre la convergence des contraintes. Le taux de convergence est environ -2, ce qui est le taux optimal. Le rebond à partir de 500 nœuds correspond au zéro machine pour les valeurs nodales (environ  $10^{-10}$ ).

### 3.4.4 Discussion des résultats

Les résultats que nous venons de montrer sont très encourageants, mais ils soulèvent plusieurs questions. Les taux de convergence observés pour la flexion seule (tableau 3.2) sont très élevés. Les expériences ont été menées avec une faible perturbation du semis de nœuds (5%). Nous ne savons pas expliquer ces taux: s'agit-il de super-convergence?

Le patch test pour une poutre de forme hélicoïdale est convergent. Les simulations d'assemblage de poutres courbes seront donc convergents. Cependant, le patch test au sens faible n'assure pas de taux de convergence comme au sens fort. Nous devons donc chercher les conditions nécessaires que doit vérifier l'intégration pour passer le patch test au sens fort. La principale difficulté est le couplage entre les différentes fonctions  $u_t, \omega_t, u_n, u_b$ . Une analyse sans l'hypothèse de Kirchhoff est peut-être plus aisée dans un premier temps.

Le choix de deux approximations différentes pour les termes  $u_t, \omega_t$  d'une part et  $u_n, u_b$  d'autre part a pour conséquence d'augmenter le nombre de points de Gauss nécessaires pour respecter les contraintes du patch test. Si le temps de calcul devient un point crucial, il est possible d'utiliser la même approximation pour toutes les fonctions. Ceci présente cependant trois désavantages:

1. On devra utiliser une approximation diffuse Lagrangienne de consistance cubique.
2. La gestion des discontinuités de  $u_n, u_b$  devra être revue.
3. On pourra avoir des problèmes liés aux conditions Brezzi-Babuska car l'ordre de dérivation des inconnues sont différents.

Nous avons aussi calculé les modes propres de poutres en utilisant le modèle diffus. L'expérience montre des résultats similaires à ceux du chapitre précédent. L'opérateur différentiel est donc bien représenté par l'opérateur discret.



|      |            | $u$   | $u' = \omega$ | $\omega' = \frac{M}{EI}$ |
|------|------------|-------|---------------|--------------------------|
| Pb 1 | E.F.       | -2.08 | -2.08         | -2.08                    |
|      | H.D. quad. | -2.27 | -2.20         | -1.02                    |
|      | H.D. cub.  | -4.59 | -3.78         | -2.45                    |
| Pb 2 | H.D. quad. | -2.27 | -2.19         | -1.08                    |
|      | H.D. cub.  | -2.95 | -2.00         | -0.85                    |
| Pb 3 | E.F.       | -2.97 | -3.03         | -2.03                    |
|      | H.D. quad. | -2.52 | -2.30         | -1.16                    |
|      | H.D. cub.  | -4.45 | -3.57         | -2.62                    |
| Pb 4 | H.D. quad. | -2.28 | -2.20         | -1.16                    |
|      | H.D. cub.  | -4.21 | -2.96         | -1.83                    |

TAB. 3.2: Taux de convergence des éléments diffus Hermite

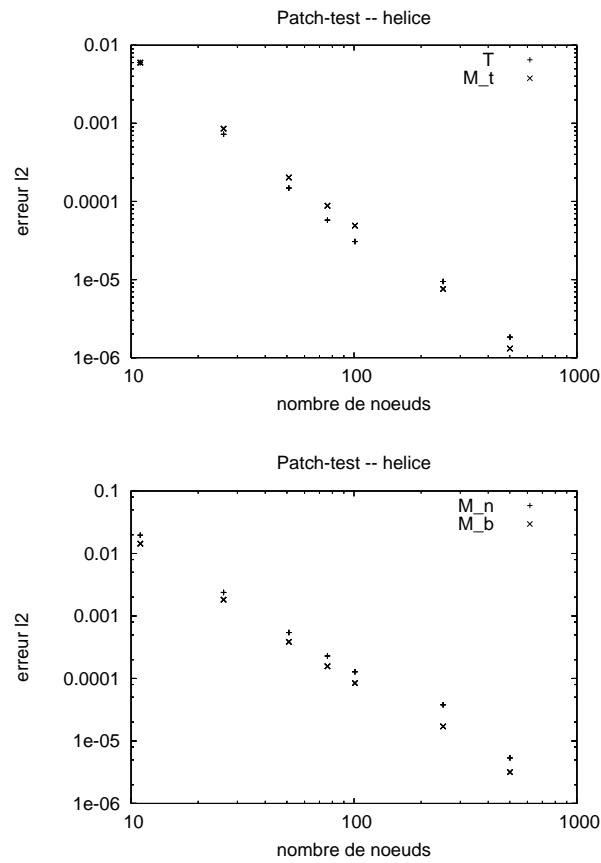


FIG. 3.5: Convergence des contraintes pour une poutre en hélice (patch test).

## 3.5 Conclusion

En conclusion, nous avons développé un modèle de poutre diffus en nous appuyant sur

- l’approximation diffuse Lagrangienne,
- l’approximation diffuse Hermitienne,
- des fonctions d’influence asymétriques,
- une intégration numérique adaptée aux fonctions de forme diffuses.

L’hypothèse mécanique de Kirchhoff nous permet de ramener la description de la poutre aux quatre fonctions  $u_t, \omega_t, u_n, u_b$  seulement. Elles sont approximées par

- l’interpolation diffuse Lagrangienne de consistance quadratique pour  $u_t$  et  $\omega_t$ ,
- l’interpolation diffuse hermitienne de consistance cubique pour  $u_n$  et  $u_b$ .
- les fonctions d’influence de type « Voronoï » et d’ordre deux sont utilisées pour les deux interpolations.

L’intégration numérique est construite à partir des patch tests et l’on utilise

- 3 points de Gauss supplémentaires par tuile,
- $n_g + 1$  monômes.

Avec ces paramètres, le modèle discret est convergent pour des poutres courbes. Nous allons donc l’utiliser dans la suite pour une application à la simulation de forage pétrolier. Dans cette application le surcoût de calcul dû à l’intégration numérique n’est pas limitant car on doit traiter les contacts roche–structure. Grâce à la continuité de l’approximation diffuse, nous pensons que les efforts seront mieux calculés par notre modèle de poutre que par un modèle éléments finis classique.



## Chapitre 4

# Application à un problème de forage

Nous avons développé un modèle de poutre discret dans la partie précédente. Nous montrons ici son utilisation pour traiter un problème de forage car nous pensons que l'utilisation de l'approximation diffuse peut améliorer l'estimation des contraintes dans la structure grâce à sa régularité. L'hypothèse de Kirchhoff doit permettre de traiter plus facilement des structures très élancées en diminuant les problèmes de blocage (dynamique du forage).

Cette partie se place dans le cadre d'une recherche active sur le forage au CGES (pétrolier en particulier). Nous avons donc bénéficié des méthodes existantes (contact 2d par exemple) mais l'application de l'approximation diffuse a aussi permis une évolution de deux aspects de la modélisation :

- la description de la géométrie d'une part,
- l'algorithme de contact 3d d'autre part.

Les détails sur la modélisation et l'état des recherches en 2d sont présentés dans [Mao99]. Nous nous focaliserons dans cette partie sur les techniques mises en œuvre pour simuler la physique du forage, et expliquerons pourquoi nous pensons que l'approximation diffuse peut permettre une meilleure estimation des efforts. Les détails sur le modèle ou le forage pétrolier se trouvent dans l'ouvrage cité précédemment [Mao99].

Nous commençons par décrire rapidement le problème physique et son modèle. L'algorithme de contact 2d est présenté puis les modifications nécessaires pour passer à 3 dimensions.

### 4.1 Problème posé

#### 4.1.1 Description du problème physique

L'industrie pétrolière utilise de plus en plus les forages déviés afin d'augmenter les taux de récupération de l'huile en place. Le contrôle de la trajectoire est l'un des facteurs essentiels dans la réussite d'un tel projet. Plusieurs technologies sont utilisées : systèmes de fond de trou actifs ou stabilisateurs en forage *rotary*. Nous allons étudier ici ce dernier cas car le forage *rotary* est moins coûteux et produit des puits plus faciles à équiper en général. La figure 4.1 montre les éléments principaux d'un forage rotary.

La garniture de forage est un tube qui transmet la rotation du moteur de surface à l'outil et permet la circulation des boues de forage dont le rôle est l'évacuation

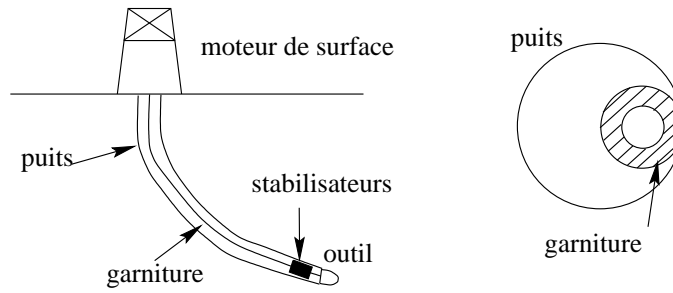


FIG. 4.1: Représentation schématique d'un système de forage.

des débris de roche. Le diamètre de ce tube varie, en particulier près de l'outil où se situent les stabilisateurs.

La simulation du forage rotary se décompose en trois étapes :

1. Simulation de la garniture de forage.
2. Mise en place d'un modèle d'interaction outil-roche.
3. Couplage de ces deux modèles.

Nous nous focalisons ici sur la première étape qui est un calcul statique de la mécanique de la garniture car l'ensemble du modèle est quasi-statique, il est décrit dans [Mao99]. Pour le calcul de trajectoire, seule la partie terminale de la garniture est simulée; en revanche, la dynamique du forage demande la simulation de l'ensemble de la structure.

### 4.1.2 Modélisation, géométrie

La garniture est modélisée comme une poutre (caractérisée par les grandeurs  $E, G, S, I, k_0$  car c'est un tube) en petits déplacements autour de l'axe du puits. Ceci suppose que l'on sache calculer les contraintes initiales correspondant à cet état d'une poutre initialement droite: ces contraintes dépendent de la géométrie du puits.

**Géométrie** La géométrie d'un puits est contruite à partir des mesures suivantes :

- longueur de puits foré,
- orientation de l'outil.

Ces données mènent à un problème d'interpolation original. On utilise en général une technique basée sur des arcs de cercle. Dans ce travail, nous avons choisi d'utiliser des arcs d'hélice qui permettent de décrire la torsion de la trajectoire et ainsi de mieux prendre en compte l'aspect 3d du problème traité.

Deux points de données successifs  $i, i + 1$  sont reliés par une hélice dont les courbures et torsion (qui sont constantes le long de l'hélice) sont calculés avec les directions des positions  $i - 1, i, i + 1$ . La géométrie obtenue interpole les mesures par une succession de segments de droite, d'arcs de cercle et d'arcs d'hélice. La construction géométrique des hélices est décrite dans [Sav99]. Comme précédemment, l'abscisse curviligne est notée  $s$ , ' la dérivation selon  $s$ , et  $(\vec{t}, \vec{n}, \vec{b})$  la base de Frenet le long de la trajectoire.

**Contraintes initiales** Les contraintes initiales associées à un arc de cercle ou un arc d'hélice sont simplement composées d'un moment constant [Tij99] :

$$\vec{M} = \begin{Bmatrix} \tau EI \\ 0 \\ \rho EI \end{Bmatrix}$$

## 4.2 Contact 2d

La principale difficulté pour la simulation de la mécanique de la garniture provient du contact avec la roche. Un algorithme qui s'appuie sur des forces nodales a été développé au CGES en 2d [Mao99]. Son intérêt principal est de ne pas modifier la matrice de raideur comme nous allons le voir dans la partie suivante. Ainsi, une fois la matrice  $K$  construite et triangulée, il suffit de résoudre des systèmes  $KU = F$  avec différents seconds membres. La construction de la matrice de raideur diffuse qui est plus longue que celle des éléments finis n'est plus le facteur limitant du calcul. De plus, les forces nodales appliquées au modèle diffus ne font pas apparaître de discontinuités de contraintes comme c'est le cas pour les éléments finis : « on rend le contact local ».

### 4.2.1 Description de l'algorithme

Supposons que l'on connaisse l'ensemble des nœuds de contact  $\mathcal{C}$ . Pour chaque nœud en contact, les conditions suivantes sont vérifiées (on note  $\vec{\nu}_c$  la direction du contact au point  $c$ , voir figure 4.2) :

–  $\|\vec{u}_c\| = e_c$  ( $e_c$  est le jeu i.e. la différence des rayons du puits et de la garniture).

– Les efforts de la roche sur la structure au nœud sont

$$\begin{aligned} \vec{F} &= -F_n^c \vec{\nu}_c + k_f F_n^c \vec{t} && \text{(frottement de Coulomb)} \\ \vec{C} &= k_m F_n^c \vec{t} && \text{(idem)} \end{aligned}$$

et ils vérifient

–  $F_n^c > 0$

où  $\vec{u}_c = u_n \vec{n} + u_b \vec{b}$  est la projection du déplacement du nœud  $c$  sur le plan  $(\vec{n}, \vec{b})$ . La condition  $\vec{u}_c \cdot \vec{\nu}_c = e_c$  est équivalente à  $\|\vec{u}_c\| = e_c$ , et comme  $\vec{\nu}$  est orthogonal à  $\vec{t}$ , la condition peut aussi s'écrire sur le déplacement total du nœud. Nous utiliserons cette deuxième forme car elle est **linéaire**.

Notons  $\vec{a}_c^d$  le déplacement du nœud  $d$  dû à des efforts de contact unitaires (i.e.  $F_n^c = 1$ ) appliqués au nœuds  $c$  et  $\vec{u}_d^o$  le déplacement de la structure sans contact. Le déplacement d'un point de la structure s'écrit par le principe de superposition

$$\vec{u} = \vec{u}^o + \sum_{c \in \mathcal{C}} F_n^c \vec{a}_c$$

et en particulier pour le nœud  $d$

$$\vec{u}_d = \vec{u}_d^o + \sum_{c \in \mathcal{C}} F_n^c \vec{a}_c^d$$

En écrivant la première condition de contact en chaque nœud, i.e.

$$\sum_{c \in \mathcal{C}} \vec{a}_c^d \cdot \vec{\nu}_d F_n^c = e_d - \vec{u}_d^o \cdot \vec{\nu}_d, \quad \forall d \in \mathcal{C}$$

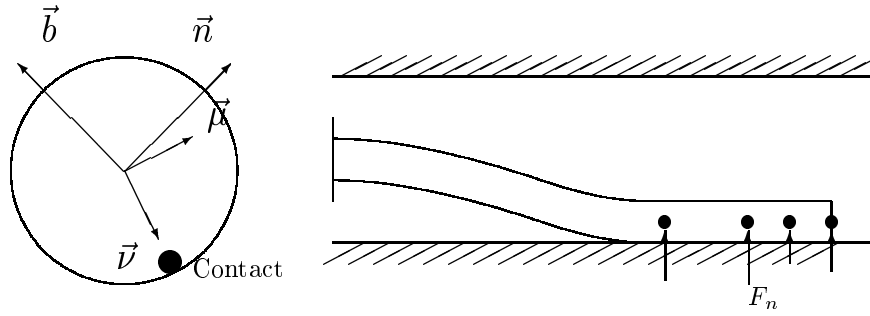


FIG. 4.2: Direction de contact, forces ponctuelles de contact aux nœuds.

on obtient un système linéaire  $AF = B$  qui permet de calculer les efforts aux nœuds de l'ensemble de contact. Notons que dans le cas où l'on néglige les frottements, la matrice  $A = [\vec{a}_c^d \cdot \vec{v}_d]_{d_c}$  de ce système linéaire est une sous-matrice de  $K^{-1}$  : ceci assure son inversibilité (elle est même définie positive). Avec frottements, la matrice  $A$  est toujours inversible mais non symétrique.

La méthode que nous venons de décrire permet de calculer les forces de contacts lorsqu'on connaît *a priori* les nœuds de contact. Pour calculer cet ensemble, nous allons itérer le calcul précédent en faisant évoluer l'ensemble des nœuds de contact. A chaque itération, on :

- ajoute le nœud le plus pénétrant,
- ôte le nœud le plus en traction (i.e. s'il existe des nœuds tels que  $F_n^c < 0$ , on enlève le plus négatif).

Ces deux règles permettent de construire un algorithme convergent (d'après [Tij99]). Le critère d'arrêt est la stationnarité de la suite des ensembles de nœuds de contact, sachant que les deux conditions ci-dessus sont respectées à des tolérances près fournies par l'utilisateur en tant que précisions respectivement sur le déplacement et sur la force.

### 4.2.2 Exemple

Pour illustrer le fonctionnement de l'algorithme que nous venons de décrire, nous allons traiter un exemple simple : on considère un élément de garniture horizontal dans un puits, encastré à gauche et sous son propre poids (voir figure 4.2). Il est possible de calculer la flèche analytique d'une telle poutre :

$$\begin{cases} j \text{ le jeu} \\ l_c = \left( \frac{72EIj}{\rho g S} \right)^{1/4} \\ u(s) = \begin{cases} -\frac{\rho g S l_c^4}{72EI} \left( 1 + 4\left(\frac{s}{l_c} - 1\right)^3 + 3\left(\frac{s}{l_c} - 1\right)^4 \right) & \text{si } s < l_c \\ j & \text{si } s \geq l_c \end{cases} \end{cases}$$

La figure 4.4 montre les étapes de l'algorithme de contact 2d.

Comme on connaît la solution analytique du problème précédent, il est possible d'étudier la convergence de l'ensemble poutre diffuse et algorithme de contact. La figure 4.5 montre l'erreur en norme  $\mathcal{L}^2$  contre le nombre de nœuds pour la flèche, la rotation de la section droite et le moment (le cas est de la flexion pure). Les nœuds sont générés avec une faible perturbation de 2.5%. La simulation mise en

place converge pour ce cas et les taux de convergence sont donnés dans le tableau 4.1. Les courbes ne sont pas régulières car la position du premier nœud de contact par rapport à  $l_c$  joue un rôle important dans l'erreur.

|               |       |
|---------------|-------|
| $u$           | -2.64 |
| $\omega = u'$ | -2.11 |
| $M = EIU''$   | -1.67 |

TAB. 4.1: *Taux de convergence pour le problème du puits horizontal (précision sur le déplacement  $10^{-10}$ , précision sur les forces 0).*

La figure 4.3 montrent que le nombre d'itérations augmente lorsque la précision demandée sur les positions augmente. La relation entre le logarithme du nombre d'itération est le logarithme de l'erreur est linéaire (pente -0.2). Le temps de calcul augmente rapidement avec le nombre d'itération et le nombre de nœuds en contact, mais la précision sur le jeu est faible ce qui nous autorise à utiliser une précision sur les positions élevées. Sur notre cas, la précision sur les forces a peu d'influence sur le nombre d'itérations.

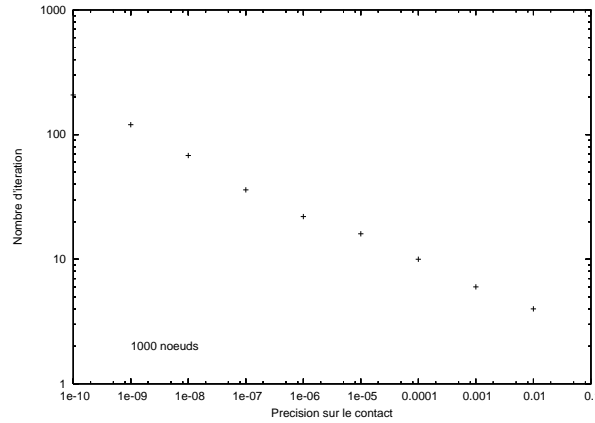


FIG. 4.3: *Évolution du nombre d'itérations en fonction de la précision sur les positions (précision sur les forces 0, 1000 nœuds).*

### 4.2.3 Discussion

Avant toute chose, notons que les frottements entre la garniture et le puits sont calculés de façon simplifiée grâce à la connaissance a priori sur le système physique. En pratique les coefficients  $k_f, k_c$  peuvent être signés pour tenir compte du déplacement selon  $\vec{t}$  ( $k_f$  est négatif si  $u_t$  est positif) et du sens de rotation de la garniture. Par ailleurs, l'utilisation du repère de Frenet simplifie ce test.

L'intérêt du modèle diffus est principalement lié à la forme de l'algorithme de contact. Comme le contact est pris en compte par des forces nodales, un modèle élément finis fait apparaître des discontinuités de contraintes aux nœuds en contact. Dans le cas précédent, le moment est la dérivée seconde de la flèche, et des éléments Hermite cubiques n'assurent qu'une approximation  $C^1$  alors que le modèle diffus admet la régularité de la fonction  $w_{ref}$  (en pratique  $C^2$  avec la spline quartique). On peut donc espérer une meilleure estimation des efforts sur l'outil de forage grâce à la régularité de l'approximation. Des tests avec paramètres réels devront être menés pour le vérifier.



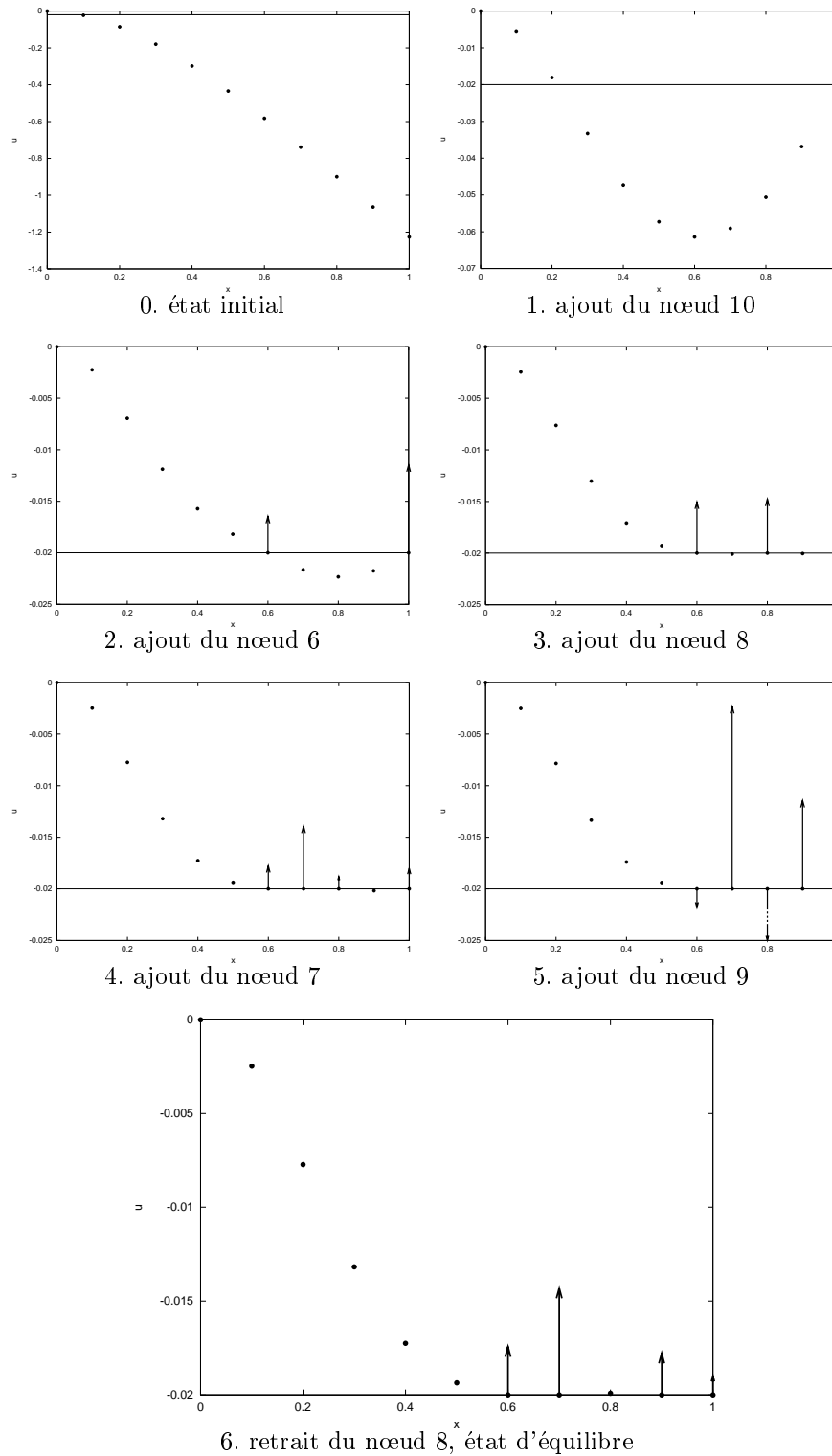


FIG. 4.4: Exemple de calcul de contact garniture-puits. La courbe horizontale représente la paroi du puits.

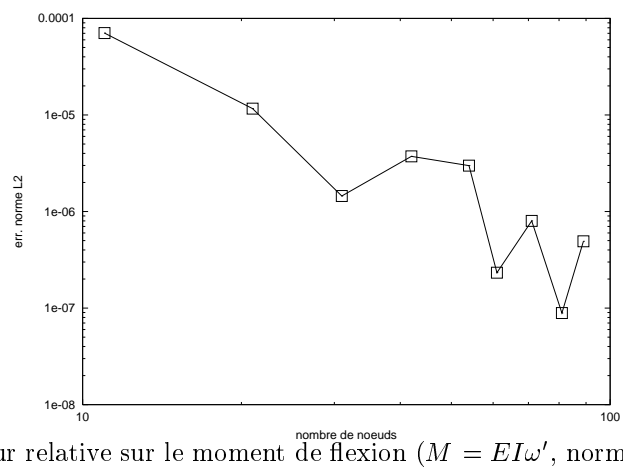
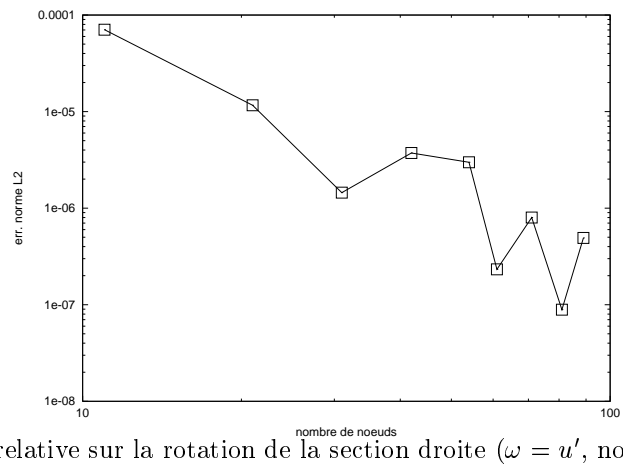
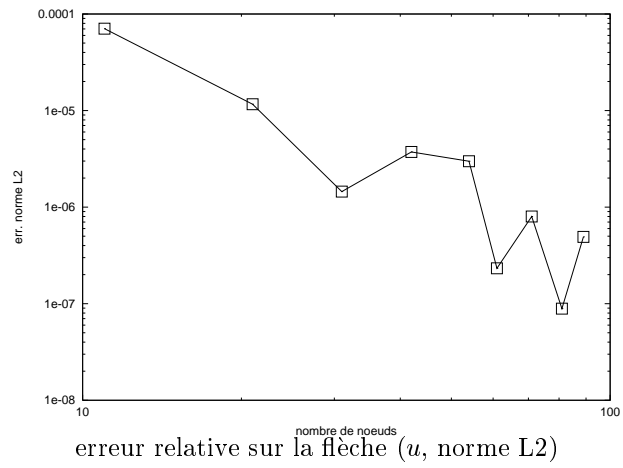


FIG. 4.5: Convergence pour le problème du forage horizontal.

L'algorithme de contact que nous avons mis en place n'est pas transposable directement en 3d : en 2d la condition « linéarisée »  $\vec{u} \cdot \vec{\nu} = j$  assure la condition de contact exactement et le choix du point de contact est évident entre les deux candidats sur la section du puits (« du côté où l'on pénètre la roche »). En 3d, les choses sont bien différentes car la condition géométrique  $\vec{u} \cdot \vec{\nu} = j$  n'est suffisante pour assurer que le contact est sur la frontière du puits que si  $\vec{\nu}$  est vraiment la direction de contact. Ainsi le passage de 2d à 3d s'accompagne d'un problème supplémentaire : celui de la détermination des directions  $\vec{\nu}$

Deux possibilités se présentent pour calculer le contact en 3d :

1. Ne plus utiliser la version linéarisée de la première condition de contact  $\vec{u} \cdot \vec{\nu} = e$  mais revenir à  $\|\vec{u}\| = e$ .
2. Trouver une technique pour calculer les « bonnes » positions de contact et utiliser la technique précédente pour calculer les efforts.

La première technique demande de résoudre un problème non-linéaire, nous avons donc inventé une technique hybride où le calcul des forces de contact est couplé à la méthode de Newton pour trouver les points de contact.

## 4.3 Contact 3d

### 4.3.1 Description

Le calcul de l'équilibre de la structure en contact avec le puits suit le même cheminement que précédemment (on itère sur l'ensemble des nœuds en contact  $\mathcal{C}$ ). On initialise la direction de contact d'un nœud entrant dans l'ensemble des nœuds en contact comme la direction de son déplacement à cette étape, i.e.  $\vec{\nu} = \frac{\vec{u}}{\|\vec{u}\|}$ . En 3d, la première condition de contact est équivalente au système d'équations

$$\begin{cases} \vec{u}_c \cdot \vec{\nu}_c = e_c, & c \in \mathcal{C} \\ \vec{u}_c \cdot \vec{\mu}_c = 0, & c \in \mathcal{C} \end{cases}$$

avec  $\vec{\mu}_c = \vec{\nu}_c \wedge \vec{t}$ . Nous allons résoudre ce système non linéaire (les vecteurs  $\vec{\nu}$  sont des inconnues) en résolvant alternativement sur les forces  $F_n^c$  et les directions  $\vec{\nu}_c$ . Les forces sont calculées à partir du premier système d'équations avec la méthode précédente. Puis les nouvelles directions de contact en annulant les tangentes des angles entre les vecteurs  $\vec{\mu}_c$  et  $\vec{u}$  en chaque nœud de contact, soit

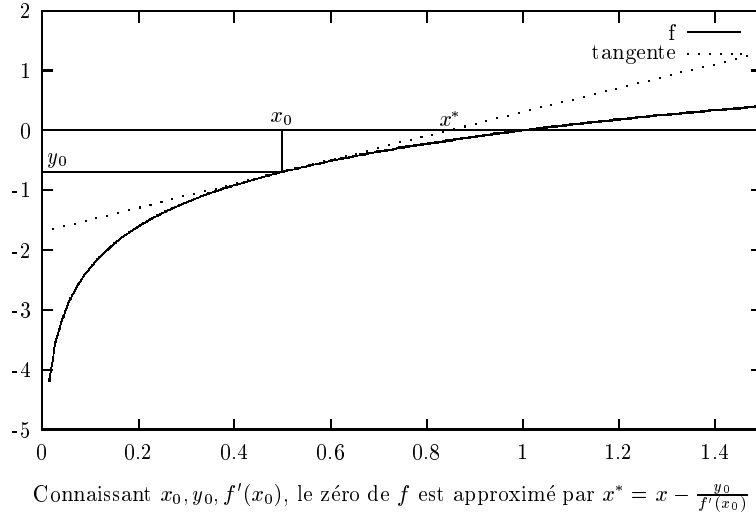
$$T_c = \vec{u} \cdot \vec{\mu}_c / e_c$$

Nous utilisons ici une linéarisation du second jeu d'équations (méthode de Newton, figure 4.6) pour modifier les directions  $\vec{\nu}_c$ . Ces deux phases sont itérées jusqu'à l'obtention d'une solution.

### 4.3.2 Calcul élémentaire des directions

Chaque tangente  $T_c$  est fonction de tous les vecteurs  $\vec{\nu}_d$  qui sont paramétrés par des angles  $\theta_d$  (ce ne sont que des directions). La méthode de Newton calcule les variations angulaires solutions du système d'équation

$$\begin{bmatrix} \frac{\partial T_1}{\partial \theta_1} & \dots & \frac{\partial T_1}{\partial \theta_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial T_m}{\partial \theta_1} & \dots & \frac{\partial T_m}{\partial \theta_m} \end{bmatrix} \begin{bmatrix} \Delta \theta_1 \\ \vdots \\ \Delta \theta_m \end{bmatrix} = - \begin{bmatrix} T_1 \\ \vdots \\ T_m \end{bmatrix}$$

FIG. 4.6: *Algorithme de Newton en 1d pour calculer le zéro d'une fonction.*

et on modifie chaque direction  $\vec{v}_c$  en normalisant le vecteur (on suppose  $\Delta\theta_c$  petit)

$$\vec{v}_c^* = \vec{v}_c + \Delta\theta_c \vec{\mu}_c$$

Évaluons les dérivées partielles  $\frac{\partial T_c}{\partial \theta_d}$ . Deux cas se distinguent :

– Si  $c = d$ , nous avons

$$\begin{cases} T_c = \vec{v}_c \cdot \vec{\mu}_c / e_c \\ \frac{\partial \vec{v}_c}{\partial \theta_c} = \vec{\mu}_c \\ \frac{\partial \vec{\mu}_c}{\partial \theta_c} = -\vec{v}_c \\ \vec{v}_c \cdot \vec{u}_c = e_c \end{cases}$$

et donc la variation de  $T_c$  est

$$\frac{\partial T_c}{\partial \theta_c} = \left( \frac{\partial \vec{u}_c}{\partial \theta_c} \cdot \vec{\mu}_c - \vec{u}_c \cdot \vec{v}_c \right) / e_c = \frac{\partial \vec{u}_c}{\partial \theta_c} \cdot \vec{\mu}_c / e_c - 1$$

or

$$\frac{\partial \vec{u}_c}{\partial \theta_c} = \sum_{e \in \mathcal{C}} \frac{\partial F_n^e}{\partial \theta_c} \vec{a}_e^c + F_n^c \vec{b}_c^c$$

avec  $\vec{b}_c^d$  le déplacement du nœud  $d$  lorsque le nœud  $c$  est soumis à une force  $-\vec{\mu}_c - B_c \vec{v}_c$ . Pour cette dernière égalité, on utilise le fait que  $\frac{\partial \vec{a}_c^d}{\partial \theta_c} = \vec{b}_c^d$ . Il reste à calculer les  $\frac{\partial F_n^d}{\partial \theta_c}$ .

– Si  $c \neq d$ , il vient

$$\frac{\partial T_c}{\partial \theta_d} = \frac{\partial \vec{u}_c}{\partial \theta_d} \cdot \vec{\mu}_c$$

et

$$\frac{\partial \vec{u}_c}{\partial \theta_d} = \sum_{e \in \mathcal{C}} \frac{\partial F_n^e}{\partial \theta_d} \vec{a}_e^c + F_n^c \vec{b}_d^c$$

Calculons maintenant les  $\frac{\partial F_n^c}{\partial \theta_d}$ . Les  $F_n^c$  sont solutions du système  $AF = B$ , donc leurs dérivées partielles de

$$A \frac{\partial F}{\partial \theta_d} = \frac{\partial B}{\partial \theta_d} - \frac{\partial A}{\partial \theta_d} F$$

Tous calculs faits, le second membre de ce système s'écrit

$$-F_n^d \begin{pmatrix} \vec{b}_d^1 \cdot \vec{\nu}_1 \\ \vdots \\ \vec{b}_d^m \cdot \vec{\nu}_m \end{pmatrix} + \begin{pmatrix} 0 \\ \vdots \\ -\vec{u}_d^o \cdot \vec{\mu}_d - \sum_{e \in \mathcal{C}} F_n^e \vec{a}_e^d \cdot \vec{\mu}_d \\ \vdots \\ 0 \end{pmatrix} \quad (d^{\text{ième}} \text{ ligne})$$

### 4.3.3 Mise en œuvre

À chaque étape de l'algorithme de contact, on doit résoudre pour chaque nœud dans  $\mathcal{C}$  deux problèmes d'élasticité (calcul de  $\vec{a}_c^d$  et  $\vec{b}_c^d$ ). Cependant la matrice de raideur a une très faible largeur de bande (24) et la résolution des systèmes d'élasticité est très rapide; ceci est d'autant plus vrai que la matrice de raideur n'est triangulée qu'une fois. En revanche, le calcul des forces  $F_n^c$  et des directions  $\vec{\nu}_c$  devient lourd lorsque le nombre de nœuds de contact augmente car la matrice  $A$  est pleine et non symétrique (frottement).

Notons aussi que l'implémentation de l'algorithme de contact demande du soin pour éviter d'encombrer la mémoire ou de recalculer de trop nombreuses fois les mêmes éléments. En effet, le calcul du contact pour le cas 2d présenté dans la partie précédente est 40 fois plus long que la construction de la matrice de raideur (pour 100 nœuds)! En 3d ce chiffre est encore plus élevé du fait de l'itération sur les directions de contact.

Les tests numériques ont montré que la technique proposée (modèle élastique «  $KU = F$  » + contact, avec discrétisation élément finis ou diffus) pour calculer l'équilibre de la garniture dans le puits est convergente dans sa version 3d (i.e. on arrive bien à calculer un état d'équilibre). La convergence des solutions calculées n'a pu être testée numériquement car nous ne connaissons pas de solution analytique de problèmes 3d. Des expériences devront être construites pour comparer ces solutions avec la réalité.

## 4.4 Conclusion

En conclusion nous avons proposé dans ce chapitre une technique basée sur les éléments diffus pour calculer l'équilibre de la garniture dans un puits au cours d'un forage dévié. L'utilisation de l'approximation diffuse permet de calculer des efforts plus lisses que les éléments finis: ceci devrait permettre une meilleure estimation des efforts à l'outil et ainsi améliorer la prédiction des trajectoires de forage. La convergence de la méthode a été démontrée pour un cas 2d<sup>1</sup>. L'algorithme de contact a été étendu en 3d grâce à une méthode originale.

Au cours de ce travail, nous avons apporté deux originalités par rapport aux méthodes existantes:

- La géométrie est décrite de façon plus riche avec l'utilisation d'arcs d'hélice.

<sup>1</sup>. Nous n'avons pas proposé de comparaison avec les éléments finis car le code dont nous disposons ne s'appuie pas sur l'hypothèse de Kirchoff et utilise des éléments Lagrangiens quadratiques pour les déplacements et les rotations

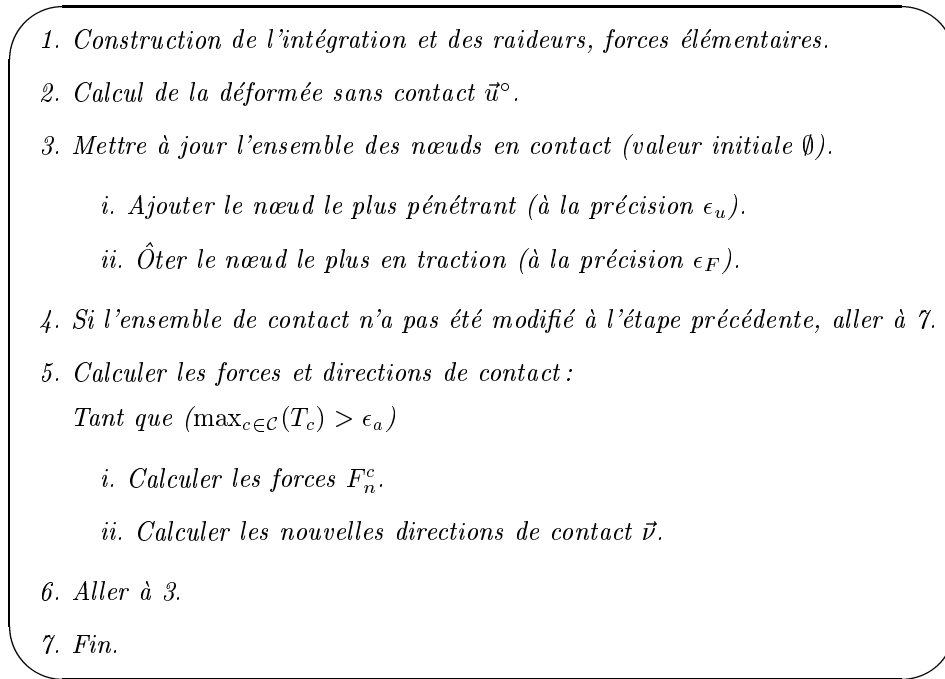


FIG. 4.7: *Séquence pour la simulation de la garniture.*

- L’algorithme de contact a été étendu au cas 3d en conservant sa robustesse.

Chacun de ces deux points présentent un grand intérêt pour l’étude du comportement azimutal des systèmes *rotary*. La construction du modèle diffus ouvre ainsi de nombreuses perspectives comme

- la construction d’une expérience pour comparer les solutions éléments finis et diffuse avec la réalité,
- l’utilisation du modèle diffus pour la simulation complète du forage, comparaison à des cas réels et aux résultats éléments finis
- l’ajout de couples de contact pour aligner la tangente de la déformée avec celle du puits,
- ...



## Cinquième partie

# Conclusion





# Conclusions et perspectives

Dans ce travail, nous avons abordé trois thèmes de recherche qui sont :

- L’approximation,
- Le maillage et la modélisation géométrique,
- La résolution numérique des équations aux dérivées partielles.

Dans la partie approximation, nous avons défini et étudié les outils de base de notre travail. L’approximation diffuse Hermite y est construite et analysée. L’application au maillage montre la puissance des techniques de moindres carrés glissants et offre des résultats encourageants sur un exemple d’application. La dernière partie contribue à l’amélioration des techniques de résolution d’équations différentielles sans maillage en définissant et analysant une méthode d’intégration numérique originale.

## Approximation

La notion de pseudo-dérivée est utilisée dans chaque application et montre ainsi son efficacité. Ce constat met en évidence l’une des deux propriétés essentielles de l’approximation diffuse : la facilité d’estimer des dérivées de grande régularité. La seconde caractéristique remarquable de l’approximation diffuse est l’absence de maillage, qui est remplacée par la notion de fonction d’influence. Comme les splines ou le krigeage ne s’appuient pas non plus sur un maillage pour calculer une approximation, nous définissons l’approximation myope : un pont entre les interpolateurs radiaux et l’approximation diffuse. Sa construction s’appuie sur une compréhension approfondie des splines d’ajustement et permet de montrer les différences entre les deux types d’approximation. Elle permet aussi une analyse originale de l’équivalence splines-krigeage basée sur le calcul explicite des représentants des formes linéaires  $L_i$  et le centrage du système de construction des splines d’ajustement (propriété 22, page 60). Les fonctions de forme myopes sont caractérisées par un système hybride et leurs pseudo-dérivées définies.

Nous avons testé les fonctions de forme myope pour la résolution d’équations différentielles, mais les résultats, présentés dans [SVS99], ne montrent pas d’améliorations particulières vis-à-vis d’une solution diffuse classique. Ce développement n’est pourtant pas un échec car il montre clairement les différences et points communs entre approximation diffuse et krigeage. L’approximation myope se présente aussi comme un bon candidat pour régulariser les krigeages en voisinages glissants.

Notre seconde contribution est la construction et l’étude de l’approximation diffuse Hermite. Nous suivons encore le même cheminement : pseudo-dérivation,

fonctions de forme et convergence (théorèmes 9 et 10). L'approximation diffuse Hermite est à la base des deux autres thèmes abordés : modélisation géométrique et équations différentielles.

En conclusion, l'étude de l'approximation diffuse nous a permis d'en proposer des extensions :

- L'approximation myope est une exploration théorique importante pour situer la technique des moindres carrés glissants vis-à-vis des interpolateurs radiaux.
- L'approximation Hermite permet de prendre en compte des données différentielles (la méthode se généralise facilement à d'autres données que de simples dérivées).

En parallèle, nous avons utilisé l'approximation diffuse lagrangienne pour de nombreuses applications pratiques comme le transport de champ entre simulations ou le lissage et la dérivation de signaux.

## Modélisation géométrique

L'approximation diffuse Hermite permet de développer une méthode d'estimation de la courbure de surfaces triangulées. Cette technique est un raffinement de la méthode d'estimation usuelle par ajustement, elle en améliore grandement la résolution. Nous avons proposé une étude expérimentale approfondie de la méthode et une comparaison à d'autres techniques (méthode de Samson, méthode duale) qui montrent la qualité de ses résultats.

Les propriétés du calcul des courbures nous permettent de construire un algorithme de reconnaissance des surfaces performant. L'étude théorique de la méthode nous permet de démontrer sa convergence (théorème 11) et de discuter en détail ses conditions d'application, les principaux résultats sont décrits dans [RSV99]. La technique de calcul duale est utilisée par A. Rassineux *et al.* pour reconstruire et adapter des maillages surfaciques [RSSV99].

Ainsi l'application de l'approximation Hermite à la modélisation géométrique est un succès, elle ouvre de nombreuses perspectives. Les courbures calculées aux nœuds sont de bons indicateurs de la rugosité de la surface, elles permettent de détecter des traits particuliers des surfaces comme :

- Les failles sur une surface géologique (projet en cours),
- les arêtes sur un modèle CAO,
- ...

La reconnaissance des surfaces est elle aussi un domaine prometteur, cependant deux points doivent encore être améliorés :

- Les surfaces reconnues sont très simples, et l'ajout d'autres surfaces élémentaires (surfaces de révolution par exemple) est nécessaire.
- Le paramétrage des différents opérateurs basés sur la courbure est peu automatisé.

Ces deux développements sont des étapes vers une reconstruction (au moins partielle) d'un modèle CAO à partir d'une surface triangulée.

## Équations différentielles

La dernière partie de notre travail concerne la résolution numérique des équations différentielles. Les techniques sans maillages sont maintenant reconnues par la communauté scientifique comme le montrent les sessions spécifiques dans la plupart des congrès internationaux. Les domaines d'applications sont de plus en plus variés; l'élasticité linéaire utilisée pour étalonner les méthodes a fait place à des problèmes plus complexes comme les grandes déformations [JLB98, BK98], la mécanique des fluides [OIZT96, CS98], le calcul d'impact [BBAC99], ou la dynamique [LJL+95]. Parallèlement les chercheurs ont continué à développer leurs méthodes avec la construction de nouvelles techniques d'approximation pour le calcul multi-échelle par exemple ou encore de l'intégration nodale. Cependant, nous avons montré dans l'étude bibliographique que certains aspects théoriques ne sont pas solides. La prise en compte des conditions aux limites ou l'intégration numérique ne sont pas satisfaisantes. Nous avons donc choisi de nous pencher sur ces problèmes fondamentaux pour donner une base solide à l'ensemble des applications.

Notre principale contribution est la mise en œuvre et le test d'une méthode d'intégration numérique adaptée à l'interpolation utilisée et qui assure le patch test (quatrième partie, chapitre 2). Dans ce chapitre, nous avons

- Décrit et analysé la technique d'intégration adaptée,
- Testé la sensibilité à chacun de ses paramètres,
- Montré qu'elle est suffisante pour obtenir de bons taux de convergence.

La technique est ainsi bien établie et nous permet d'envisager son extension à 2 ou 3 dimensions. L'intégration numérique adaptée a aussi été étendue à l'étude de la flexion des poutres (chapitre 3). Pour cela, nous avons proposé

- Une analyse du patch test pour la flexion.
- L'extension de la technique de calcul de l'intégration adaptée pour la flexion.
- Son application à l'approximation Hermite.

Nous pouvons alors construire un modèle de poutre 3d pour résoudre un problème issu du forage pétrolier en

- Améliorant la description de la géométrie (arcs d'hélice),
- Définissant un algorithme de contact 3d.

Le problème non-linéaire (contact) semble mieux résolu grâce à l'approximation diffuse qui conserve la continuité des efforts dans la structure. Le modèle a été exposé dans [SVT+00, SVTB00].

D'un point de vue plus général, la méthode d'intégration proposée permet d'ouvrir de nombreux thèmes de recherche :

- On peut utiliser la pseudo-dérivée pour construire une méthode convergente (cas Lagrangien). Ainsi, on pourra utiliser toute la puissance de ce concept (par exemple, construire des fonctions de forme avec des propriétés pseudo-différentielles données).
- La gestion des conditions aux limites peut être étudiée sans que l'intégration en masque certains effets.
- L'analyse multi-échelle et des indicateurs d'erreurs pourront être développés.



## Sixième partie

# Annexes



## Annexe A

# Calcul de l'approximation diffuse

Le calcul de l'approximation diffuse demande la résolution d'un système linéaire en chaque point. Lorsqu'on cherche les fonctions de forme, il s'agit même d'un système linéaire à second membre multiple. Ceci mène à la construction de méthodes spécifiques pour le calcul des fonctions de forme diffuses. Les algorithmes utilisés pour l'approximation diffuse Lagrangienne se traduisent très facilement au cas Hermitien, nous nous contenterons donc du premier cas.

Nous proposons l'algorithme adapté au cas monodimensionnel qui est utilisé pour le calcul de poutre par exemple. Nous décrivons ensuite l'algorithme proposé par Breikopf *et. al.* dans [BTV99a] qui est plus général. Les complexités des deux techniques numériques sont analysées et comparées.

### A.1 Cas 1d, dérivée réelle

**Calcul des fonctions de forme :** On part de la caractérisation des fonctions de forme diffuses de Guyon [Guy95a] :

$$\left\{ \begin{array}{l} \min \sum_{i \in I(x)} \frac{N_i^2}{w_i(x)} \\ \sum_{i \in I(x)} N_i(x) = 1 \\ \forall k \in [1, d], \sum_{i \in I(x)} (x_i - x)^k N_i(x) = 0 \end{array} \right. \quad (\text{A.1})$$

Ce qui conduit au système de construction :

$$\begin{bmatrix} W(x)^{-1} & P(x) \\ P^\top(x) & 0 \end{bmatrix} \begin{Bmatrix} N(x) \\ \lambda(x) \end{Bmatrix} = \begin{Bmatrix} 0 \\ e_1 \end{Bmatrix}$$

où

- $W(x)$  est la matrice diagonale des  $w_i(x)$ ,
- $P(x)$  est la matrice des contraintes (conditions de consistance centrées) :

$$P^\top(x) = \begin{bmatrix} 1 & \dots & 1 \\ x_1 - x & \dots & x_{n_x} - x \\ \vdots & & \vdots \\ (x_1 - x)^d & \dots & (x_{n_x} - x)^d \end{bmatrix}$$



$$- e_1 = \langle 1, 0, \dots, 0 \rangle^\top.$$

Et ce système se résout en

$$\begin{cases} N(x) &= -W(x)P(x)\lambda(x) = W(x)P(x)(P^\top(x)W(x)P(x))^{-1}e_1 \\ \lambda(x) &= -(P^\top(x)W(x)P(x))^{-1}e_1 \end{cases}$$

Notons que les expressions ci-dessus montrent que la fonction de forme  $N_i$  admet la forme (et ceci définit la fonction  $n_i$ )

$$N_i(x) = -w_i(x) \left( \sum_{k=1}^d \lambda_k(x)(x_i - x)^k \right) = -w_i(x)n_i(x) \quad (\text{A.2})$$

On reconnaît ici la formulation de RKPM où  $N_i$  est le produit de  $w_i$  et d'une fonction de correction polynomiale  $C(x, x_i - x) = -n_i(x)$ . Les coefficients de ce polynôme sont ceux de RKPM si

- La matrice des moments est calculée de façon discrète.
- Les poids d'intégration  $V_i$  sont tous égaux.

Nous utiliserons dans la suite, les fonctions  $\chi_i(x)$ ,  $i \in I(x)$  définies par

$$\chi_i(x) = \sum_{k=1}^d \lambda_k(x)k(x_i - x)^{k-1}$$

**Dérivées des fonctions de forme :** L'expression des fonctions de forme montre qu'il est pénible de calculer leurs dérivées, cette tâche consomme d'ailleurs beaucoup de CPU [Kro96]. Le cas de la dérivée diffuse est beaucoup moins pénalisant, car il suffit de remplacer  $e_1$  par  $e_i$  pour obtenir la pseudo-dérivée  $i^{\text{ième}}$  [Guy95a]. De plus, l'algorithme de Breitkopf [BTV99a] offre une méthode rapide de résolution du système (A.1) avec des seconds membres du type  $E_i = \langle 0, e_i \rangle^\top$ .

Penchons nous maintenant sur le calcul des dérivées « réelles » des fonctions de forme  $N_i$ . Pour cela, nous allons noter  $\mathcal{A}(x)$  la matrice du système (A.1) et  $M(x)$  le vecteur  $\langle N(x), \lambda(x) \rangle^\top$ . Le calcul de  $M'(x)$  mène au système :

$$\begin{bmatrix} \mathcal{A}'(x) & \mathcal{A}(x) \\ \mathcal{A}(x) & 0 \end{bmatrix} \begin{Bmatrix} M(x) \\ M'(x) \end{Bmatrix} = \begin{Bmatrix} 0 \\ E_1 \end{Bmatrix}$$

et finalement, on obtient  $M(x) = -\mathcal{A}^{-1}(x)\mathcal{A}'(x)\mathcal{A}^{-1}(x)E_1$ .

Le cas de la dérivée seconde s'écrit

$$\begin{bmatrix} \mathcal{A}''(x) & 2\mathcal{A}'(x) & \mathcal{A}(x) \\ \mathcal{A}'(x) & \mathcal{A}(x) & 0 \\ \mathcal{A}(x) & 0 & 0 \end{bmatrix} \begin{Bmatrix} M(x) \\ M'(x) \\ M''(x) \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ E_1 \end{Bmatrix}$$

soit  $M''(x) = \mathcal{A}^{-1}(x) (2\mathcal{A}'(x)\mathcal{A}^{-1}(x)\mathcal{A}'(x) - \mathcal{A}''(x)) \mathcal{A}^{-1}(x)E_1$ . De même, le calcul des dérivées successives de  $N(x)$  ne nécessite « que » l'inversion de  $\mathcal{A}(x)$ . Or cette matrice a une forme très particulière que l'on peut exploiter.

En premier lieu, on peut chercher une décomposition LU de  $\mathcal{A}(x)$  :

$$\mathcal{A}(x) = \begin{bmatrix} S^{-1}(x) & 0 \\ \eta^\top(x) & R^\top(x) \end{bmatrix} \begin{bmatrix} S^{-1}(x) & \eta(x) \\ 0 & -R(x) \end{bmatrix}$$

avec

- $S^{-1}(x)$  la matrice diagonale des  $\sqrt{w_i(x)}$ ,
- $\eta(x) = S(x)P(x)$ ,
- $R(x)$  la matrice de la décomposition Cholesky de  $A(x) = P(x)^\top W(x)P(x)$ .

On est donc toujours obligé de traiter numériquement au moins une matrice de taille  $d$  pour calculer les dérivées réelles des fonctions  $N_i$ .

**Dérivée première :** Commençons par calculer le second membre de l'équation  $\mathcal{A}(x)M'(x) = -\mathcal{A}'M(x)$ , en remarquant que  $P'^T(x)N(x) = -e_2$  :

$$\mathcal{A}'(x)M(x) = \left\{ \left\{ \frac{w_i(x)'}{w_i(x)} n_i(x) - \sum_{k=1}^d \lambda_k(x) k (x_i - x)^{k-1} \right\}_i \right\} - e_2$$

En prenant compte de la décomposition de  $\mathcal{A}$ , le système de calcul du vecteur  $M'$  prend la forme<sup>1</sup> :

$$\begin{bmatrix} I & 0 \\ 0 & R^T \end{bmatrix} \begin{bmatrix} V^{-1} & \eta \\ 0 & -R \end{bmatrix} \begin{Bmatrix} N' \\ \mu \end{Bmatrix} = \begin{Bmatrix} \left\{ \sqrt{w_i} \chi_i - \frac{w'_i}{\sqrt{w_i}} n_i \right\}_{i \in I(x)} \\ \{\zeta_k\}_{k \in [0, d]} \end{Bmatrix} \quad (\text{A.3})$$

avec  $\zeta_k(x) = \sum_{i=1}^n w_i(x)(x_i - x)^k \chi_i(x) - \sum_{i=1}^n w'_i(x)(x_i - x)^k n_i(x) + \delta_k^2$  et  $\delta_i^j$  le symbole de Kronecker. Le vecteur  $\mu$  est alors solution du système  $A(x)\mu(x) = -\zeta(x)$ , et il vient d'après le système précédent et après quelques lignes de calcul :

$$N'_i(x) = w_i(x)\chi_i(x) - w'_i(x)n_i(x) - w_i(x) \sum_{k=0}^d \mu_k(x)(x_i - x)^k \quad (\text{A.4})$$

Cette forme est bien cohérente avec l'expression (A.2), et l'intérêt principal de ce développement est d'avoir explicité le second membre du système de calcul de  $\mu$ . De plus, cela prouve que<sup>2</sup>

$$\begin{aligned} n'_i(x) &= \sum_{k=0}^d \mu_k(x)(x_i - x)^k - \chi_i(x) \\ &= \sum_{k=0}^d \mu_k(x)(x_i - x)^k - \sum_{k=1}^d \lambda_k(x) k (x_i - x)^{k-1} \end{aligned}$$

**Dérivée seconde :** La dérivation de l'expression de  $N'_i$  donne

$$N''_i(x) = -w''_i n_i(x) - 2w'_i(x)n'_i(x) - w_i(x)n''_i(x) \quad (\text{A.5})$$

et la dérivée seconde de  $n_i$  s'écrit

$$n''_i(x) = \sum_{k=0}^n \nu_k(x)(x_i - x)^k - 2 \sum_{k=1}^n \mu_k(x) k (x_i - x)^{k-1} + \sum_{k=2}^n \lambda_k(x) k(k-1)(x_i - x)^{k-2}$$

avec  $\nu_k(x)$  la dérivée de  $\mu_k(x)$ . La difficulté principale est donc le calcul de la dérivée  $\nu_k(x)$ , comme l'était celui de  $\mu_k(x)$  pour la dérivée première.

Or

$$\mathcal{A}''(x)M(x) = \left\{ \left\{ \frac{w''_i(x)}{w_i(x)} n_i(x) - 2 \frac{w'_i{}^2(x)}{w_i^2(x)} n_i(x) \right\}_i + P''(x)\lambda \right\} - 2e_3$$

et

$$\mathcal{A}'(x)M'(x) = \left\{ \left\{ -\frac{w'_i}{w_i} \chi_i + \frac{w''_i}{w_i^2} n_i + \frac{w'_i}{w_i} \sum_{k=0}^d \mu_k(x)(x_i - x)^k \right\}_i + P'\mu \right\} - 2e_3$$

1. On oublie la notation fonctionnelle  $(x)$  pour alléger les expressions  
2. Ce que confirme le calcul direct !

d'où on tire l'expression du second membre du système dérivée seconde:

$$\mathcal{A}''(x)M(x) + 2\mathcal{A}'(x)M'(x) = \left\{ \begin{array}{l} \left\{ \frac{w_i''(x)}{w_i(x)}n_i(x) + 2\frac{w_i'(x)}{w_i(x)}n_i'(x) \right\}_i + P''(x)\lambda + 2P'(x)\mu \\ -2e_3 \end{array} \right\}$$

Ceci nous invite à définir les fonctions

$$\left\{ \begin{array}{l} \varphi_i(x) = \sum_{k=2}^d \lambda_k(x)k(k-1)(x_i-x)^{k-2} \\ \psi_i(x) = \sum_{k=1}^d \mu_k(x)k(x_i-x)^{k-1} \end{array} \right.$$

et l'exploitation de la décomposition de  $\mathcal{A}(x)$  nous mène au second membre du système de calcul de  $\nu$ , soit, si

$$\begin{aligned} \xi_k(x) &= \sum_{i=1}^n w_i''(x_i-x)^k n_i(x) + 2 \sum_{i=1}^n w_i'(x)(x_i-x)^k n_i'(x) \\ &\quad + \sum_{i=1}^n w_i(x)(x_i-x)^k (\varphi_i(x) - 2\psi_i(x)) + 2\delta_k^3 \end{aligned}$$

alors le vecteur  $\nu(x)$  est solution du système  $A(x)\nu(x) = -\xi(x)$ . Et ceci permet de calculer les fonctions  $N_i''$ .

**Mise en œuvre :** Une fonction de calcul des fonctions de forme EFG<sup>3</sup> et de leur dérivée première ou seconde suit donc les étapes suivantes :

1. Calculer  $A = P^\top W P$  et sa décomposition Cholesky.
2. Calculer  $\lambda = -A^{-1}e_1$ .
3. Calculer  $n$  (algorithme de Horner).
4. Si on veut les  $N_i$ ,  $N = -Wn$ , FIN.
5. Calculer  $\mu$ .
  - i. Stocker le vecteur des  $\chi_i$  dans un tableau temporaire (Horner).
  - ii. Construire le vecteur  $\zeta$  (Horner) et calculer  $\mu = -A^{-1}\zeta$ .
6. Calculer  $n'$  (Horner).
7. Si on veut les  $N_i'$ ,  $N' = -W'n - Wn'$ , FIN.
8. Calculer  $\nu$ .
  - i. Stocker le vecteur des  $\varphi_i - 2\psi_i$  dans un tableau temporaire (Horner).
  - ii. Construire le vecteur  $\xi$  (Horner) et calculer  $\nu = -A^{-1}\xi$ .
9. Calculer  $n''$  (Horner).
10. Calculer  $N'' = W''n + 2W'n' + Wn''$ , FIN.

Les besoins en mémoire d'un tel algorithme est :  $d^2$  pour  $A$  (sa décomposition Cholesky est stockée dans  $A$ ),  $3d$  pour  $\lambda$ ,  $\mu$  et  $\nu$ ,  $m(=\text{Card}(I(x)))$  de tableau temporaire et  $3m$  pour  $n$ ,  $n'$  et  $n''$ , soit finalement  $d(d+3) + 4m$ .

Nous avons mentionné l'emploi de l'algorithme de Horner pour le calcul des polynômes afin de montrer combien le calcul des dérivées réelles des fonctions de forme MLS est coûteux. Les résolutions de systèmes d'équations linéaires ( $\lambda$ ,  $\mu$  et  $\nu$ ) utilisent toutes la décomposition de  $A$  calculée en 1. Bien entendu, l'algorithme de Breitkopf est utilisé pour calculer les dérivées diffuses des fonctions de forme.

---

3. On dérive réellement.

## A.2 Algorithme de Breitkopf

La caractérisation des fonctions de forme ou de leurs pseudo-dérivées avec consistance polynomiale mène à un système de la forme

$$\begin{bmatrix} W^{-1} & P_1 & \cdots & P_j & \cdots & P_k \\ P_1^\top & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & & & \vdots \\ P_j^\top & 0 & & \ddots & & 0 \\ \vdots & \vdots & & & \ddots & \vdots \\ P_k^\top & 0 & \cdots & 0 & \cdots & 0 \end{bmatrix} \begin{pmatrix} N \\ mu_1 \\ \vdots \\ \mu_j \\ \vdots \\ \mu_k \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix}.$$

C'est la base des deux algorithmes que nous allons décrire. Le principe de l'algorithme de Breitkopf est d'éliminer les multiplicateurs de Lagrange  $\mu_i$  un à un en finissant par  $\mu_j$ .

Supposons que  $j = 1$  (on peut toujours se ramener à ce cas par inversion de ligne). La première équation

$$N = -W \left( \sum_{i=1}^{k-1} \mu_i P_i + \mu_k P_k \right)$$

injectée dans la dernière nous permet d'éliminer les multiplicateurs de Lagrange  $\mu_k$  (en se rappelant que  $P_k^\top W P_k$  est un scalaire)

$$\mu_k = -\frac{P_k^\top W}{P_k^\top W P_k} \sum_{i=1}^{k-1} \mu_i P_i$$

soit pour  $N$

$$N = -W \left( \sum_{i=1}^{k-1} \mu_i \left( I - \frac{P_k^\top W}{P_k^\top W P_k} \right) P_i \right).$$

Le système de caractérisation de  $N$  devient alors

$$\begin{bmatrix} \widetilde{W}_k^{-1} & P_1 & \cdots & P_{k-1} \\ P_1^\top & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ P_{k-1}^\top & 0 & \cdots & 0 \end{bmatrix} \begin{pmatrix} N \\ \mu_1 \\ \vdots \\ \mu_{k-1} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

où

$$\widetilde{W}_k = W \left( I - \frac{P_k P_k^\top W}{P_k^\top W P_k} \right)$$

De la même façon, nous pouvons éliminer les multiplicateurs de Lagrange jusqu'à  $P_1$  grâce à la suite finie de matrice  $\widetilde{W}_i$  définie par l'algorithme suivant

1. soit  $\widetilde{W}_k$  définie comme précédemment
2. pour tout  $i$  dans  $\llbracket 2, k-1 \rrbracket$ ,
  - i. calculer  $\eta_i = \widetilde{W}_{i+1} P_i$
  - ii.  $\widetilde{W}_i = \widetilde{W}_{i+1} - \frac{\eta_i \eta_i^\top}{P_i^\top \eta_i}$
3.  $\widetilde{W}_1 = \frac{1}{P_1^\top \widetilde{W}_2 P_1} \widetilde{W}_2$

et les fonctions de forme sont données par

$$N = \widetilde{W}_1 P_1$$

### A.3 Complexités

Nous avons compté les multiplications nécessaires pour calculer les fonctions de forme avec chacun des deux algorithmes proposés. Nous supposons que les matrices  $W$  et  $P$  sont calculées, avec  $n$  nœuds d'influence et un degré  $d$  en 1d. Le nombre de multiplication est :

$$\mathbf{Breitkopf} \quad : \quad n(4n + 4 + (d - 2)(n + 1))$$

$$\mathbf{Algorithme 1d} \quad : \quad n(d^2 + 2d + 1) + \frac{d(2d^2 + 3d - 1)}{6}$$

Ainsi le facteur dominant dans les deux calculs sont différents. Pour le premier algorithme, les termes dominants sont en  $n^2$  et  $n^2d$ . Pour le second algorithme, on a  $nd^2$  et  $d^3$ . Cependant,  $n$  et  $d$  ont de faibles plages de variations (surtout  $d \dots$ ), ce qui ne permet pas a priori de choisir entre les deux algorithmes. En pratique, nous utilisons l'algorithme de Breitkopf pour les problèmes d'approximation car on peut le rendre robuste. L'algorithme spécifique 1d est utilisé dans la librairie poutre car l'expérience a montré qu'il est le plus rapide dans ce cas où l'on calcule les dérivées réelles de l'approximation Hermitienne. Notons pour finir que le calcul de l'interpolation diffuse demande aussi des techniques spécifiques et/ou la modification des algorithmes précédents.

## Annexe B

# Construction de la Topo-géométrie

L'approximation diffuse repose sur la définition de la fonction  $w(\cdot, \cdot)$  et des propriétés importantes (continuité, supports des fonctions de forme) sont liées à cette fonction. On peut considérer que la construction de  $w$  est équivalente à la construction d'un maillage dans le cas de l'approximation par sous-domaines. On distingue deux grandes familles :

- L'une basée sur un calcul des plus proches voisins,
- L'autre sur la notion de zone d'influence.

La première technique est efficace, rapide et bien adaptée au problème d'approximation, mais au prix de mauvaises propriétés (supports de fonctions de forme complexes, continuité). La seconde technique permet de construire des approximations avec de bonnes propriétés, mais elle est plus délicate à automatiser.

La construction de  $w$  est relativement peu abordée dans la littérature : on trouve des références sur la première technique dans les articles relatifs à la collocation [LO80, LDT96]; mais seuls [DO96, Guy95b] proposent des constructions dans le second cas. Nous nous intéressons ici aux fonctions d'influence de la forme  $w_i(x)$ , car elles sont utilisées pour les équations différentielles. Notons que cette forme pose des problèmes géométriques assez difficiles comme le calcul de la liste d'influence  $I(x)$  en un point donné lorsqu'on doit gérer de nombreux nœuds. Cet aspect n'est pas abordé ici ; dans la librairie DIAM il est abordé en quadrillant l'espace par une grille.

Les notations sont celles du chapitre 2.

## B.1 Aspects généraux

### B.1.1 Construction par croissance

La construction des fonctions d'influence  $w_i$  dans  $\mathbb{R}^d$  doit respecter l'existence de l'approximation diffuse et sa continuité. En pratique, on se contente d'une condition sur le cardinal de  $I(x)$  :

$$\forall x, \text{Card}(I(x)) > n \tag{B.1}$$

où  $n$  est le nombre de fonctions de base utilisées pour la définition de  $\bar{u}$ . On vérifie *a posteriori* l'existence de l'approximation (i.e. que le système linéaire  $A(x)a(x) = b(x)$  est bien inversible).

Par ailleurs, si l'on connaît des fonctions d'influence  $w_i$  et les zones d'influences associées  $V_i$  qui vérifient la condition B.1, alors toute famille de fonctions d'influence  $w'_i$  telles que les zones d'influence associées  $V'_i$  vérifient

$$\forall i, V_i \subset V'_i$$

respecte aussi B.1. Ceci permet de passer d'une topo-géométrie à une autre *par croissance*. L'objectif est donc de trouver des fonctions d'influence  $w_i$  qui soient « les plus petites possibles ». On trouve dans [Guy95b] une technique qui s'appuie sur cette idée et construit des fonctions poids à partir des pavés de Voronoï d'ordre 1.

### B.1.2 Voisinage asymétrique

Il est d'usage de définir les fonctions d'influence à partir d'une cloche de référence  $w_{ref}$  de support  $] -1, 1[$  et d'associer à chaque nœud un rayon d'influence  $r_i$  pour définir  $w_i$  par

$$w_i(x) = w_{ref} \left( \frac{\|x - x_i\|}{r_i} \right)$$

Le choix de la distance  $\|x - x_i\|$  permet de construire des zones d'influence de forme différente :

- sphère (disque en 2d) avec la norme  $\|x\| = \sqrt{\sum_{j=1}^d x_j^2}$ ,
- cube (carré en 2d) avec la norme infinie

Le produit tensoriel est aussi couramment utilisé sous la forme

$$w_i(x) = \prod_{j=1}^d w_{ref} \left( \frac{x^j - x_i^j}{r_i^j} \right)$$

qui permet de définir un rayon d'influence par dimension.

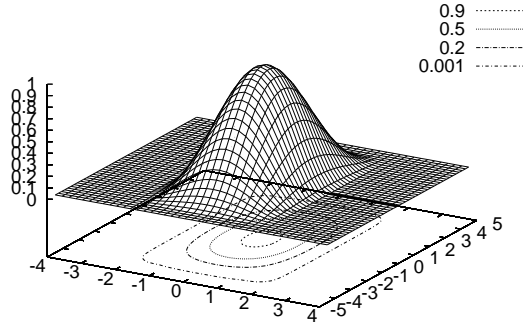


FIG. B.1: Exemple d'une fonction d'influence avec domaine d'influence rectangulaire :  $w_i$  est définie par produit tensoriel.

Pour aller plus loin, on conserve le produit tensoriel, mais on remarque que le terme  $\frac{x^j - x_i^j}{r_i^j}$  est une bijection de  $[x_i^j - r_i^j, x_i^j + r_i^j]$  sur  $[-1, 1]$ . Nous définissons donc

$$w_i(x) = \prod_{j=1}^d w_{ref} \left( \eta_i^j(x^j) \right)$$

où la fonction  $\eta_i^j$  est une bijection d'un segment  $[y_i^j, z_i^j]$  contenant  $x_i^j$  sur  $[1, -1]$ .

En 1d, on définit un voisinage a priori  $[x_i^w, x_i^e]$  qui contient  $x_i$ , on construit la fonction  $\eta_i$  (comme un polynôme ou une homographie en fonction du rapport  $\frac{x_i^w - x_i}{x_i^e - x_i}$ ) comme le montre la figure B.2. Cette construction permet de construire des fonctions d'influence adaptées au support d'intégration et aussi régulières que l'on souhaite (voir le chapitre sur l'intégration). En 1d, on obtient aussi des fonctions d'influence « minimales » comme le montre la section suivante.

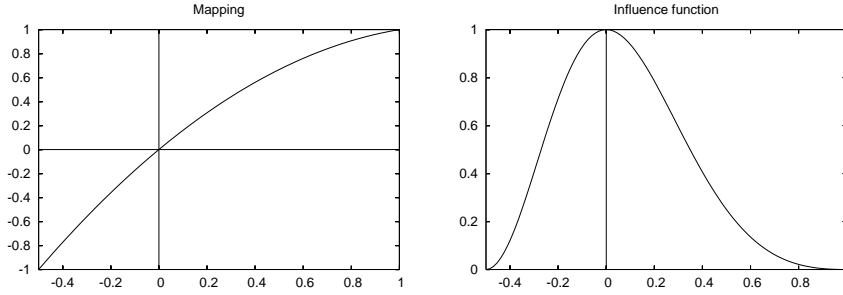


FIG. B.2: *Bijection et fonction d'influence asymétrique.*

## B.2 Cas 1d : voisinage de type Voronoï

### B.2.1 Consistance linéaire

Nous considérons un ensemble de nœuds  $x_0, \dots, x_n$  ordonnés par ordre croissant et leurs milieux  $y_i = \frac{x_i + x_{i+1}}{2}$ . On cherche des fonctions poids qui assurent l'existence de l'approximation diffuse dans  $[x_0, x_n]$ . Nous utilisons la technique de Guyon : les segments  $[y_{i-1}, y_i]$  sont les pavés de Voronoï d'ordre 1 et on demande 3 nœuds d'influence partout. Choisissons  $[y_{i-2}, y_{i+1}]$  (figure B.3) comme voisinage de  $x_i$ . Les voisinages de  $x_0, x_1, x_{n-1}, x_n$  sont calculés en ajoutant des nœuds virtuels  $x_{-2}, x_{-1}, x_{n+1}, x_{n+2}$ .

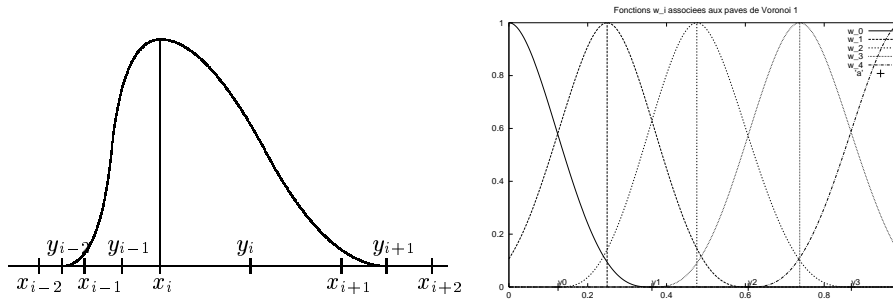


FIG. B.3: *Définition du voisinage de type Voronoï d'ordre 1.*

Ainsi, nous avons les listes d'influence suivantes :

- si  $x$  appartient à  $]y_{i-1}, y_i[$ , alors  $I(x) = \{i - 1, i, i + 1\}$ .
- $I(y_i) = \{i, i + 1\}$ .



La condition  $I(x) > 2$  est vérifiée presque partout, et on a  $I(X) \geq 2$  partout ce qui assure l'existence de l'approximation diffuse linéaire. Par ailleurs, les influences des nœuds  $i-1$  et  $i+2$  s'éteignent continûment au point  $y_i$  et la continuité y est assurée. Nous avons donc construit des fonctions d'influence qui assurent l'existence et la continuité de l'approximation diffuse avec un nombre *minimal* de nœuds d'influence par points  $x$ . Cette propriété est utile pour limiter la largeur de bande des matrices de raideur.

### B.2.2 Cas général

La construction précédente s'étend à des consistances supérieures  $k$  (i.e. pour  $1, \dots, x^k$ ) en distinguant les consistances paires et impaires.

**Construction pour une consistance paire  $2m$**  Le voisinage du nœud  $x_i$  est  $]x_{i-m-1}, x_{i+m+1}[$ . Les listes d'influence sont données par

- si  $x$  appartient à  $]x_i, x_{i+1}[$ ,  $I(x) = \{i - m, \dots, i, \dots, i + m + 1\}$ ,
- $I(x_i) = \{i - m, \dots, i, \dots, i + m\}$ .

Notons que cette construction assure l'interpolation sans modification des fonctions d'influence tout en conservant les propriétés de continuité comme dans le cas précédent.

**Construction pour une consistance impaire  $2m + 1$**  Le voisinage du nœud  $x_i$  est  $]y_{i-m-2}, y_{i+m+1}[$ . Les listes d'influence sont données par

- si  $x$  appartient à  $]y_i, y_{i+1}[$ ,  $I(x) = \{i - m, \dots, i + m + 2\}$ ,
- $I(y_i) = \{i - m, \dots, i + m + 1\}$

**Frontières** La construction pose quelques difficultés aux frontières. Les fonctions d'influence des nœuds  $x_0, \dots, x_k$  sont modifiées pour être constantes (égale à 1) à gauche et la même transformation est effectuée à droite pour les nœuds  $x_{n-k}, x_n$ . Ainsi l'approximation diffuse devient continûment l'interpolation Lagrangienne à la frontière. Ceci simplifie les raccords (voir soudure dans le chapitre relatif aux poutre) et la prise en compte des conditions aux limites.

## Annexe C

# Surfaces utilisées dans les tests

Voici la liste des surfaces utilisées dans la troisième partie. Pour chaque surface, nous donnons une ou plusieurs vues en rendu réaliste. Ces surfaces peuvent être obtenues en contactant O. Stab ([stab@cges.ensmp.fr](mailto:stab@cges.ensmp.fr)).

### C.1 Modèle de vache

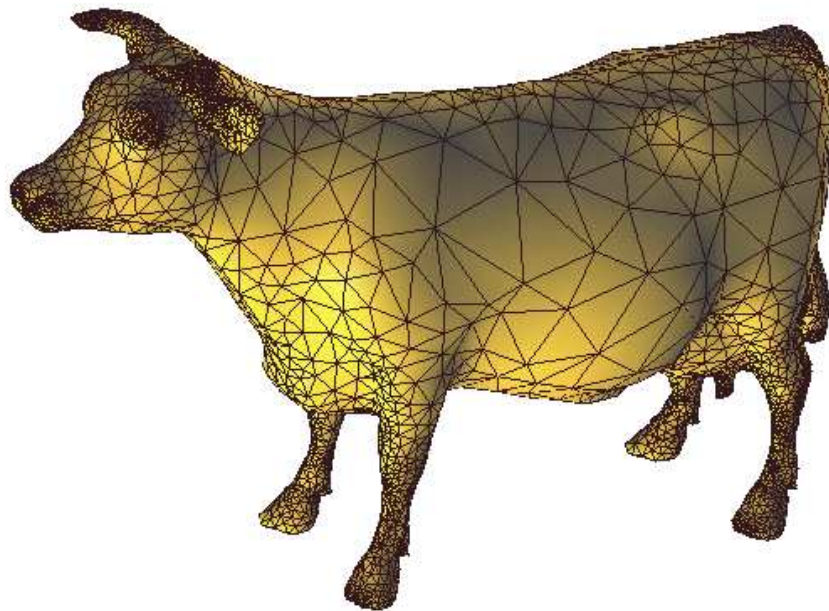


FIG. C.1: *Modèle de vache*

## C.2 Cylindres

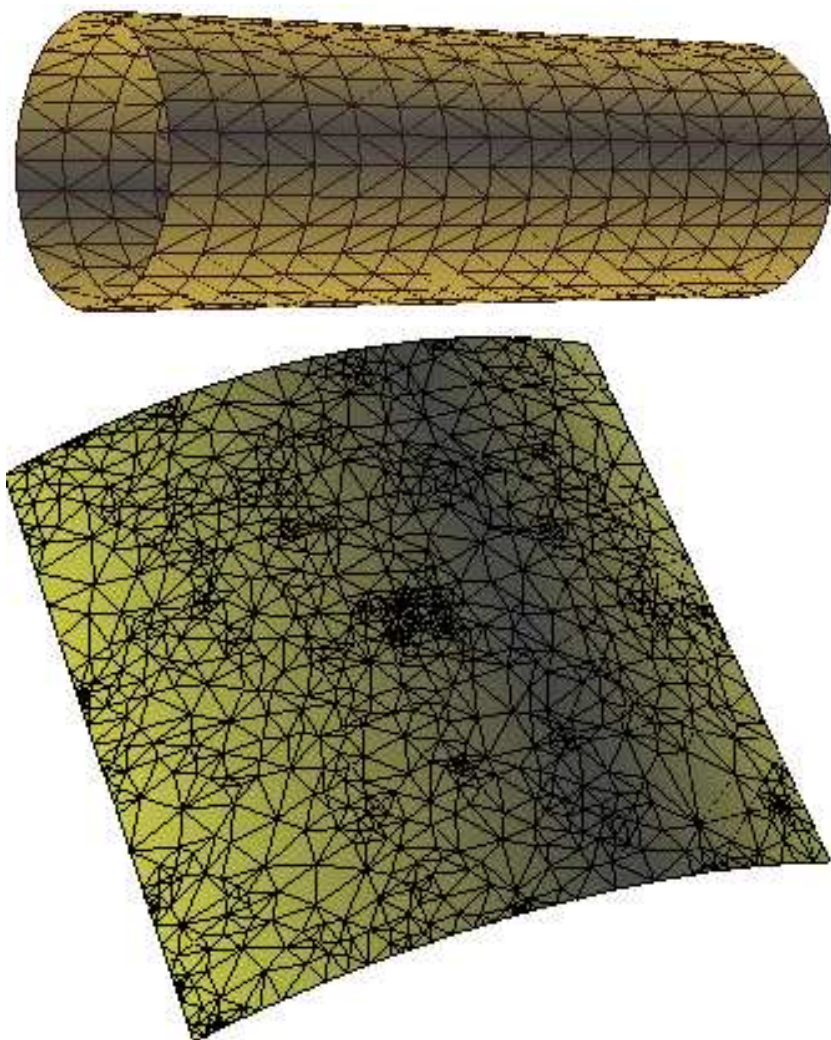


FIG. C.2: Modèles de cylindre avec des maillages réguliers et irréguliers.

### C.3 Surface PSA

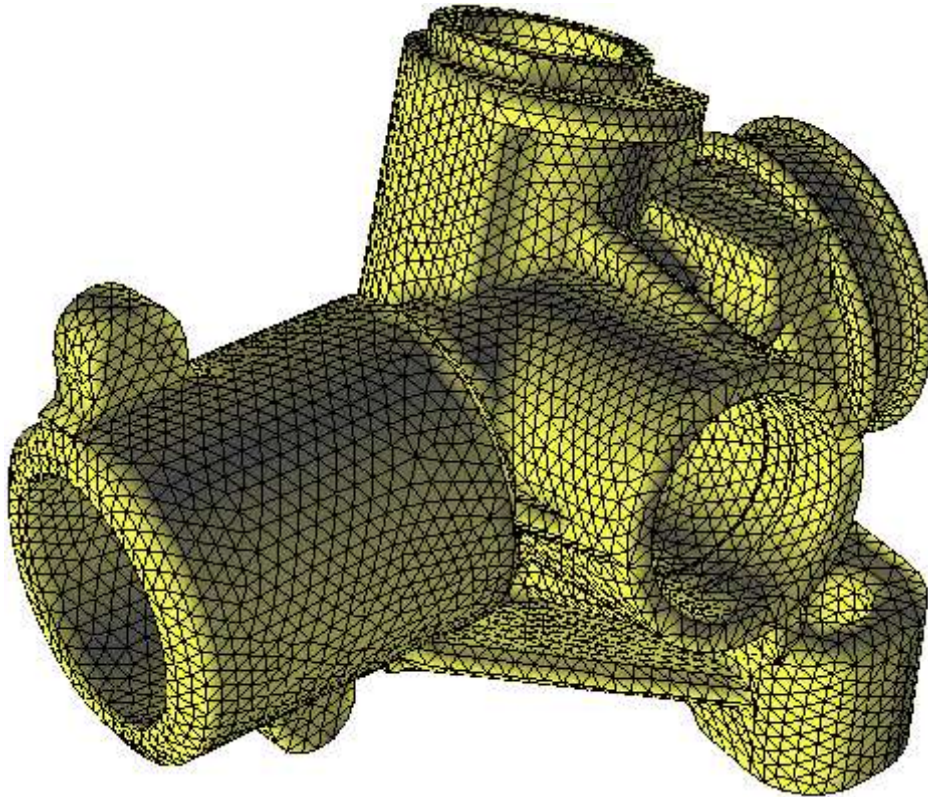


FIG. C.3: *Surface PSA (pièce de boîte de vitesses).*

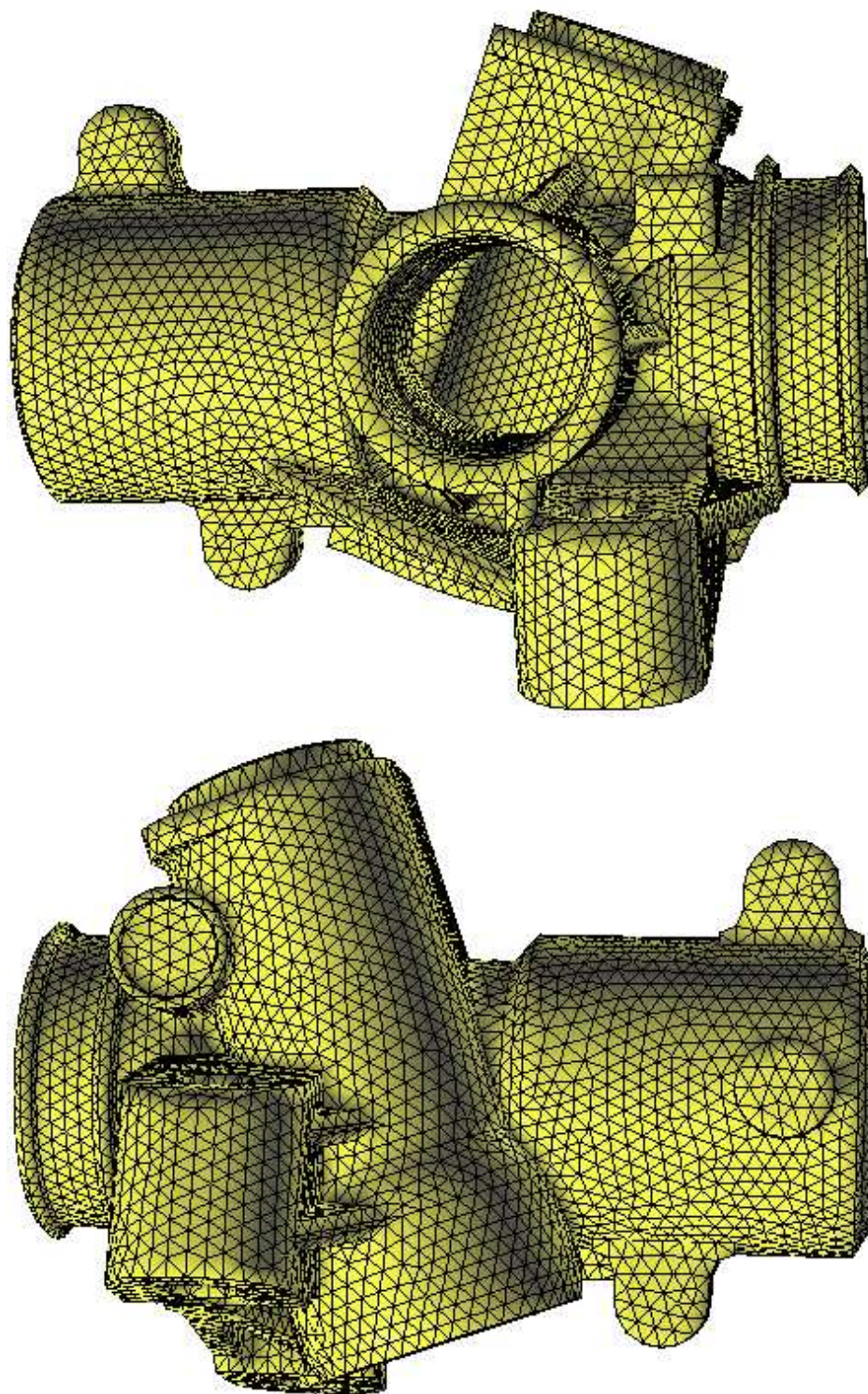


FIG. C.4: Surface PSA : vue de chaque côtés de la pièce.

## C.4 Surface Sony

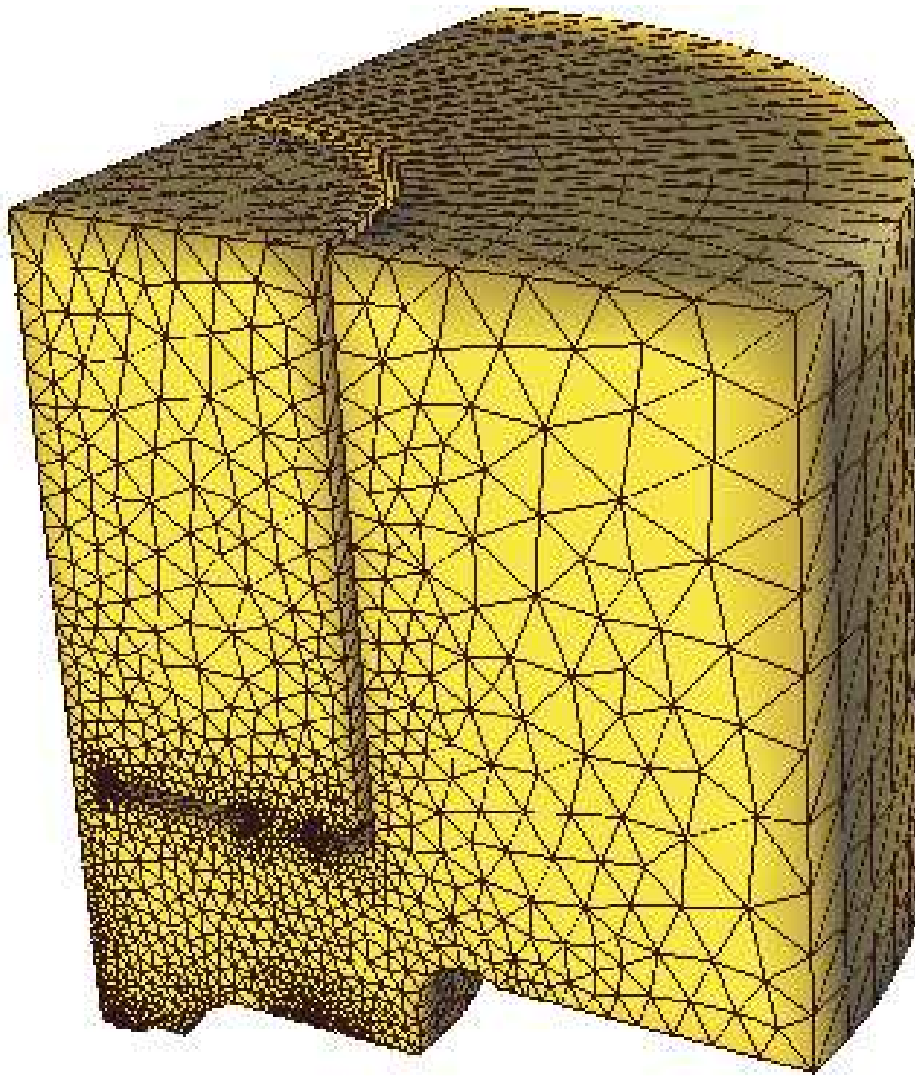


FIG. C.5: *Surface Sony (rouleau de magnétoscope).*

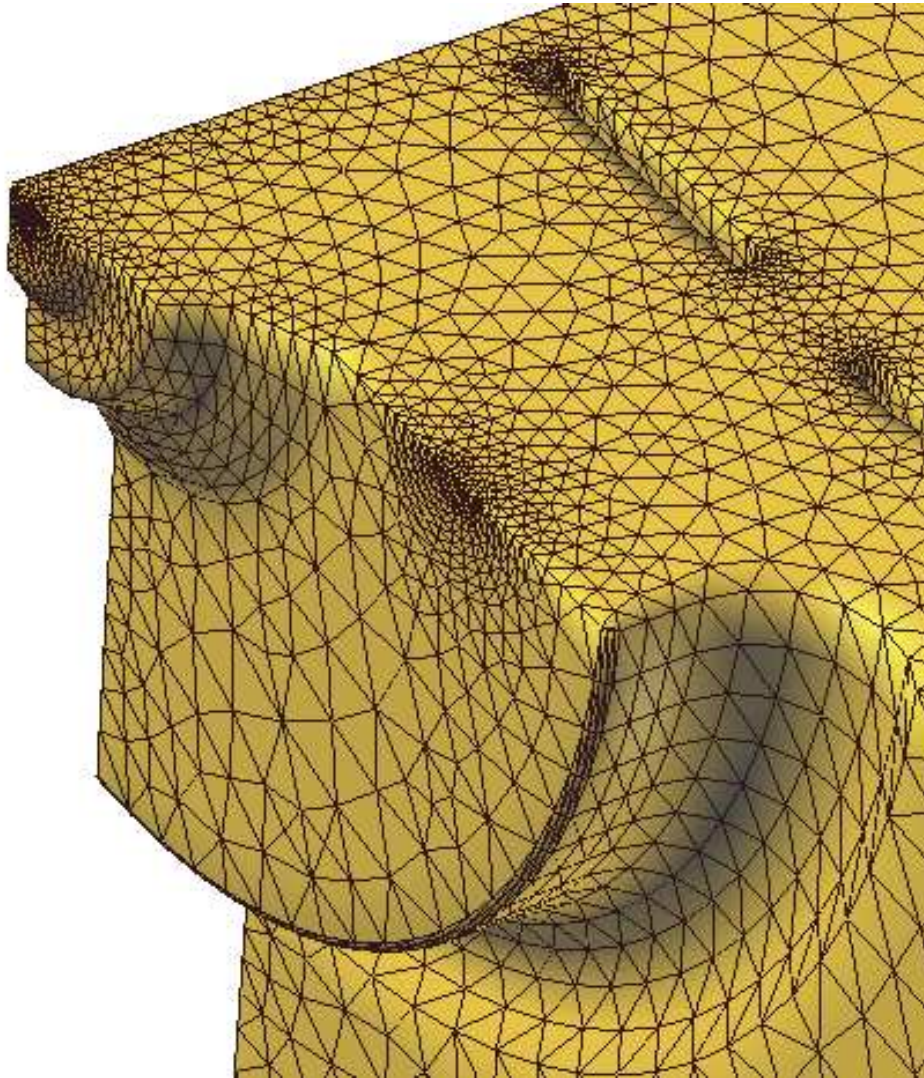


FIG. C.6: *Surface Sony (détail).*

## C.5 Surface French

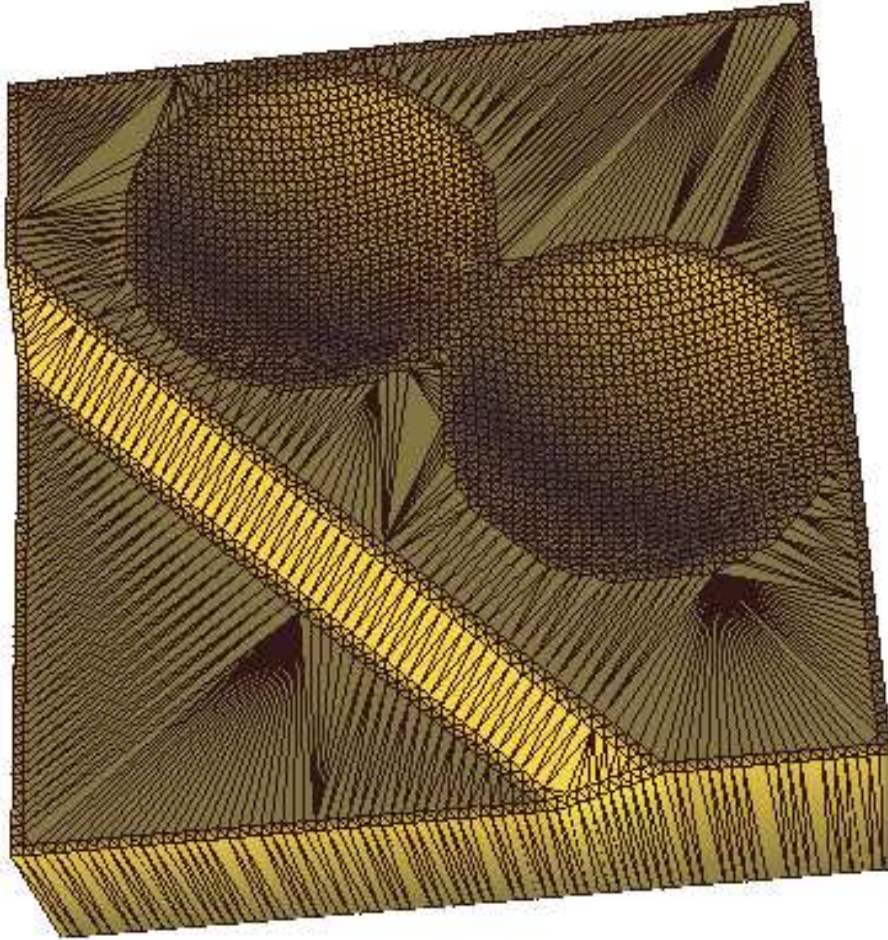


FIG. C.7: *Surface French.*



## C.6 Surface Benz

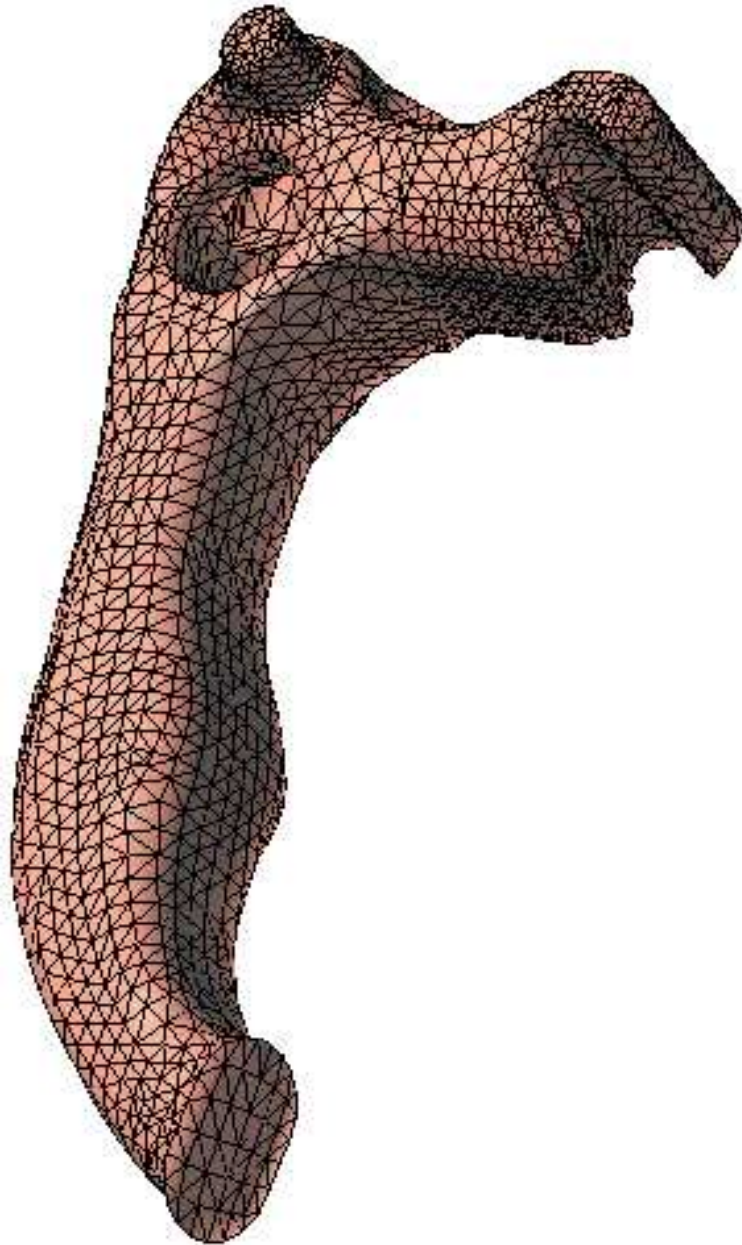


FIG. C.8: *Surface Benz.*

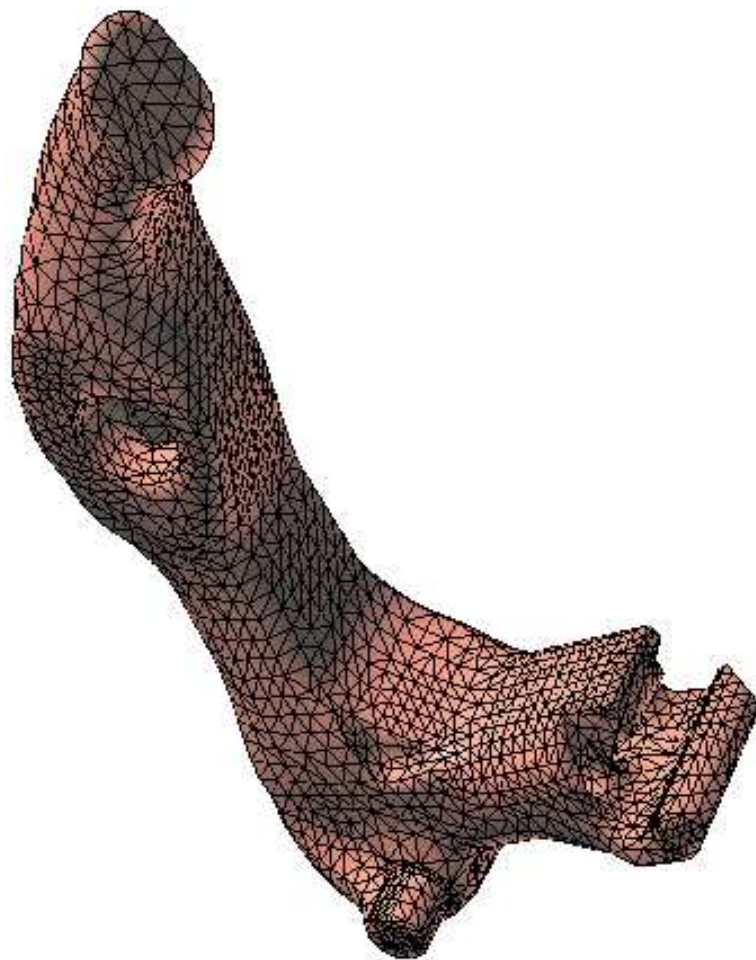


FIG. C.9: *Surface Benz.*



## Annexe D

# Convergence de l'approximation Hermite

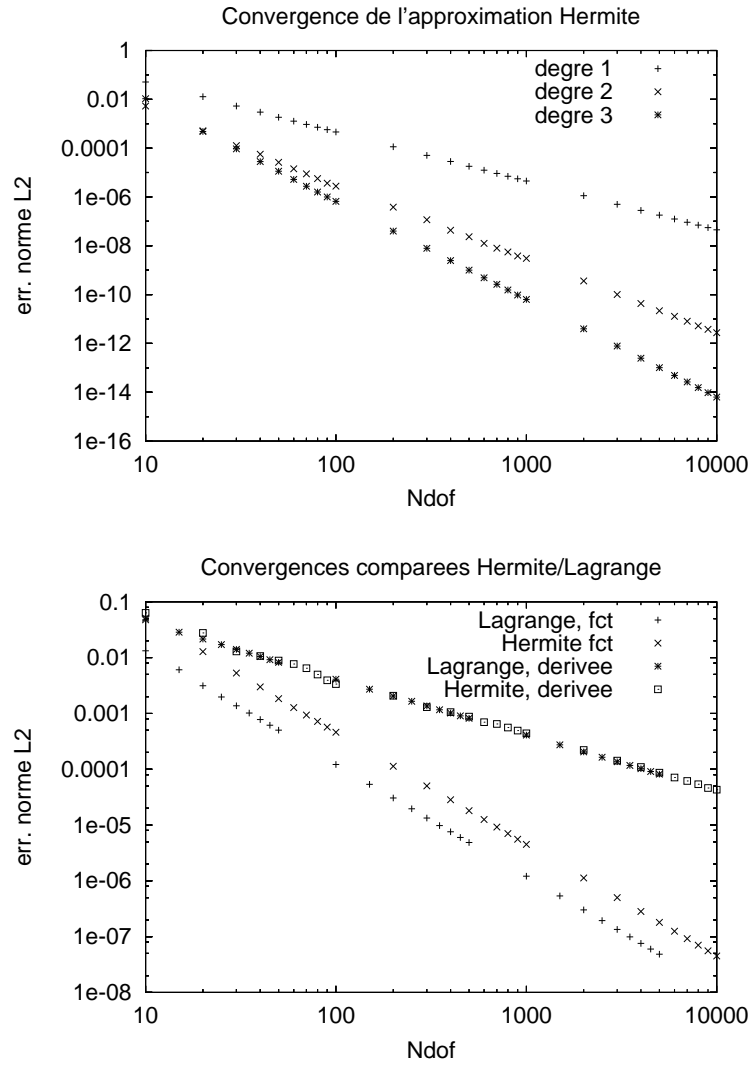


FIG. D.1: Convergence de l'approximation , fonction test 1.

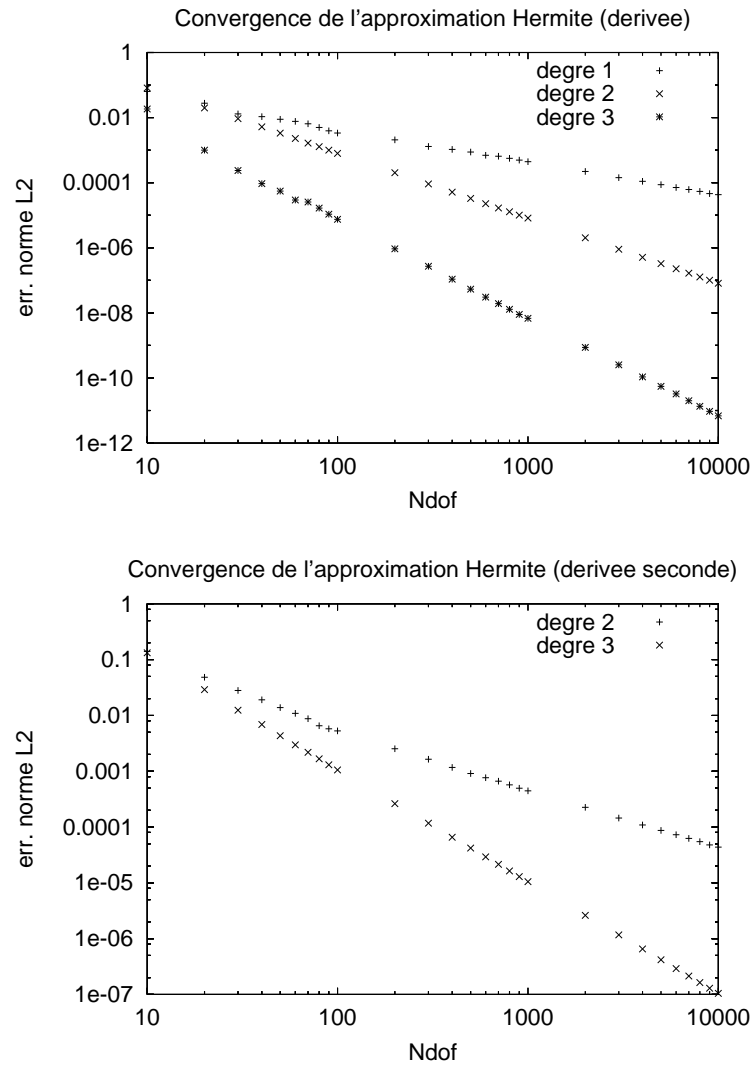


FIG. D.2: *Convergence de l'approximation , fonction test 1.*

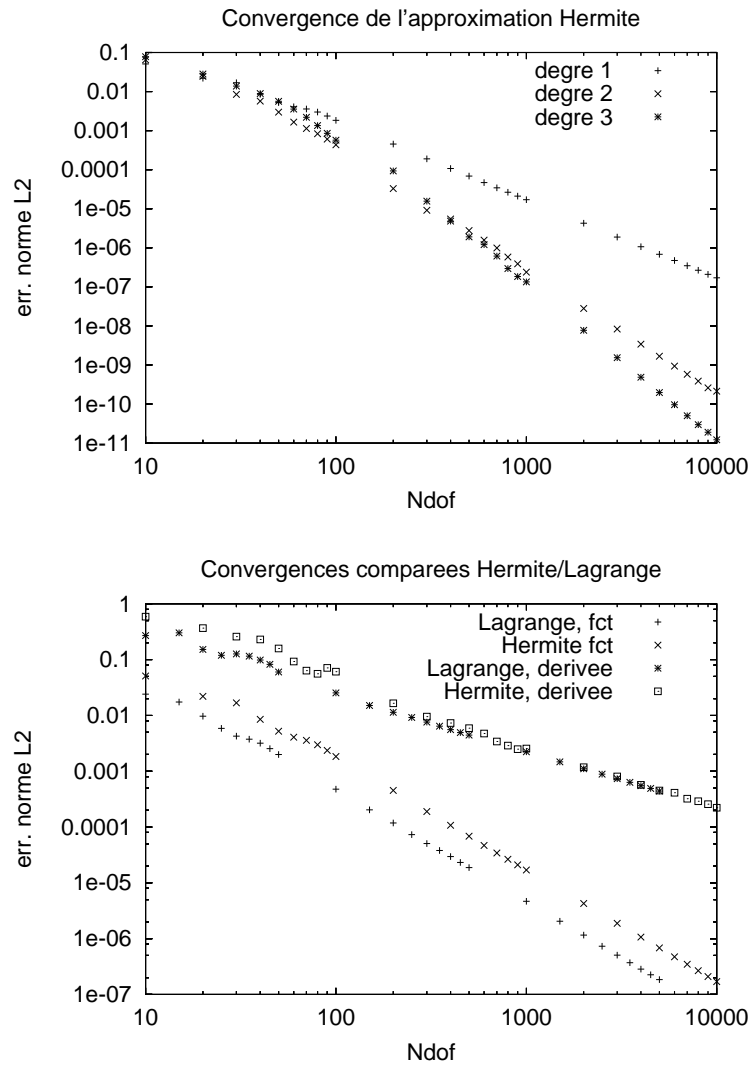


FIG. D.3: Convergence de l'approximation , fonction test 2.

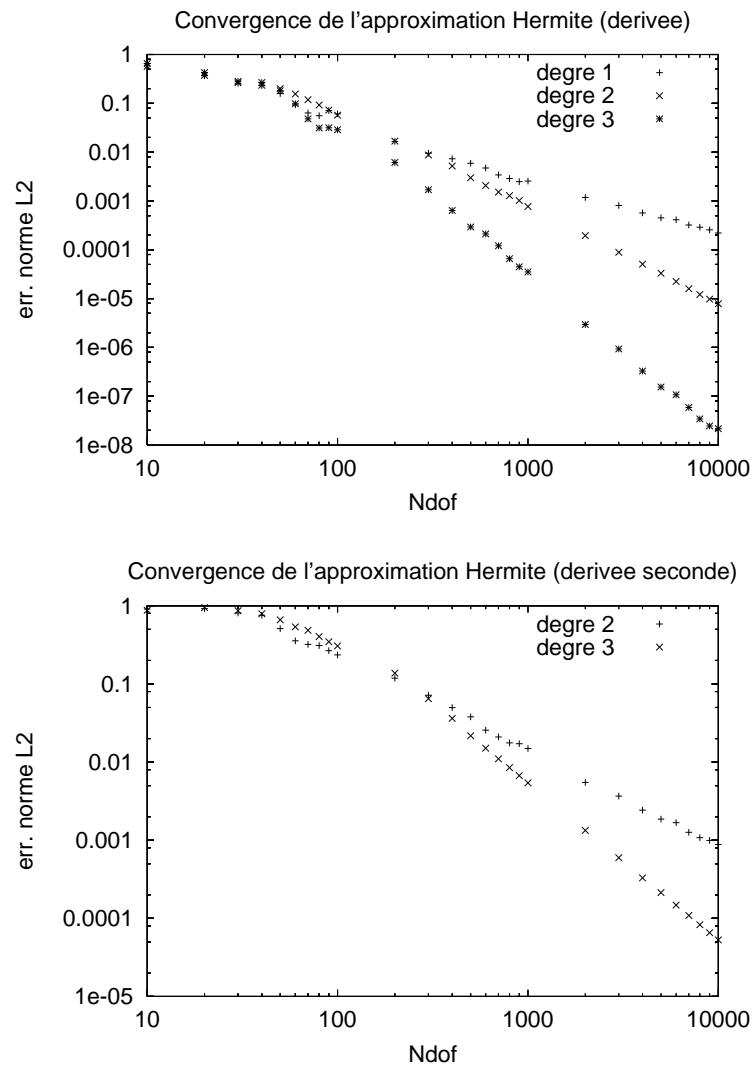


FIG. D.4: *Convergence de l'approximation , fonction test 2.*



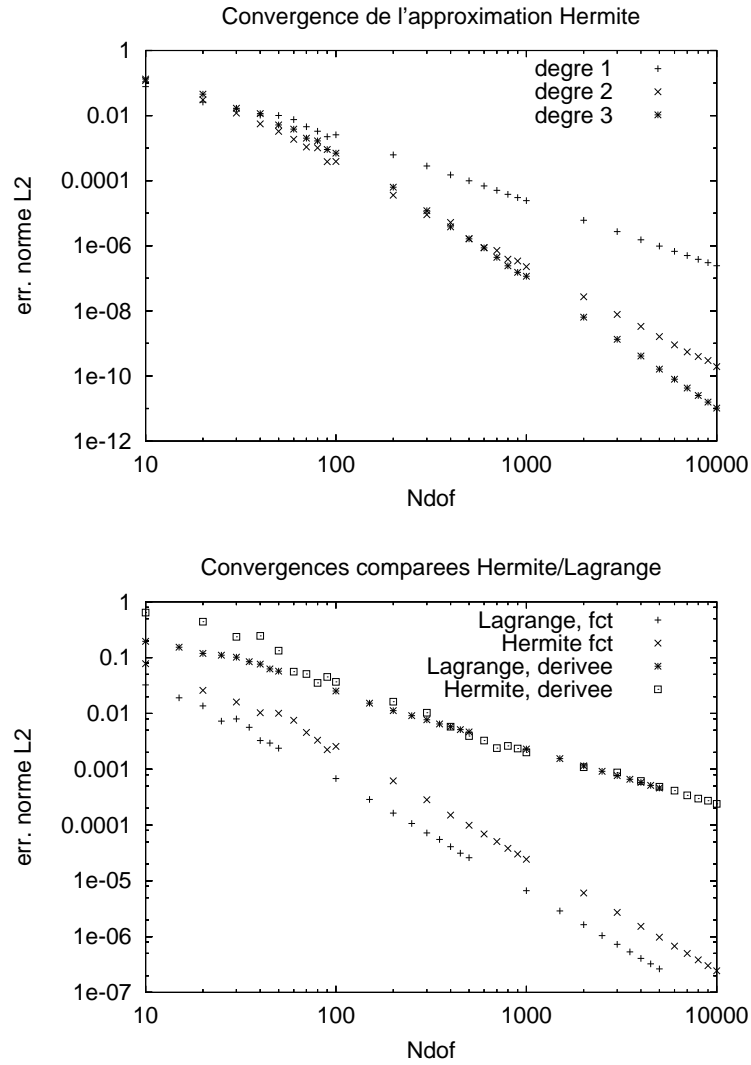


FIG. D.5: Convergence de l'approximation , fonction test 3.

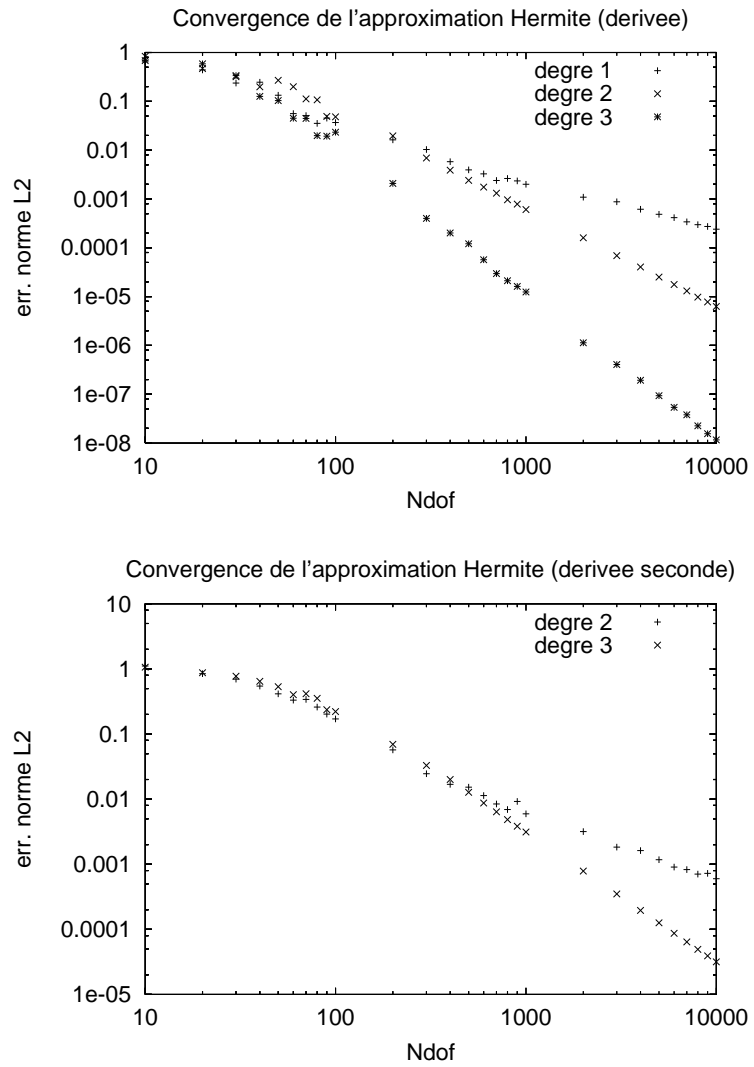


FIG. D.6: *Convergence de l'approximation , fonction test 3.*

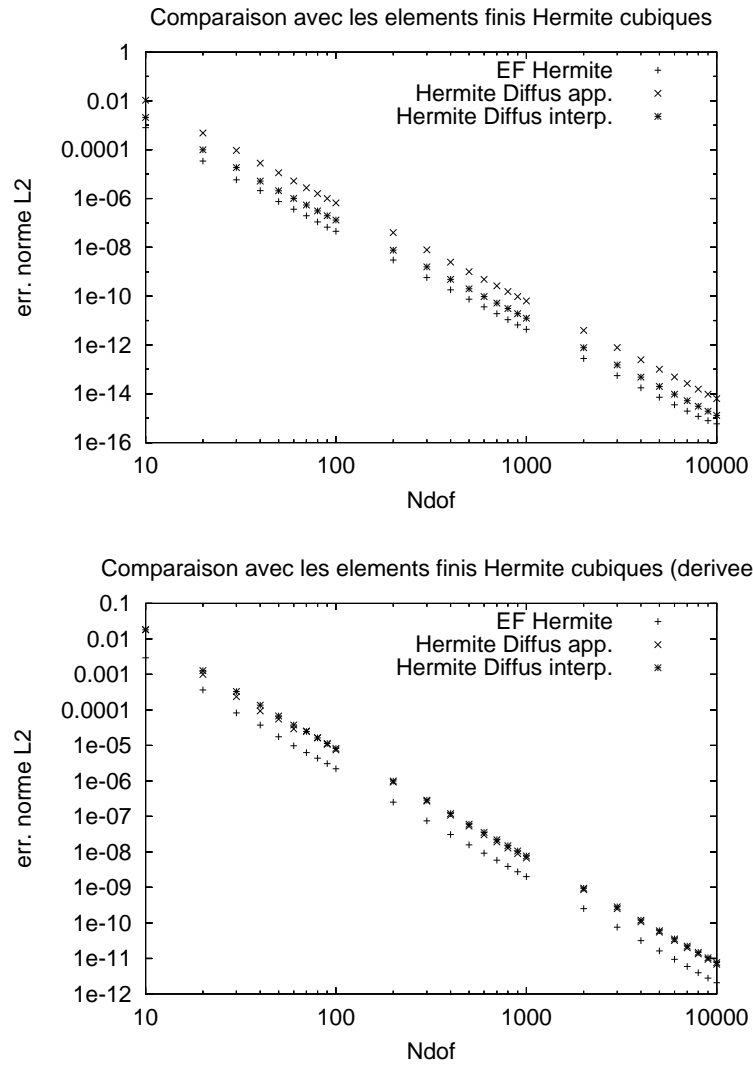


FIG. D.7: *Approximation diffuse Hermite cubique et Hermite cubique fonction test 1.*

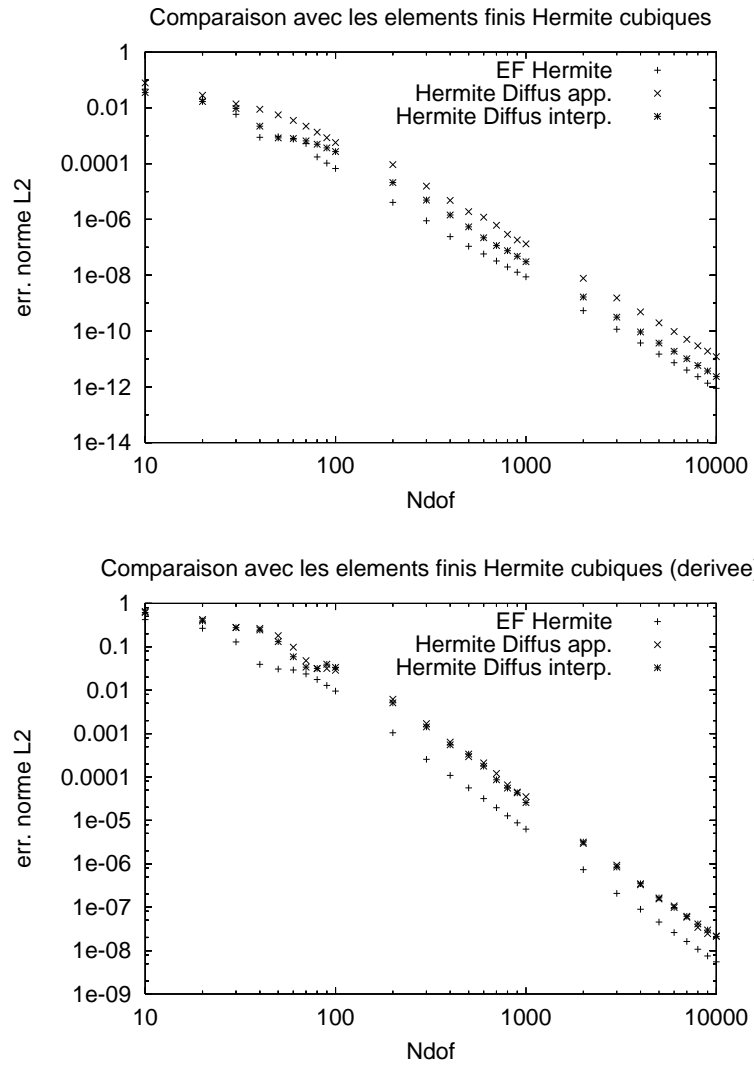


FIG. D.8: *Approximation diffuse Hermite cubique et Hermite cubique fonction test 2.*

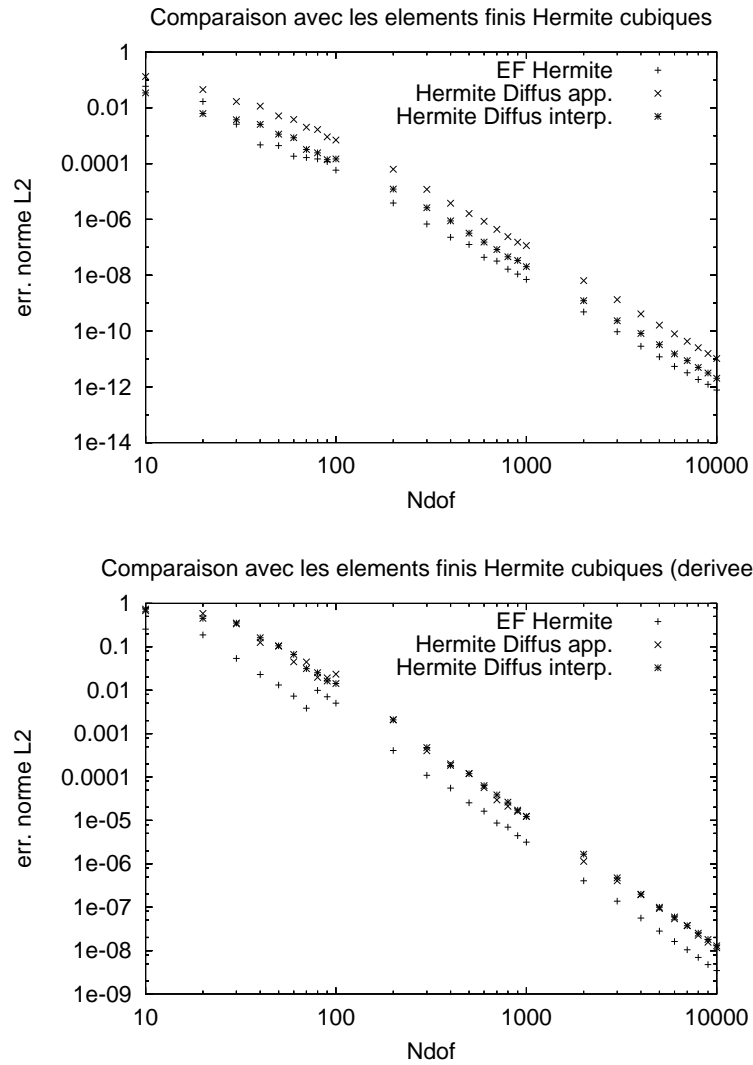


FIG. D.9: *Approximation diffuse Hermite cubique et Hermite cubique fonction test 3.*

## Annexe E

# Équation de la chaleur

### E.1 Mesures de la qualité de répartition d'un semis de nœuds 1d

La mesure de l'influence du semis de nœuds sur la qualité des simulations sans maillage suppose que l'on connaisse des critères de mesure de la qualité du semis de nœuds.

Nous avons utilisé trois indicateurs très simples qui sont (les nœuds sont classés par ordre croissant et sont les  $n + 1$  extrémités d'une partition de  $[0, 1]$ )

1. La distance minimale entre deux nœuds  $d = \frac{1}{h} \min \{x_{i+1} - x_i, i \in \llbracket 1, n \rrbracket\}$
2. La distance maximale entre deux nœuds  $D = \frac{1}{h} \max \{x_{i+1} - x_i, i \in \llbracket 1, n \rrbracket\}$
3. La variance des écarts entre nœuds  $\sigma^2 = \frac{1}{n} \sum_{i=0}^n \left(\frac{1}{n} - (x_{i+1} - x_i)\right)^2$

La figure E.1 montre les courbes  $(d, D)$ ,  $(d, \sigma^2)$ ,  $(d, \sigma)$  et  $(D, \sigma)$  pour des semis réguliers plus ou moins perturbés. Les différents estimateurs sont relativement corrélés, on peut donc se contenter d'utiliser un de ces indicateurs. Nous avons choisi de travailler avec l'indicateur  $\sigma$  qui montre le mieux l'influence du semis de nœuds sur la qualité de la simulation.

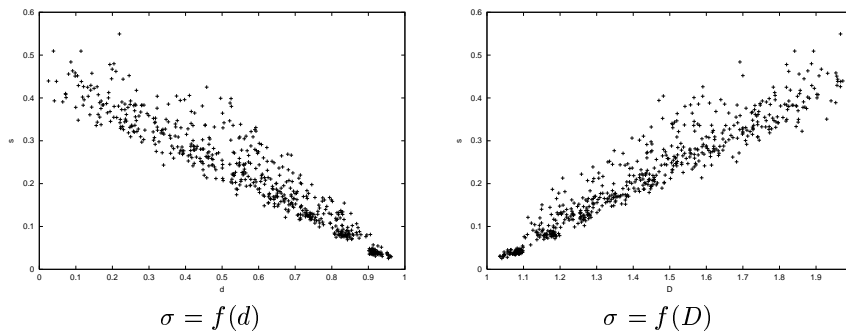


FIG. E.1: Relation entre les mesures de qualité d'un semis de nœuds 1d. Courbes expérimentales.

## E.2 Résultats expérimentaux

### E.2.1 Influence du semis de nœuds

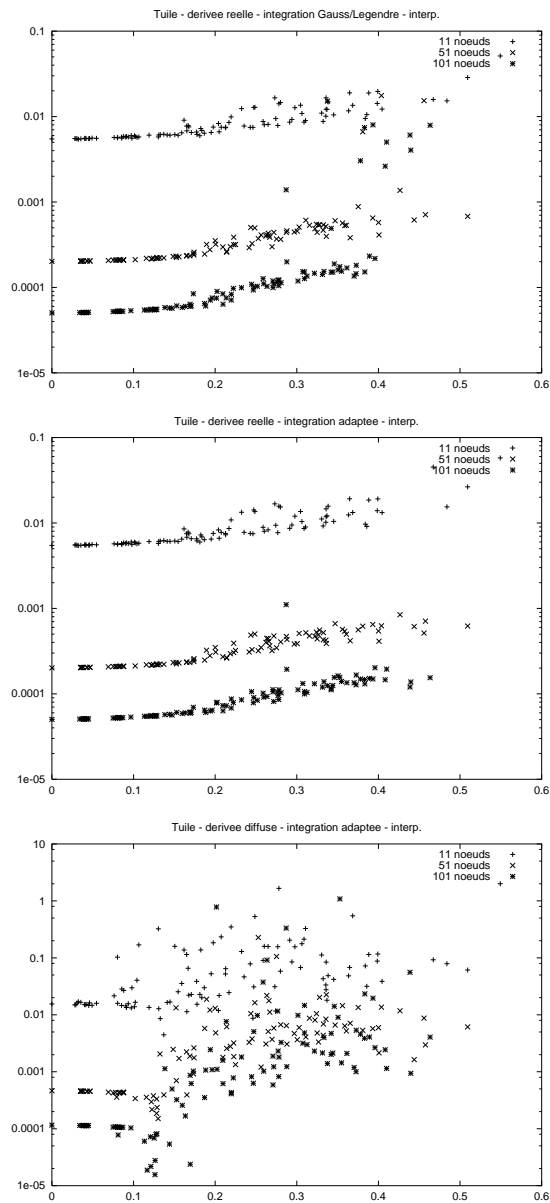


FIG. E.2: Influence du semis de nœuds sur les méthodes à tuiles.

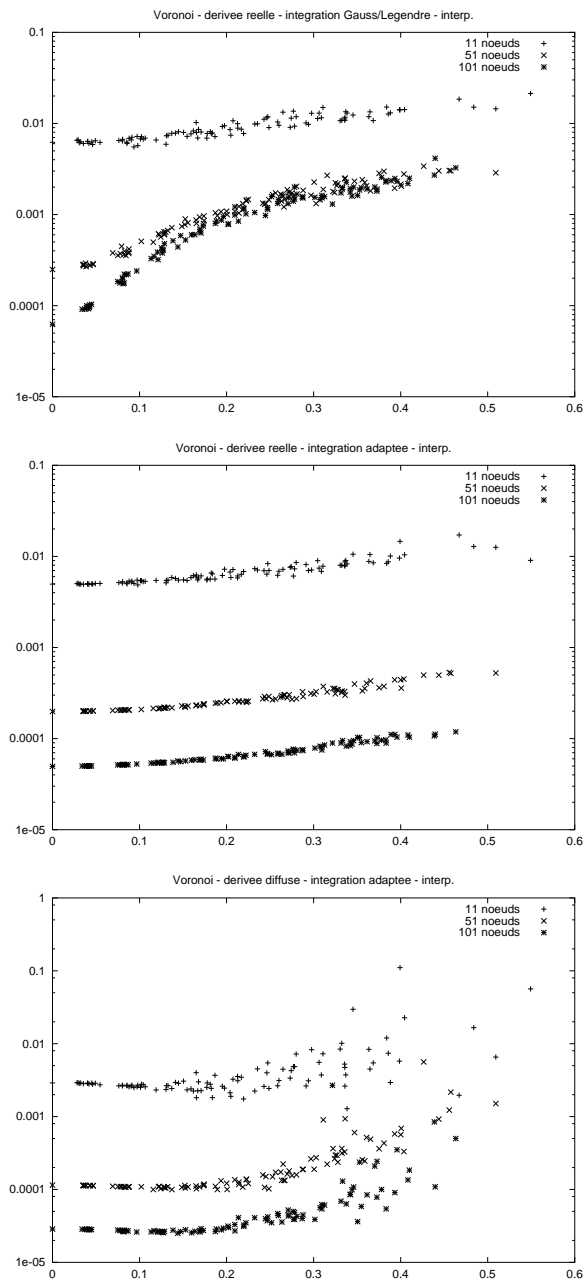


FIG. E.3: Influence du semis de nœuds sur les méthodes « voronoï » interpolantes.



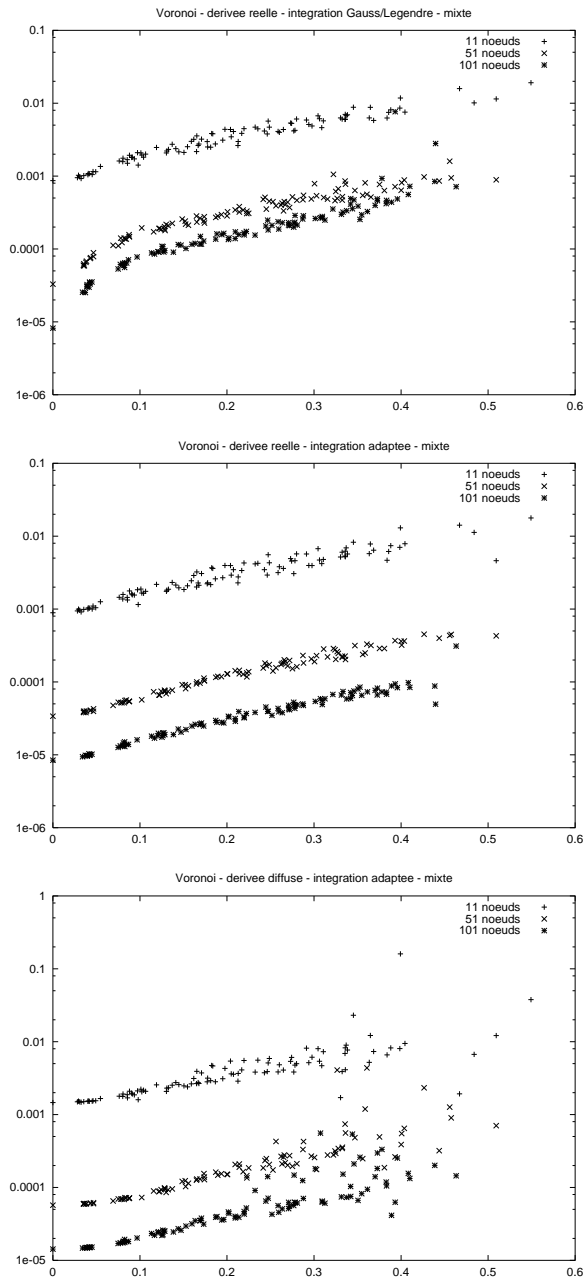


FIG. E.4: Influence du semis de nœuds sur les méthodes « voronoï » mixtes.

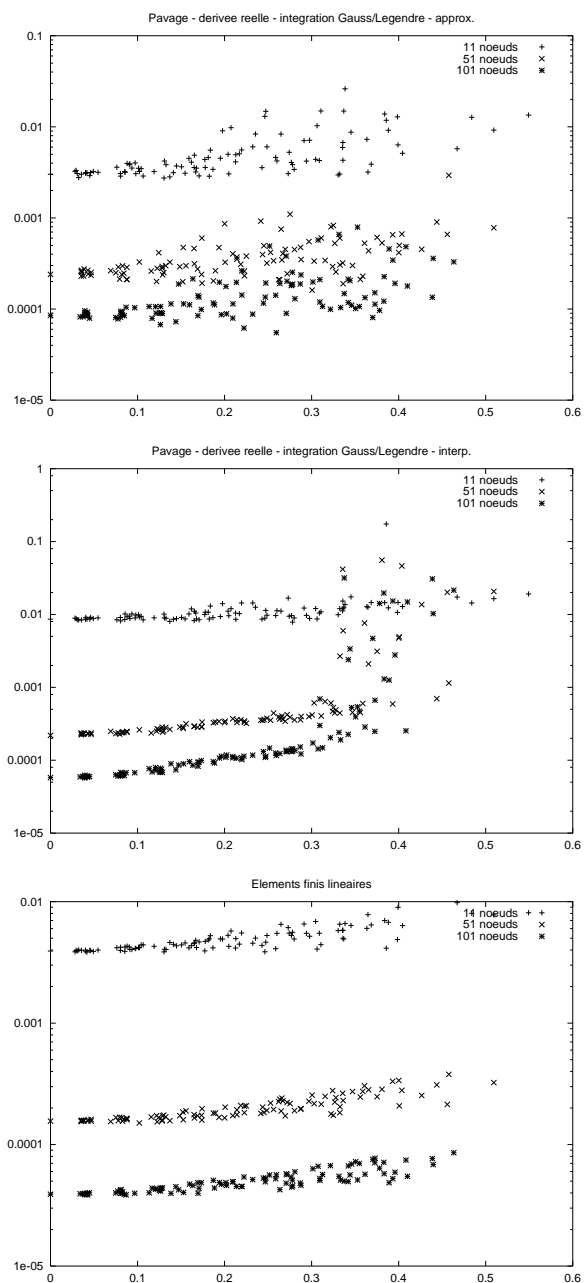


FIG. E.5: Influence du semis de nœuds sur les méthodes efg (avec approximation et interpolation) et éléments finis.

## E.2.2 Dirichlet – Problème 1

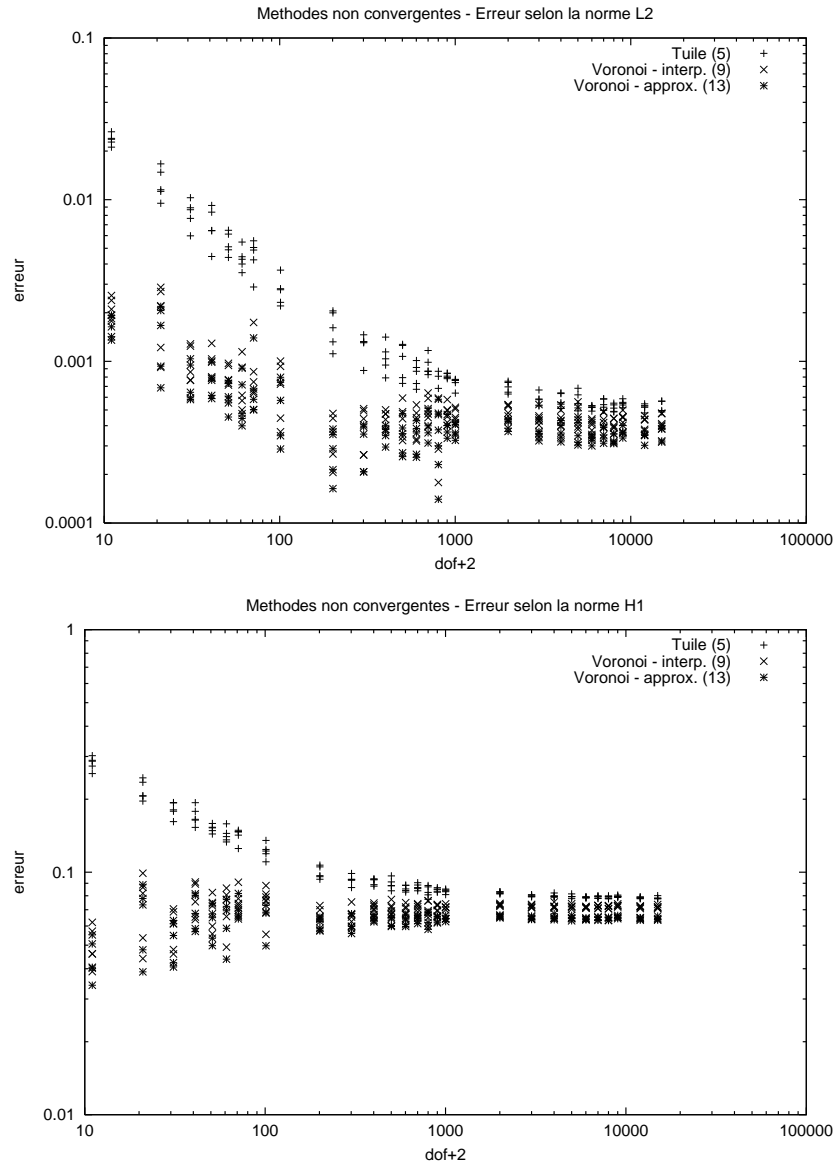


FIG. E.6: Méthodes 5, 9 et 13 (Gauss–Legendre, dérivée diffuse) sur le problème 1. Analyse de la convergence.

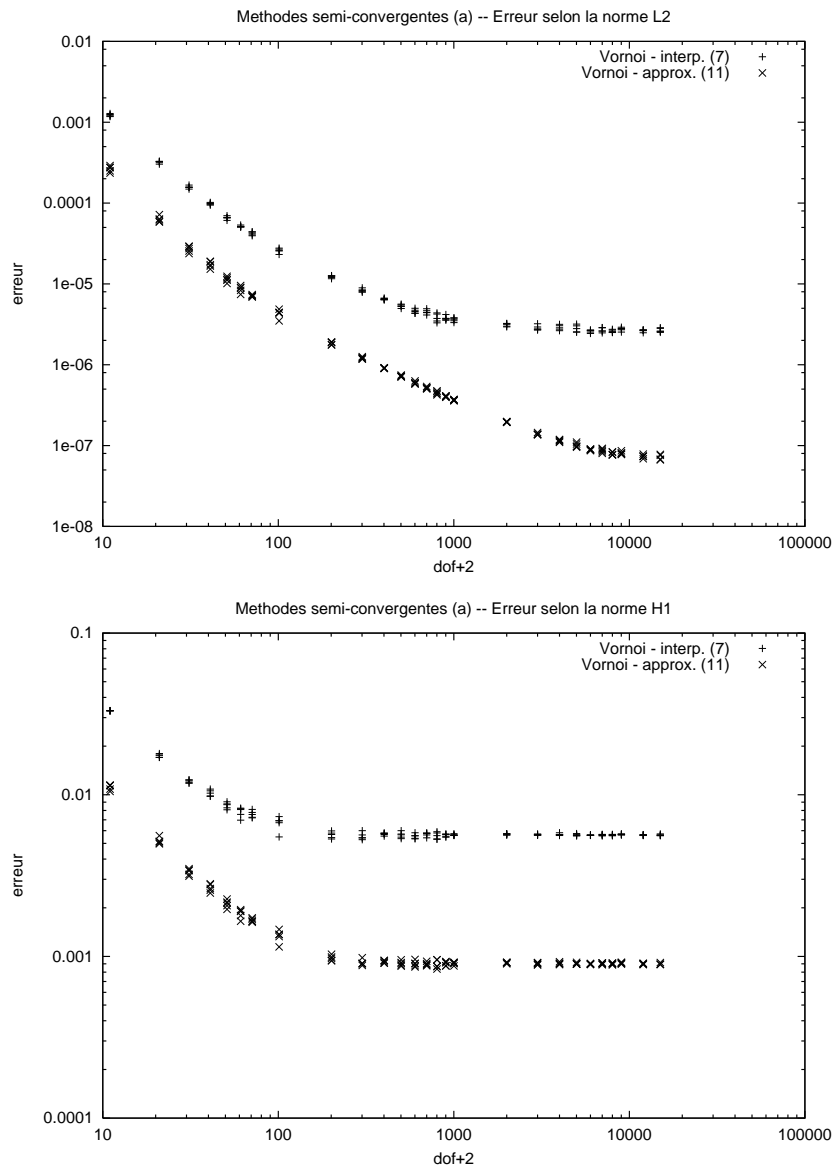


FIG. E.7: Méthodes 7 et 11 (Dérivée réelle, Gauss–Legendre) sur le problème 1. Analyse de la convergence.

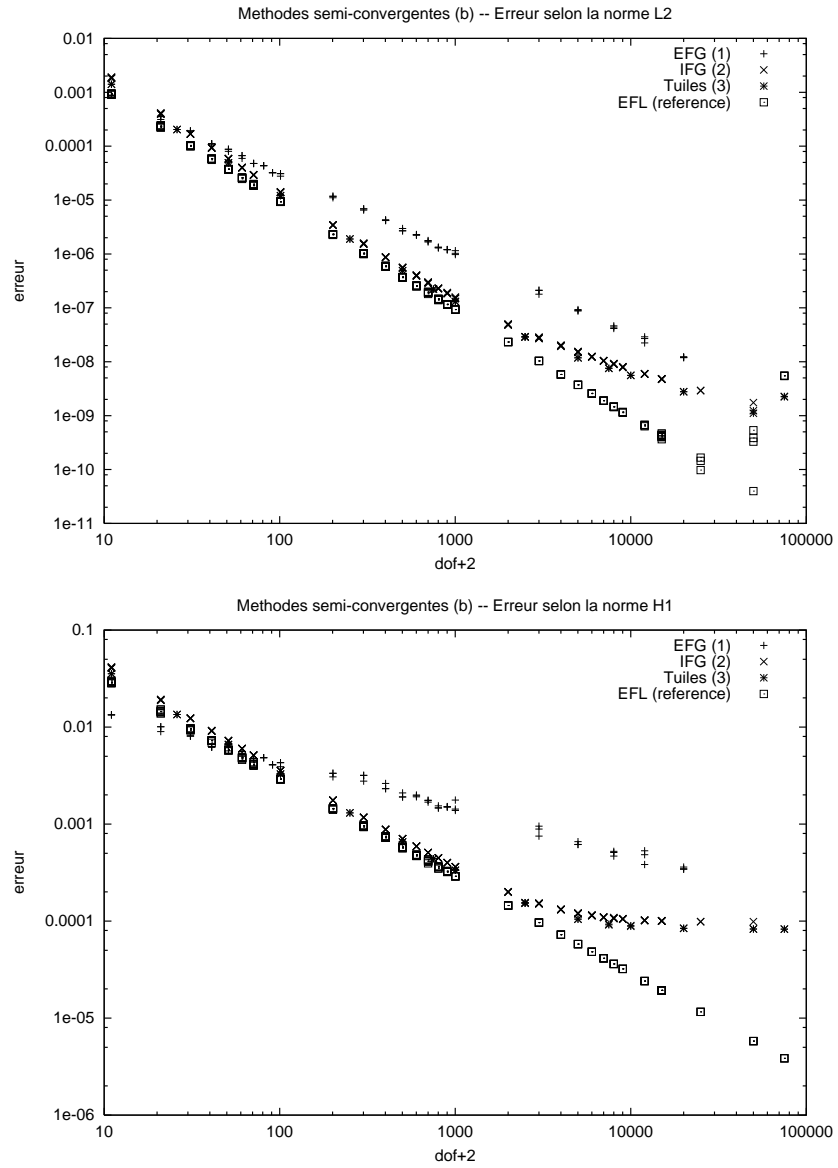


FIG. E.8: Méthodes 1, 2 et 3 (Dérivée réelle, Gauss-Legendre) sur le problème 1. Analyse de la convergence.

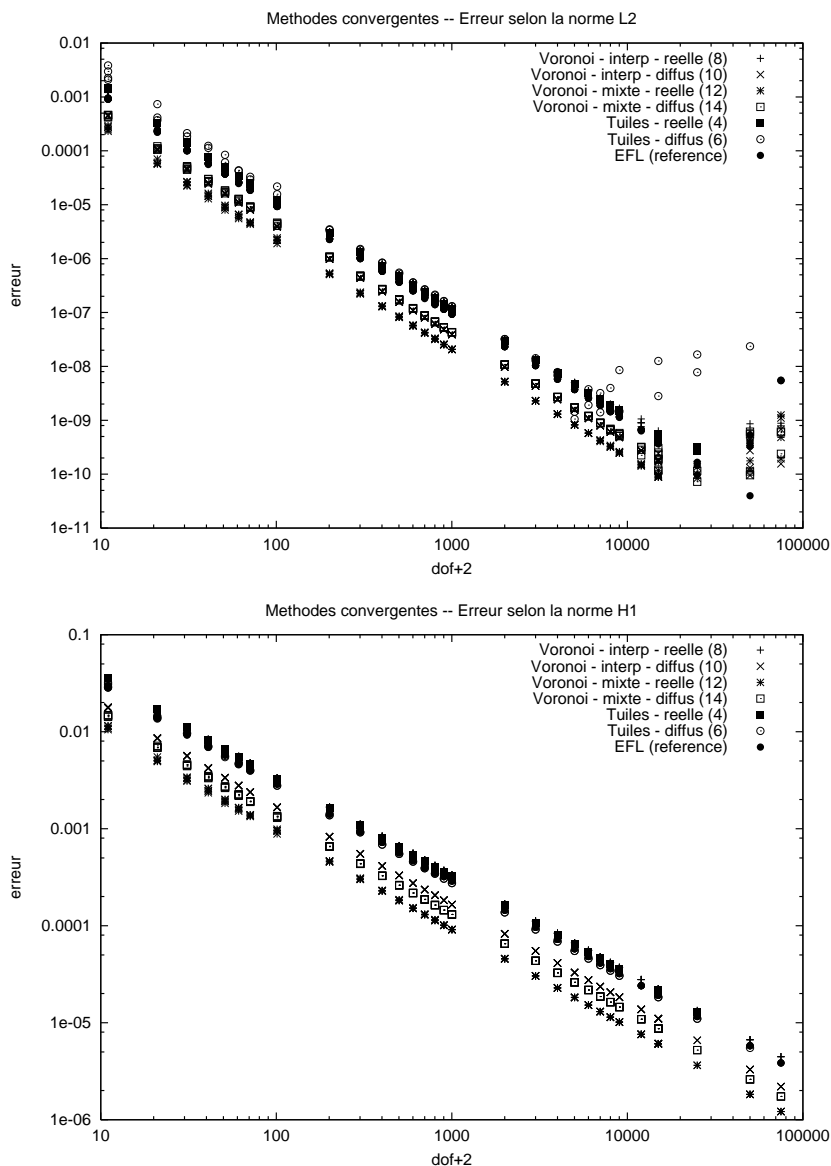


FIG. E.9: Méthodes convergentes sur le problème 1.

## E.2.3 Dirichlet – Problème 3

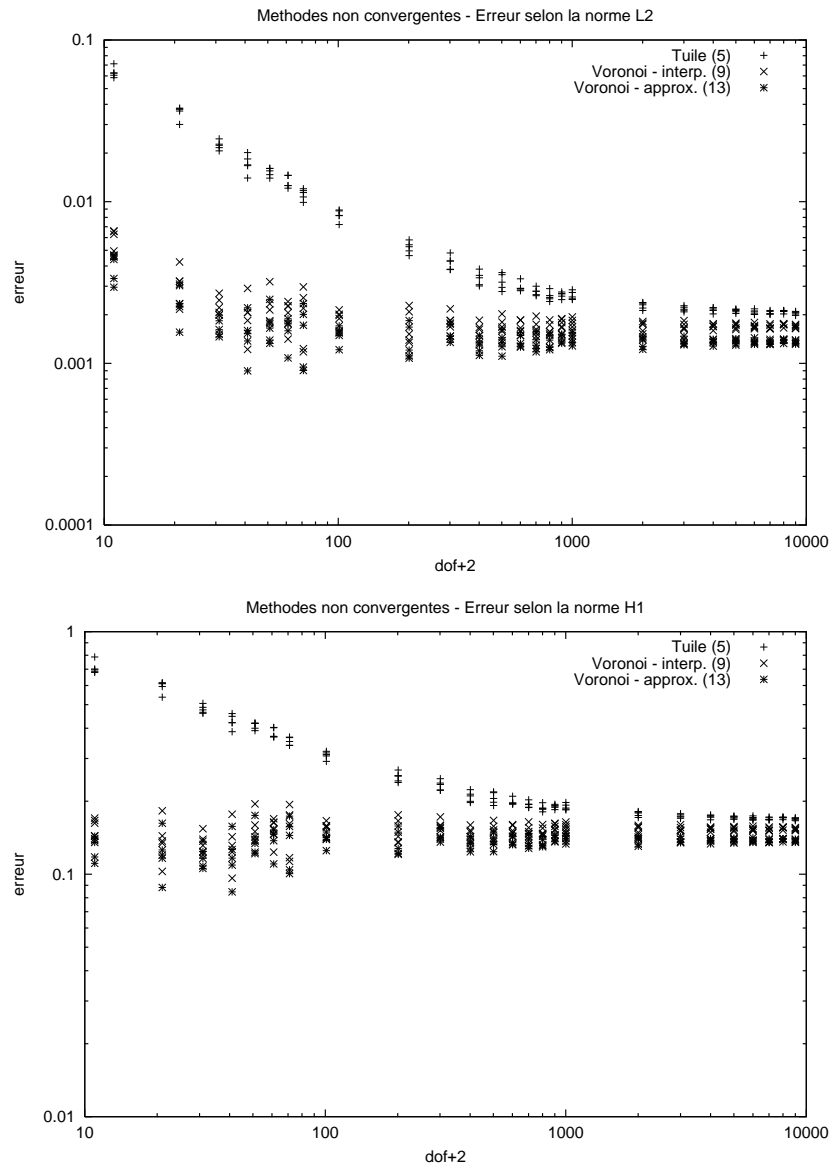


FIG. E.10: Méthodes 5, 9 et 13 (Gauss–Legendre, dérivée diffuse) sur le problème 3. Analyse de la convergence.

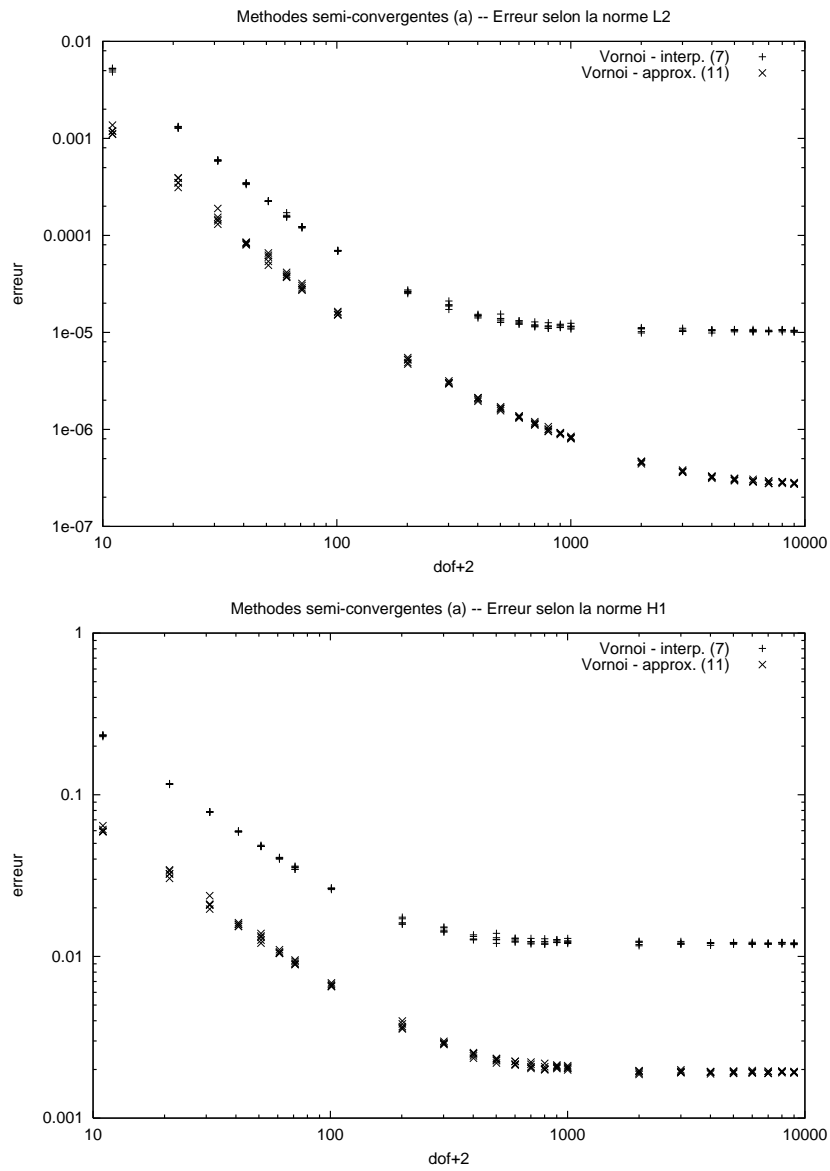


FIG. E.11: Méthodes 7 et 11 (Dérivée réelle, Gauss–Legendre) sur le problème 3. Analyse de la convergence.



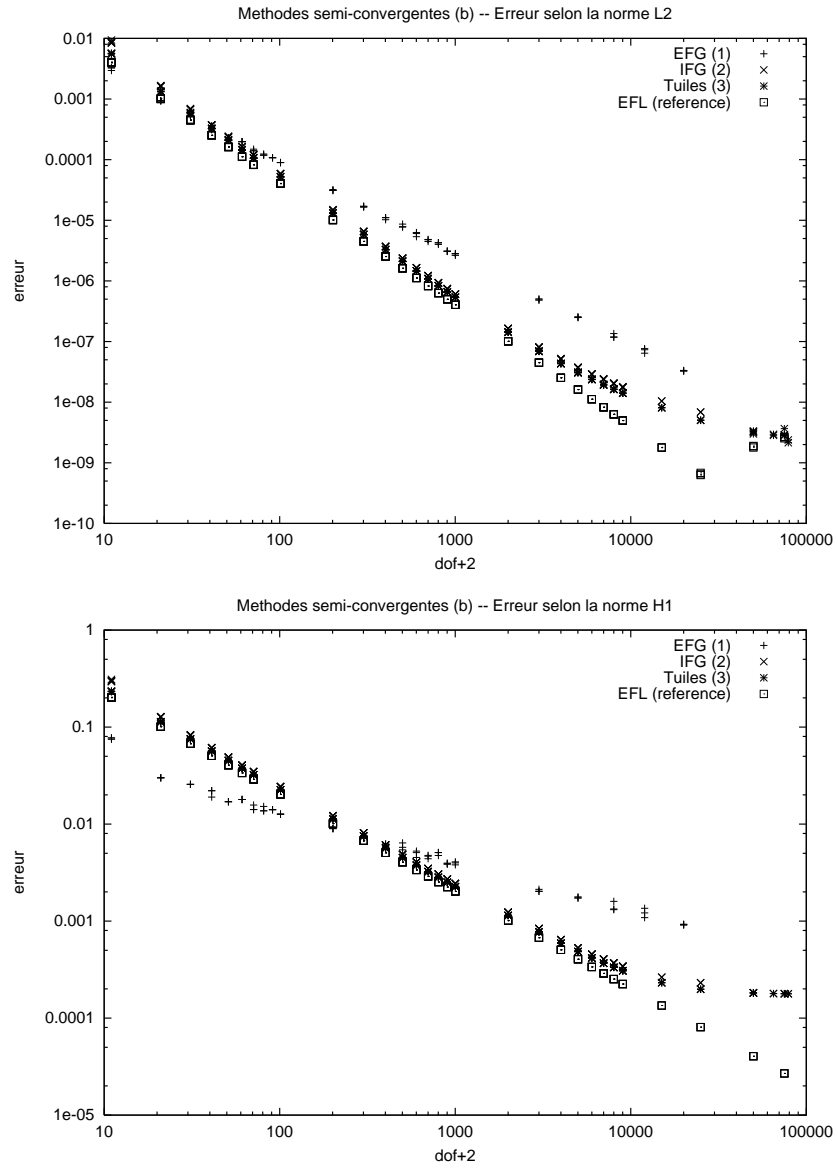


FIG. E.12: Méthodes 1, 2 et 3 (Dérivée réelle, Gauss-Legendre) sur le problème 3. Analyse de la convergence.

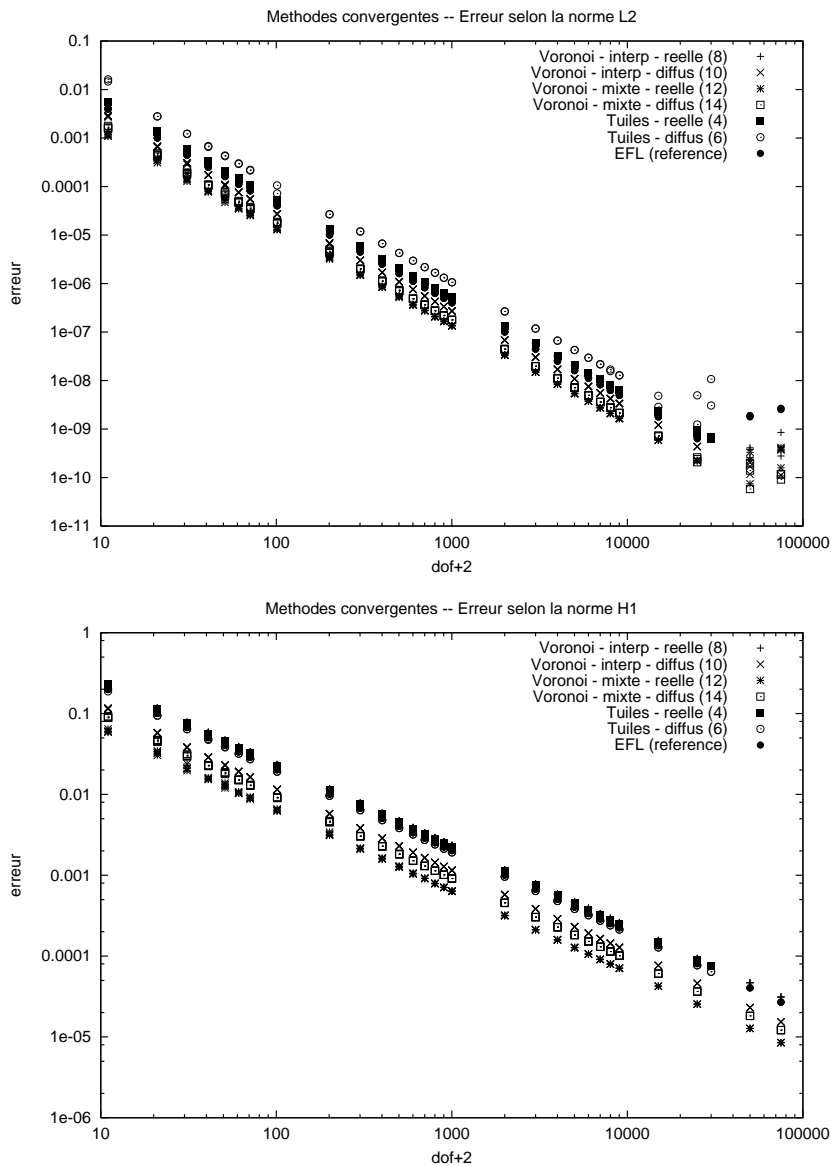


FIG. E.13: Méthodes convergentes sur le problème 3.

## Patch test

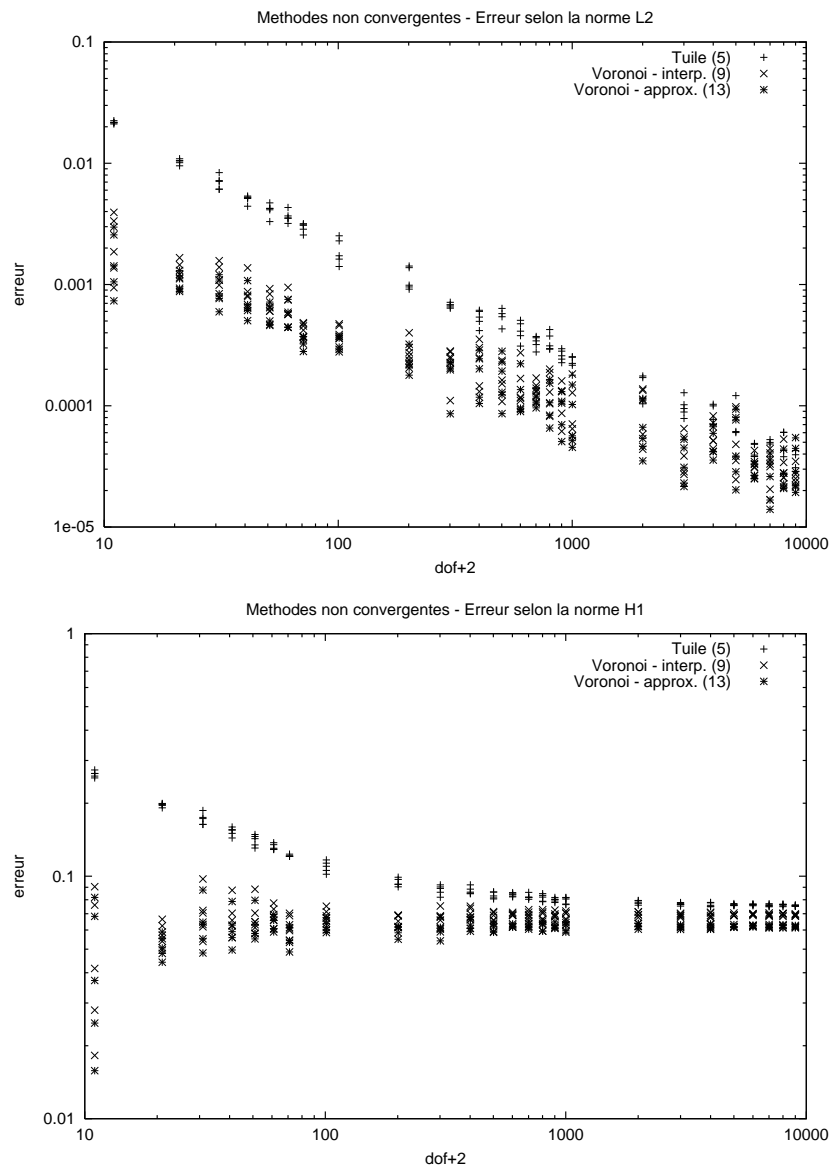


FIG. E.14: Méthodes 5, 9 et 13 (Gauss–Legendre, dérivée diffuse) sur le patch test. Analyse de la convergence.

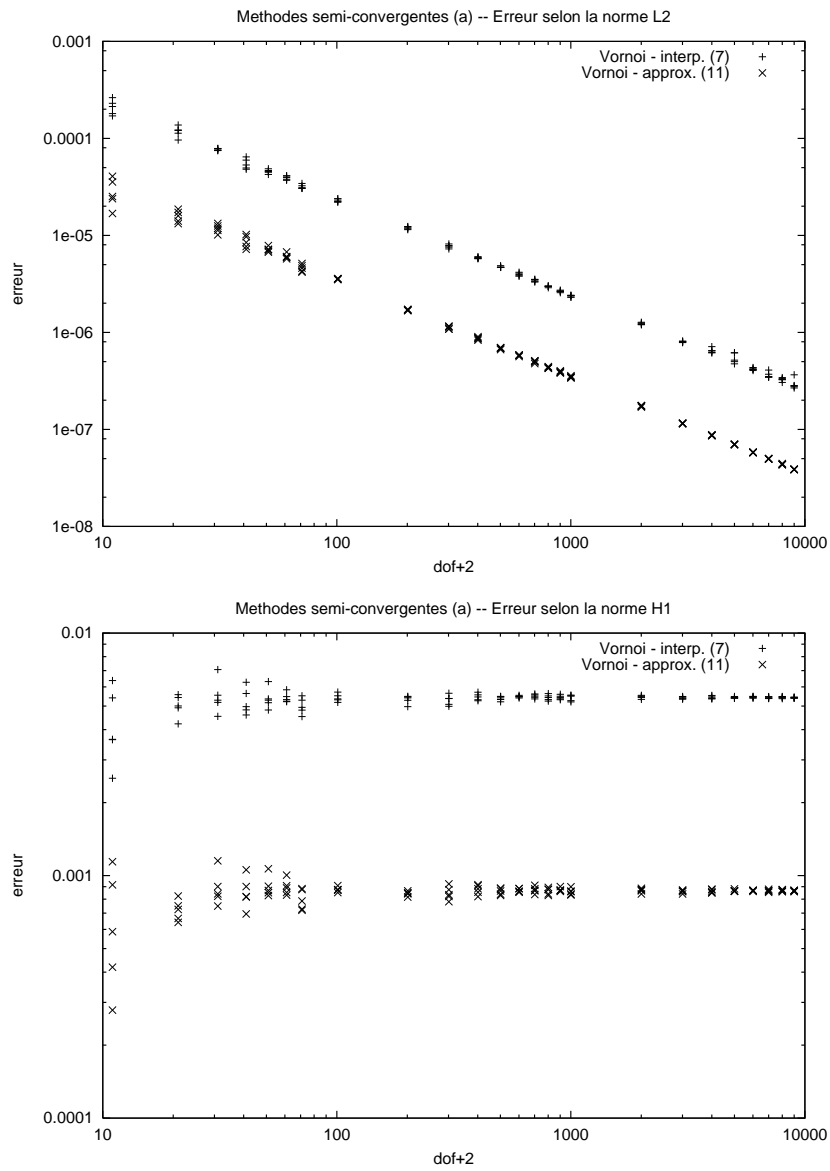


FIG. E.15: Méthodes 7 et 11 (Dérivée réelle, Gauss-Legendre) sur le patch test. Analyse de la convergence.

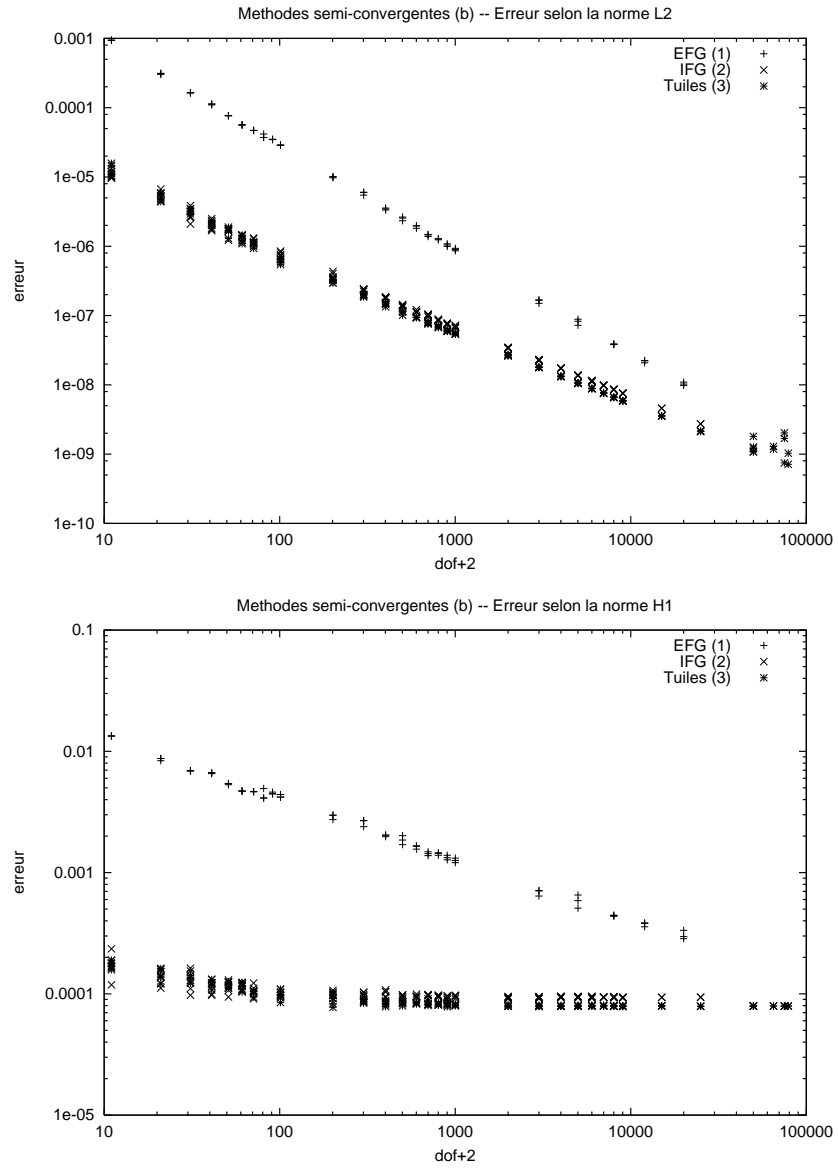


FIG. E.16: Méthodes 1, 2 et 3 (Dérivée réelle, Gauss-Legendre) sur le patch test. Analyse de la convergence.

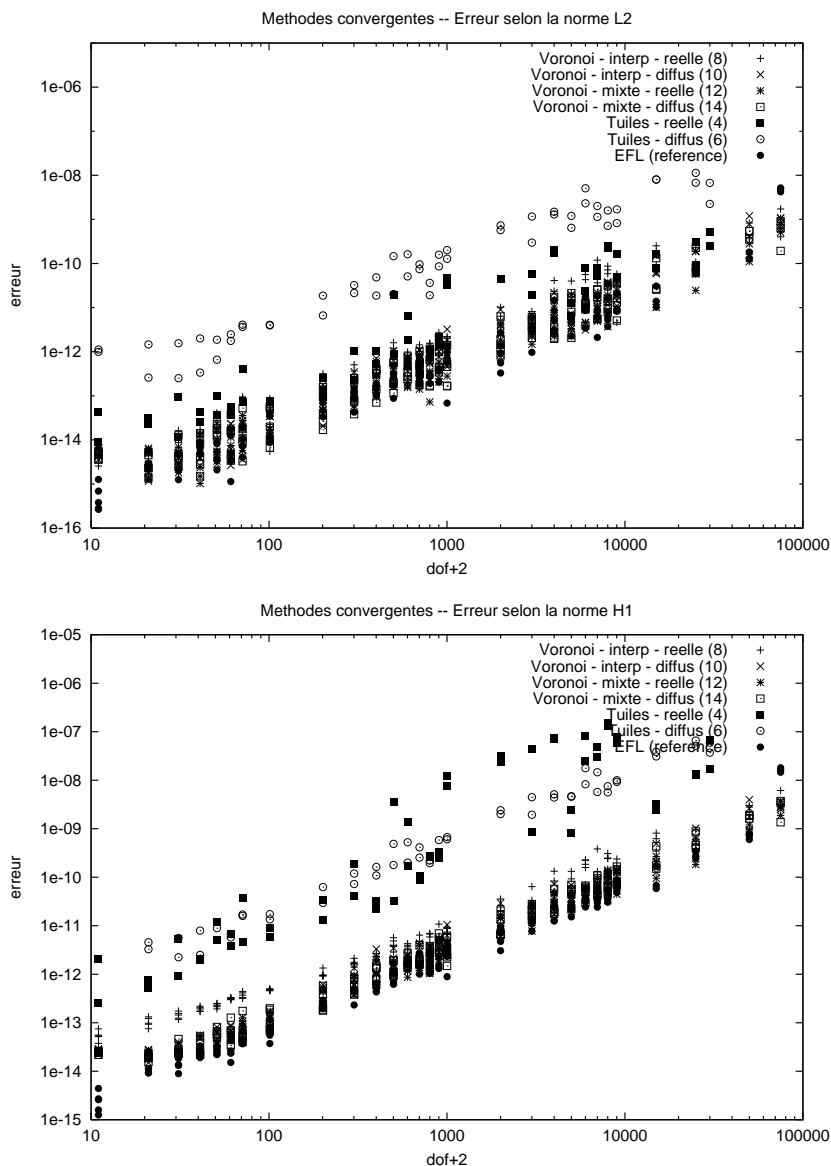


FIG. E.17: Méthodes convergentes sur le patch test.

## E.2.4 Dirichlet – Problème 2

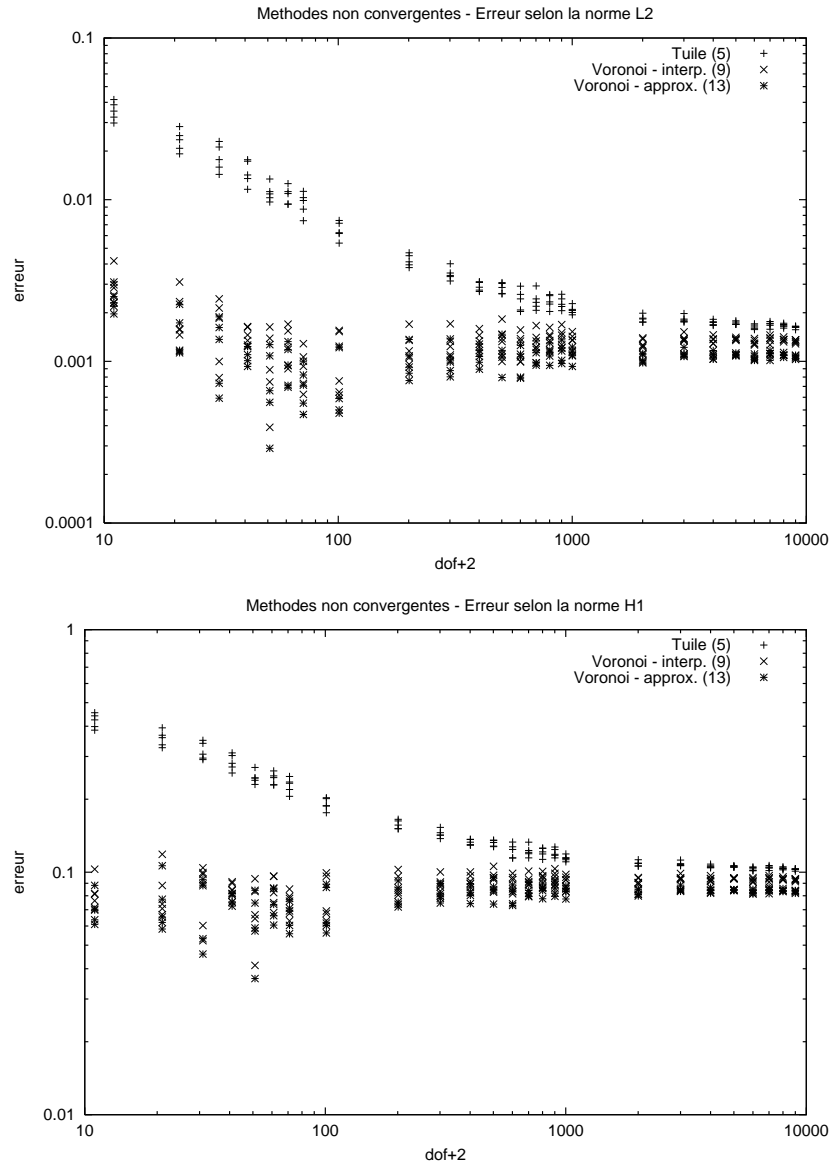


FIG. E.18: Méthodes 5, 9 et 13 (Gauss–Legendre, dérivée diffuse) sur le problème 2. Analyse de la convergence.

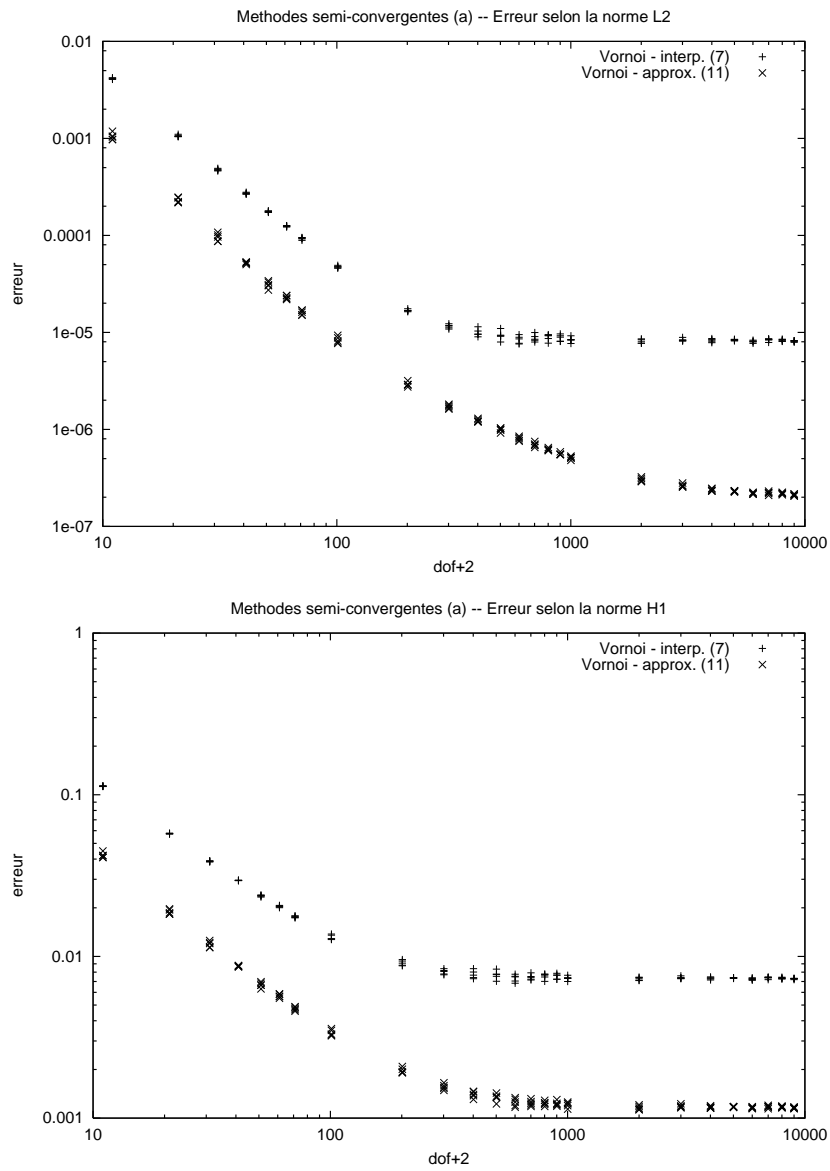


FIG. E.19: Méthodes 7 et 11 (Dérivée réelle, Gauss–Legendre) sur le problème 2. Analyse de la convergence.



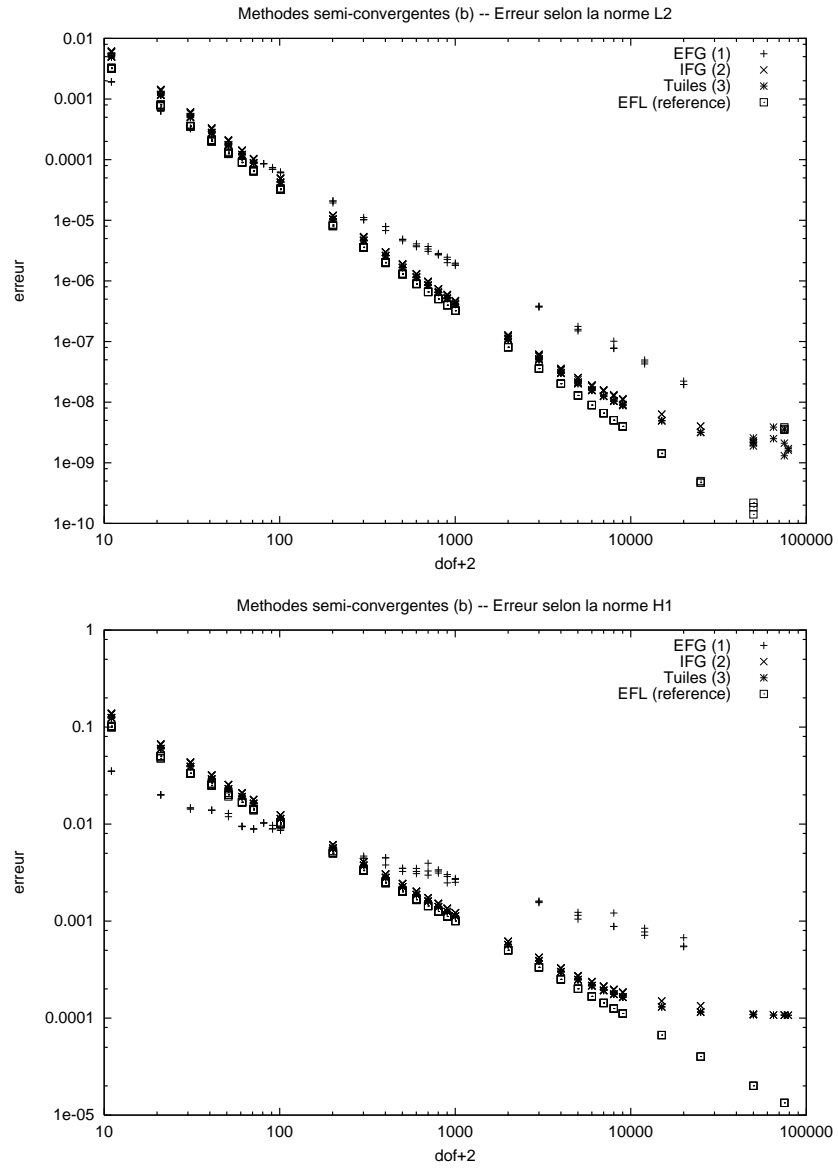


FIG. E.20: Méthodes 1, 2 et 3 (Dérivée réelle, Gauss-Legendre) sur le problème 2. Analyse de la convergence.

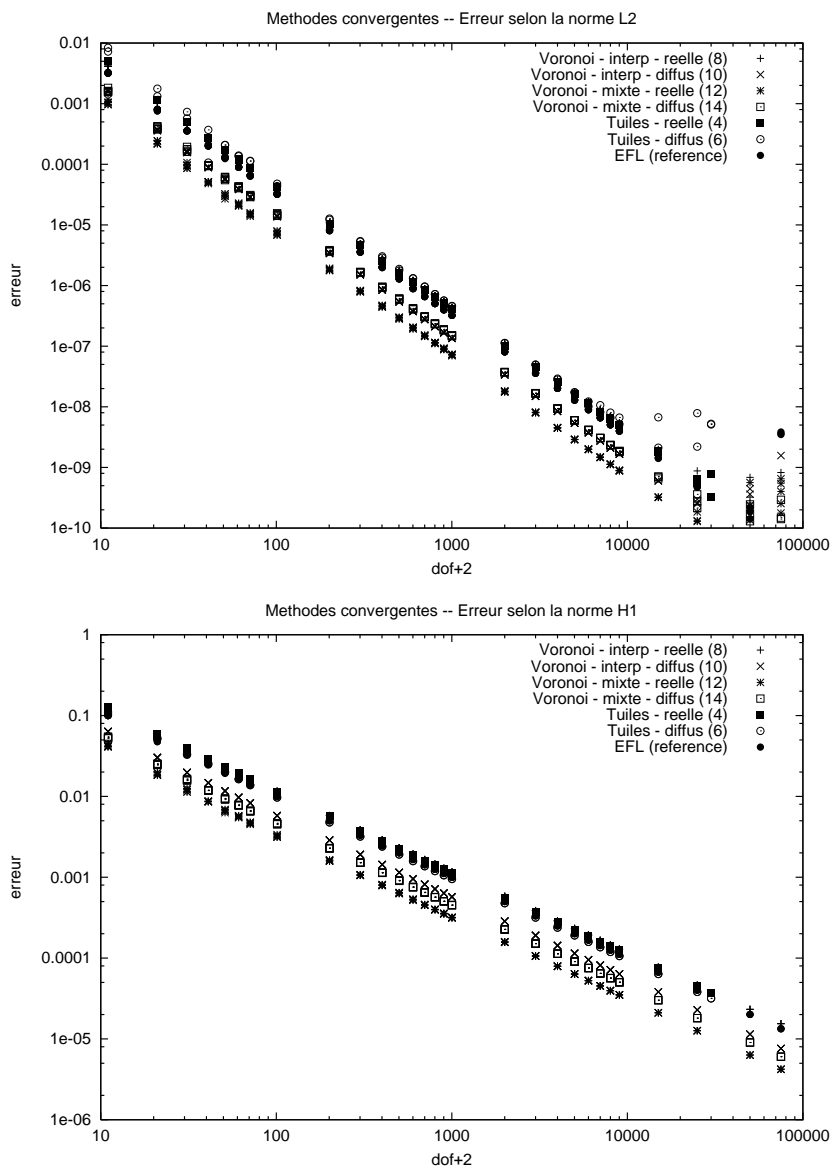


FIG. E.21: Méthodes convergentes sur le problème 2.

## E.2.5 Dirichlet – Problème 4

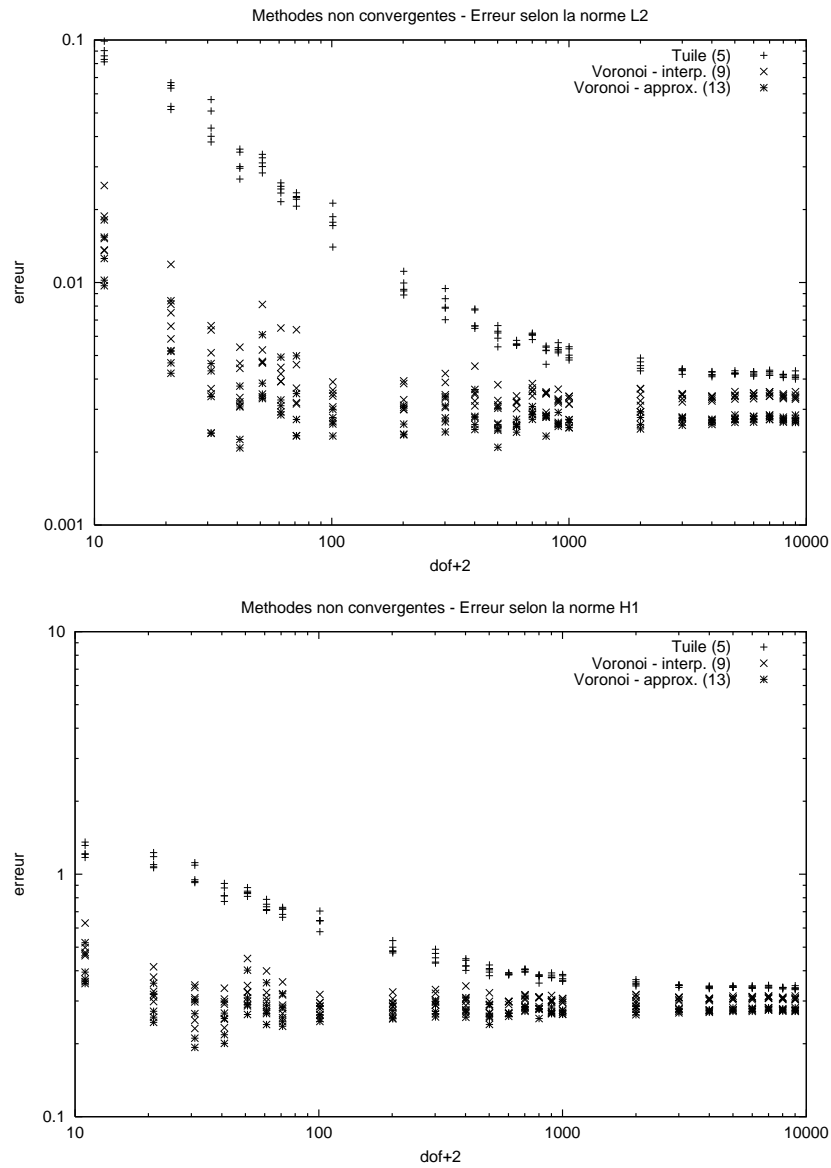


FIG. E.22: Méthodes 5, 9 et 13 (Gauss–Legendre, dérivée diffuse) sur le problème 4. Analyse de la convergence.

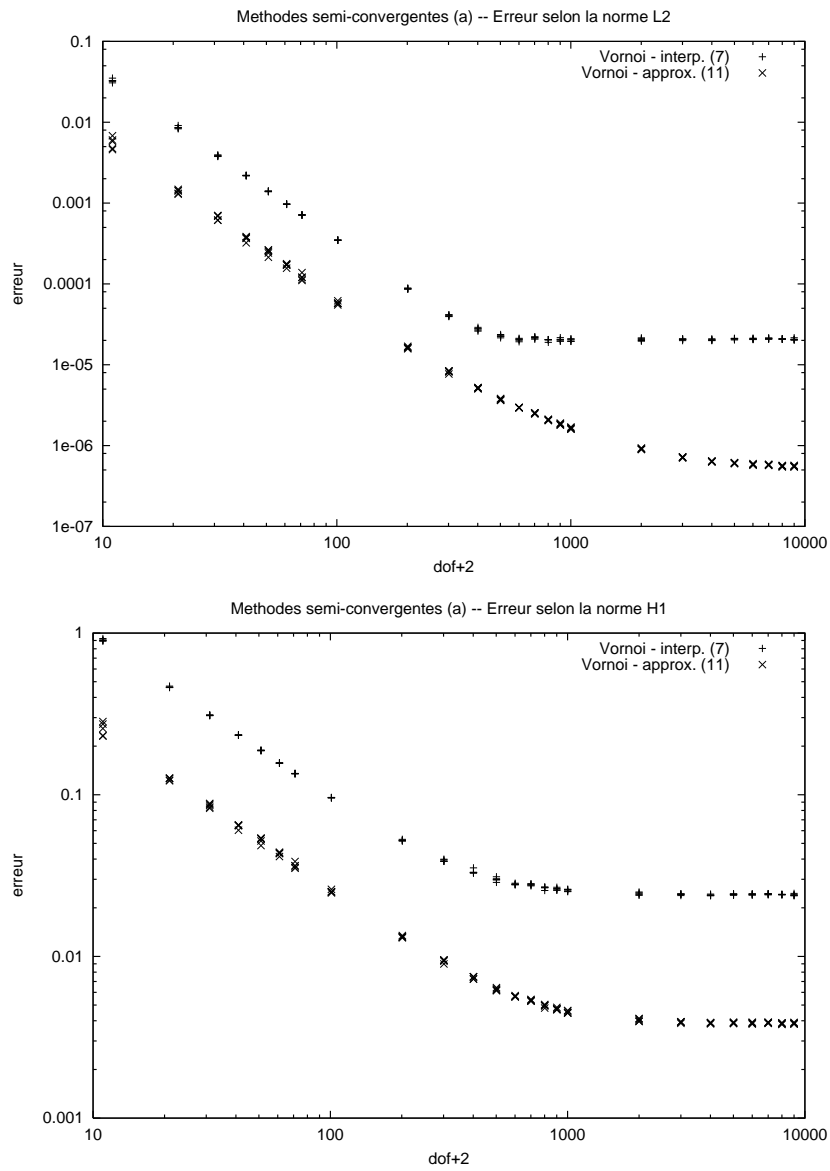


FIG. E.23: Méthodes 7 et 11 (Dérivée réelle, Gauss–Legendre) sur le problème 4. Analyse de la convergence.

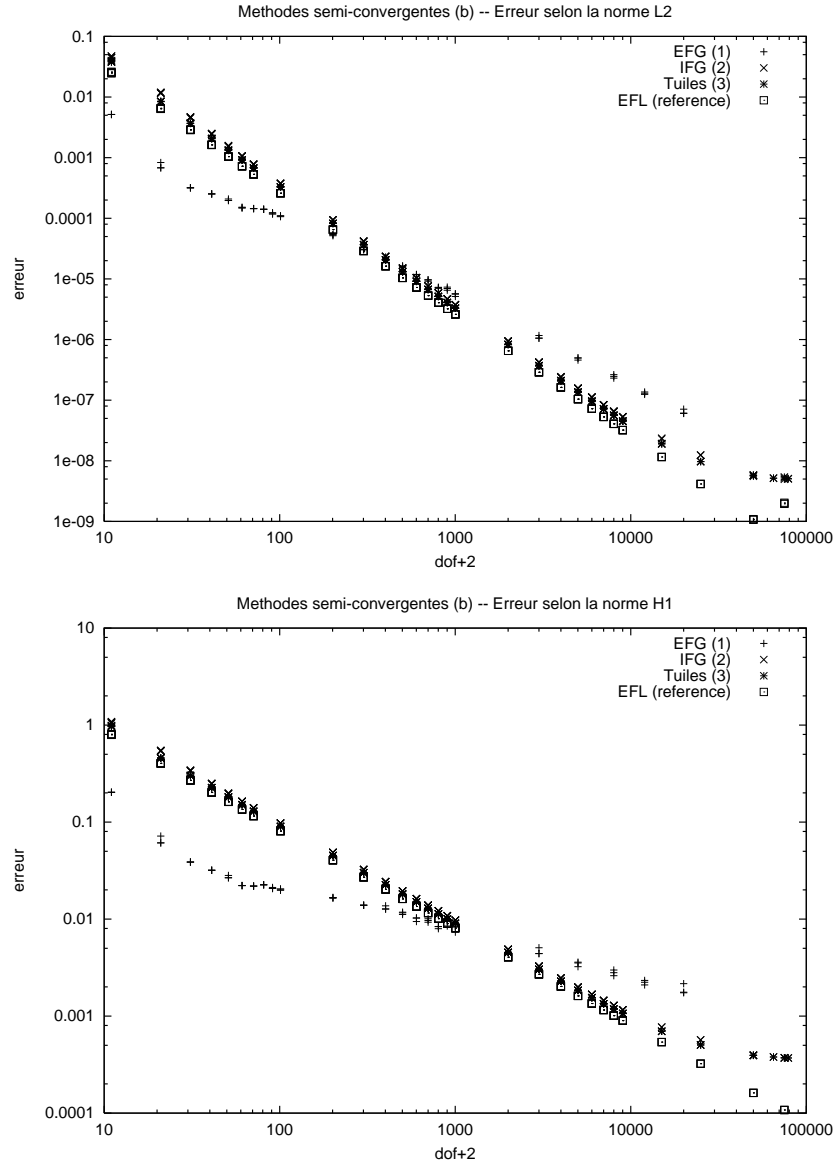


FIG. E.24: Méthodes 1, 2 et 3 (Dérivée réelle, Gauss-Legendre) sur le problème 4. Analyse de la convergence.

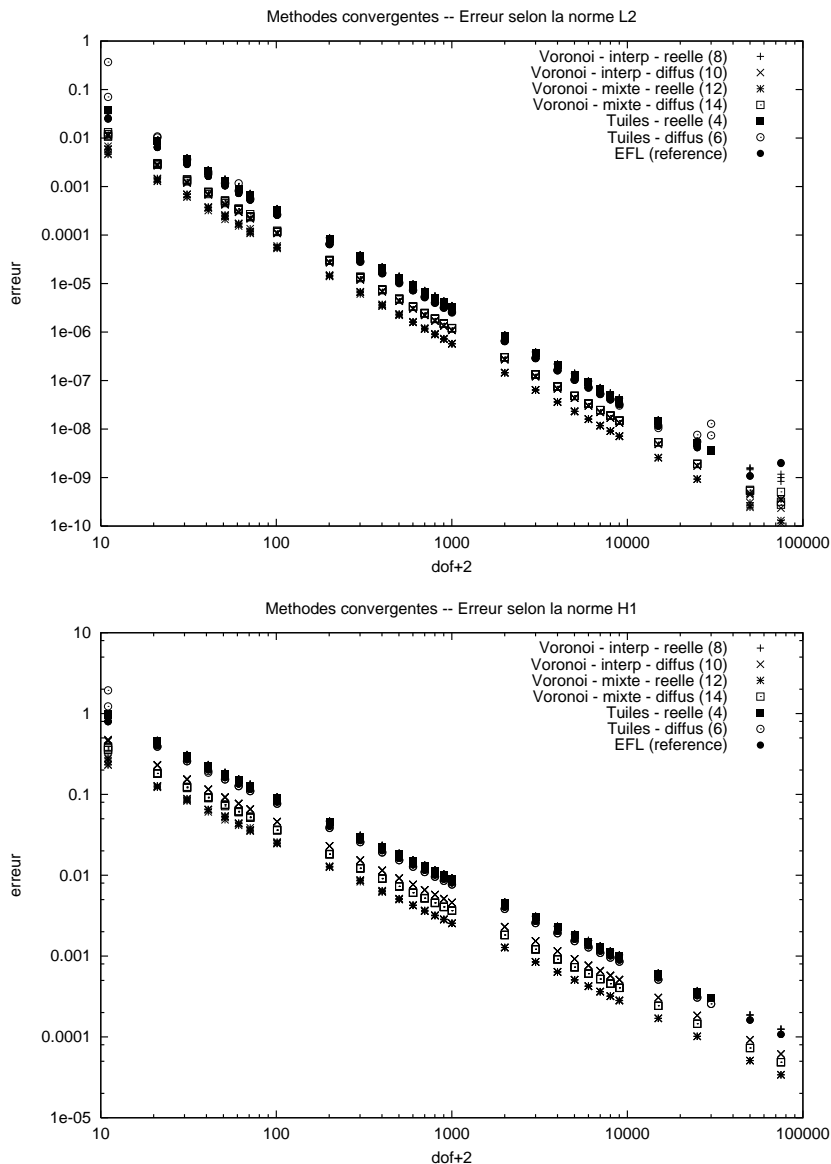


FIG. E.25: Méthodes convergentes sur le problème 4.

## E.2.6 Dirichlet – Problème 5

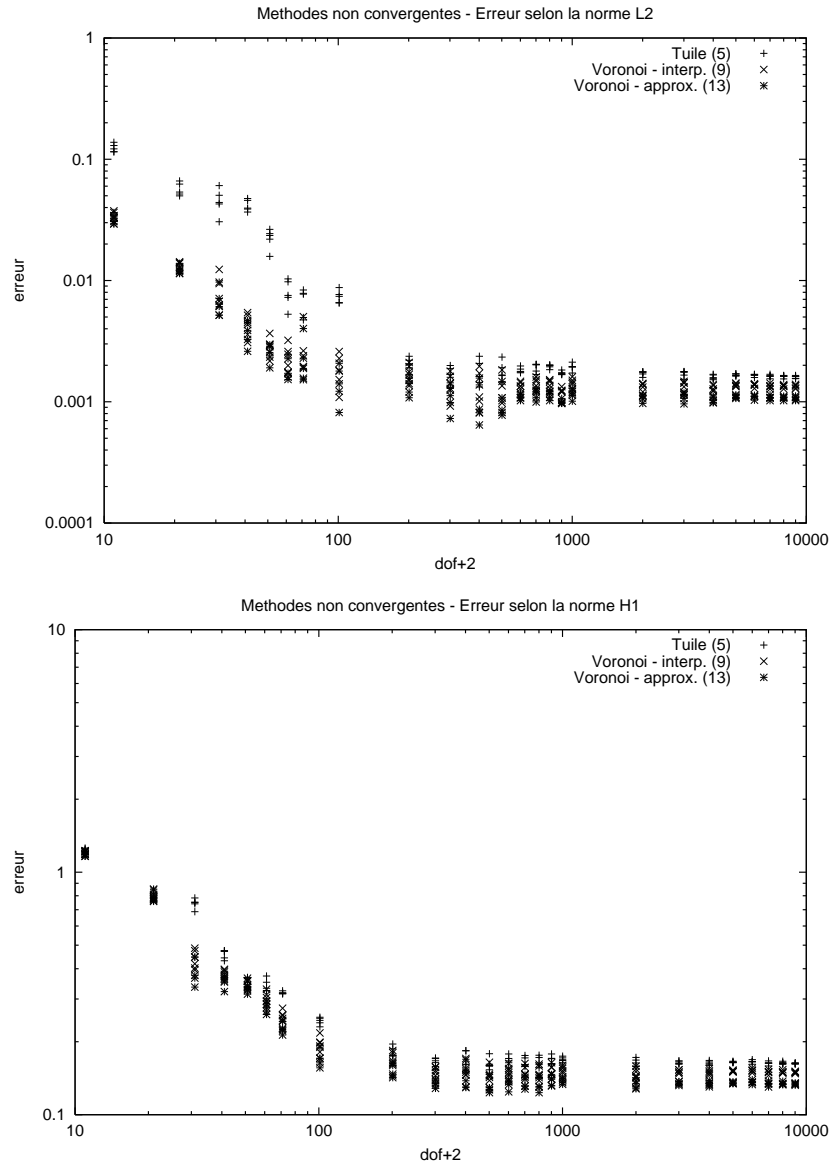


FIG. E.26: Méthodes 5, 9 et 13 (Gauss–Legendre, dérivée diffuse) sur le problème 5. Analyse de la convergence.

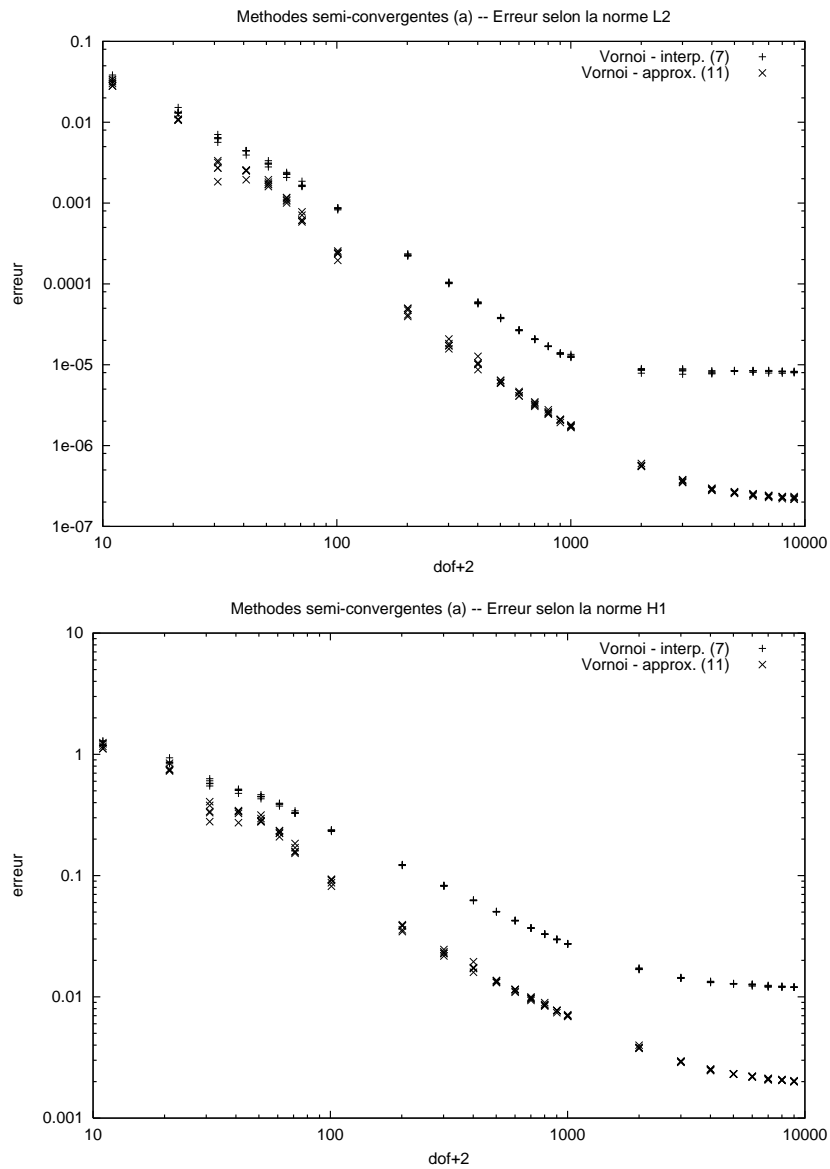


FIG. E.27: Méthodes 7 et 11 (Dérivée réelle, Gauss–Legendre) sur le problème 5. Analyse de la convergence.



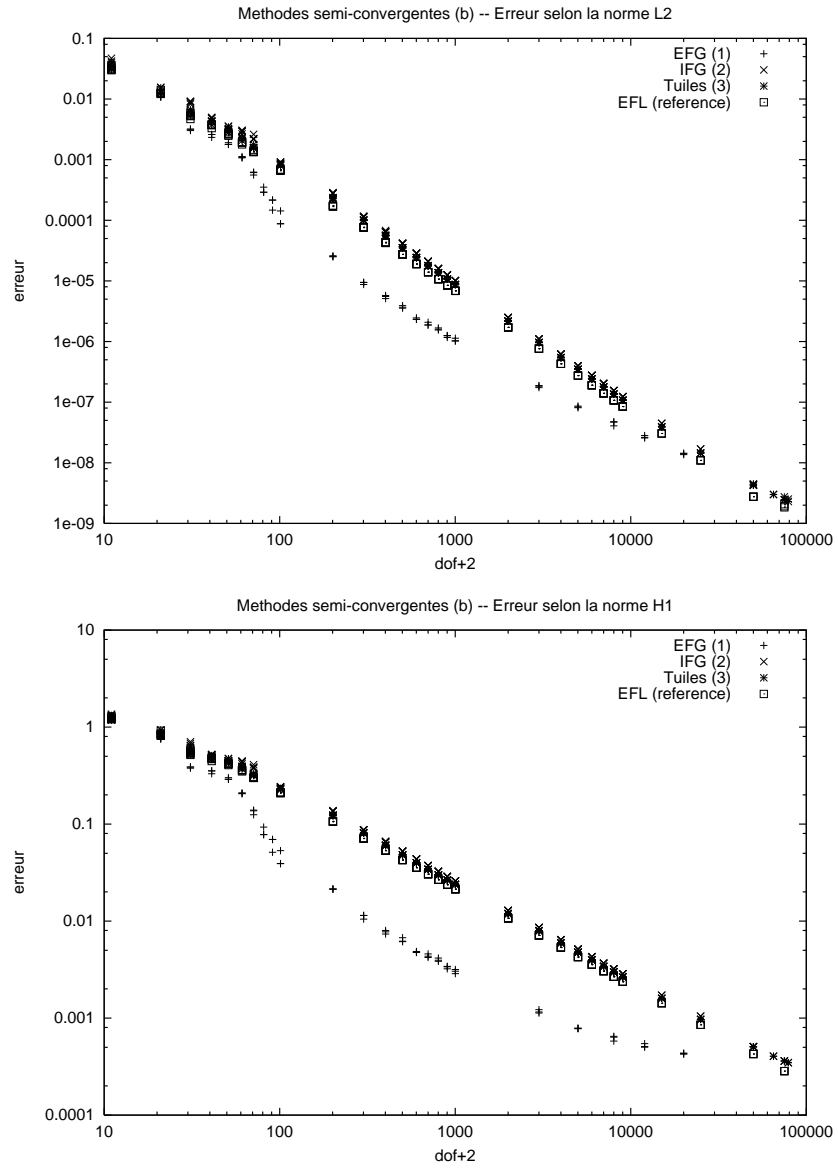


FIG. E.28: Méthodes 1, 2 et 3 (Dérivée réelle, Gauss-Legendre) sur le problème 5. Analyse de la convergence.

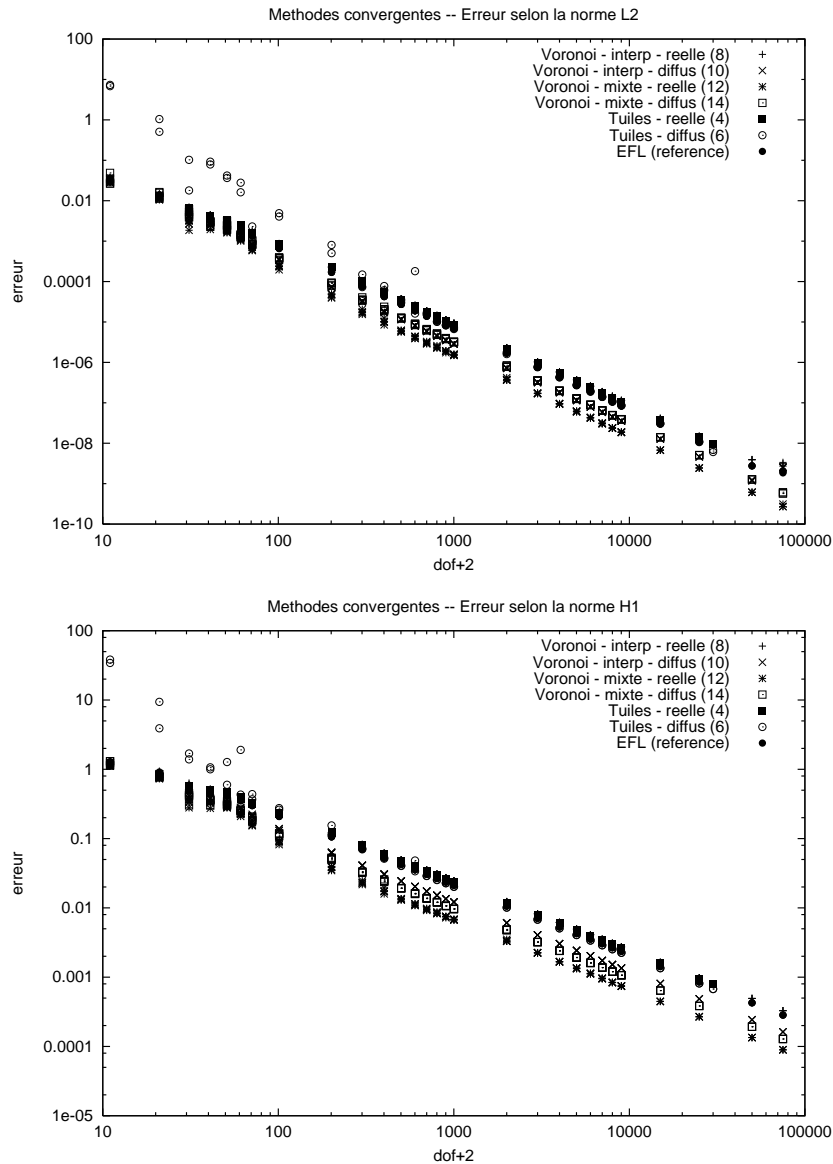


FIG. E.29: Méthodes convergentes sur le problème 5.

## E.2.7 Dirichlet – Problème 6

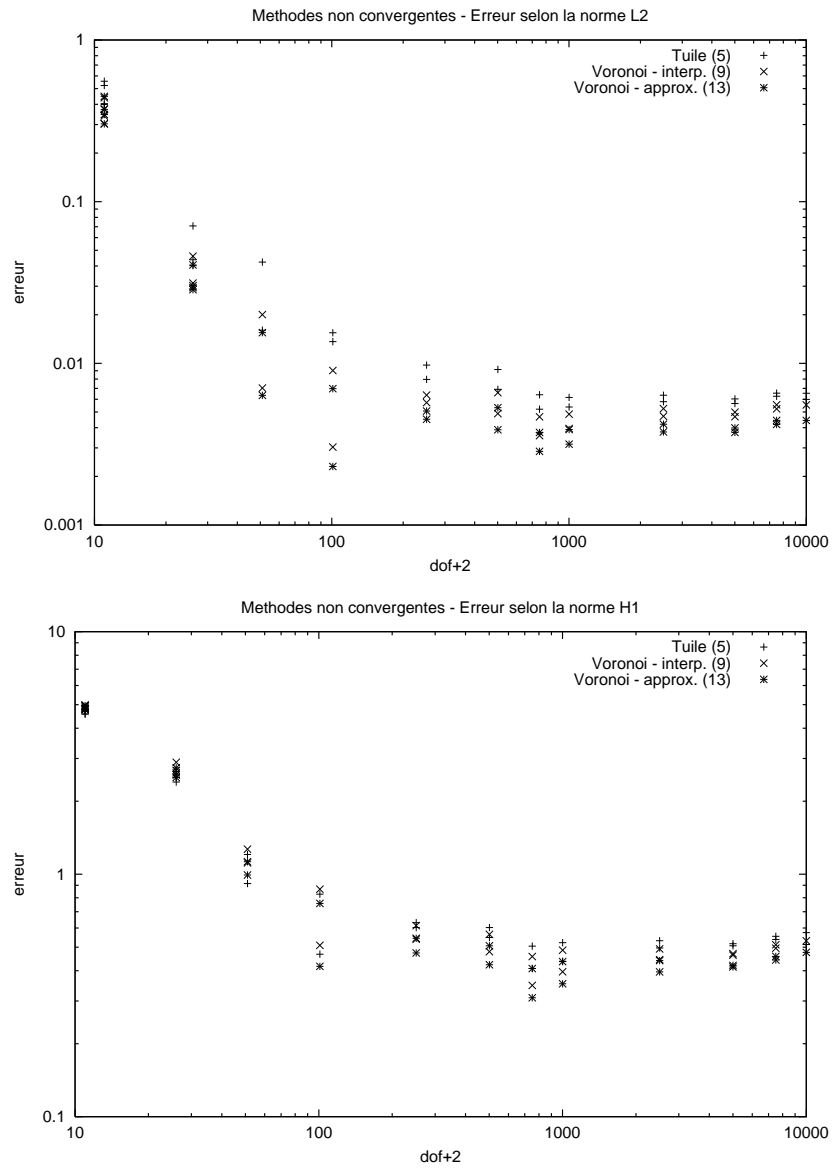


FIG. E.30: Méthodes 5, 9 et 13 (Gauss–Legendre, dérivée diffuse) sur le problème 6. Analyse de la convergence.

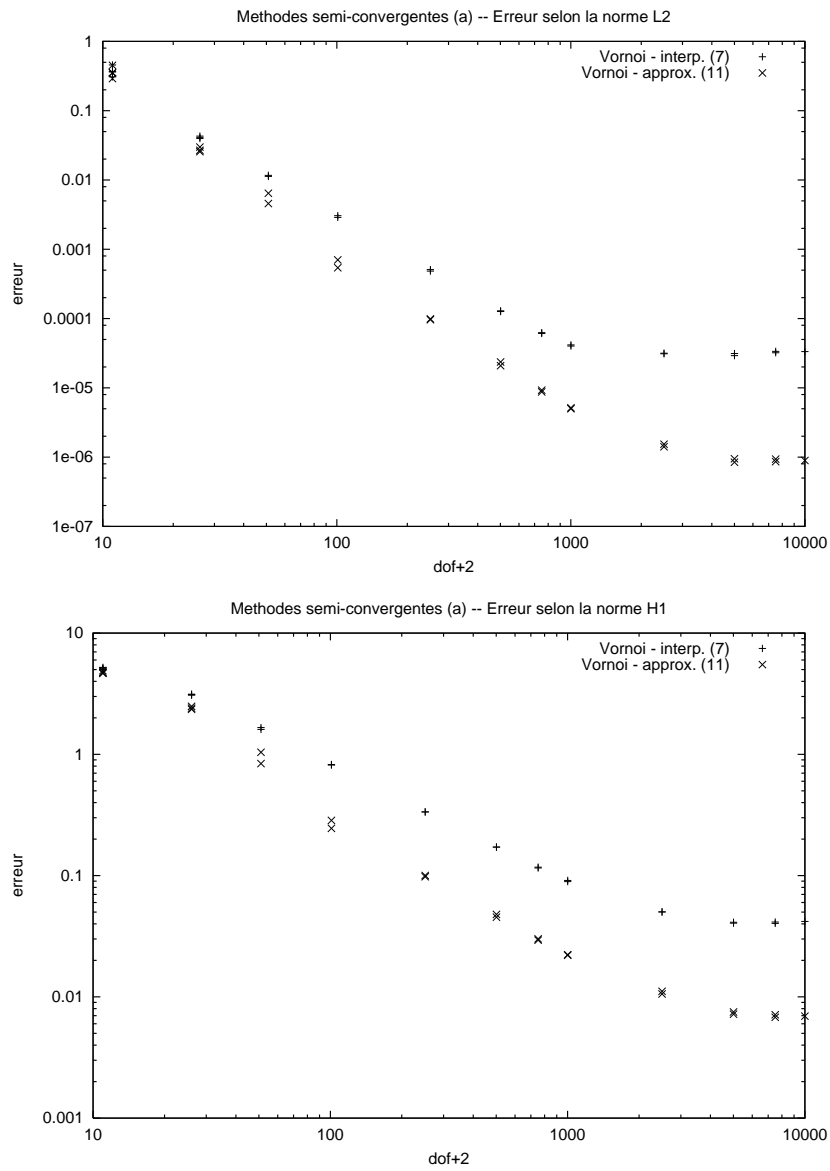


FIG. E.31: Méthodes 7 et 11 (Dérivée réelle, Gauss–Legendre) sur le problème 6. Analyse de la convergence.

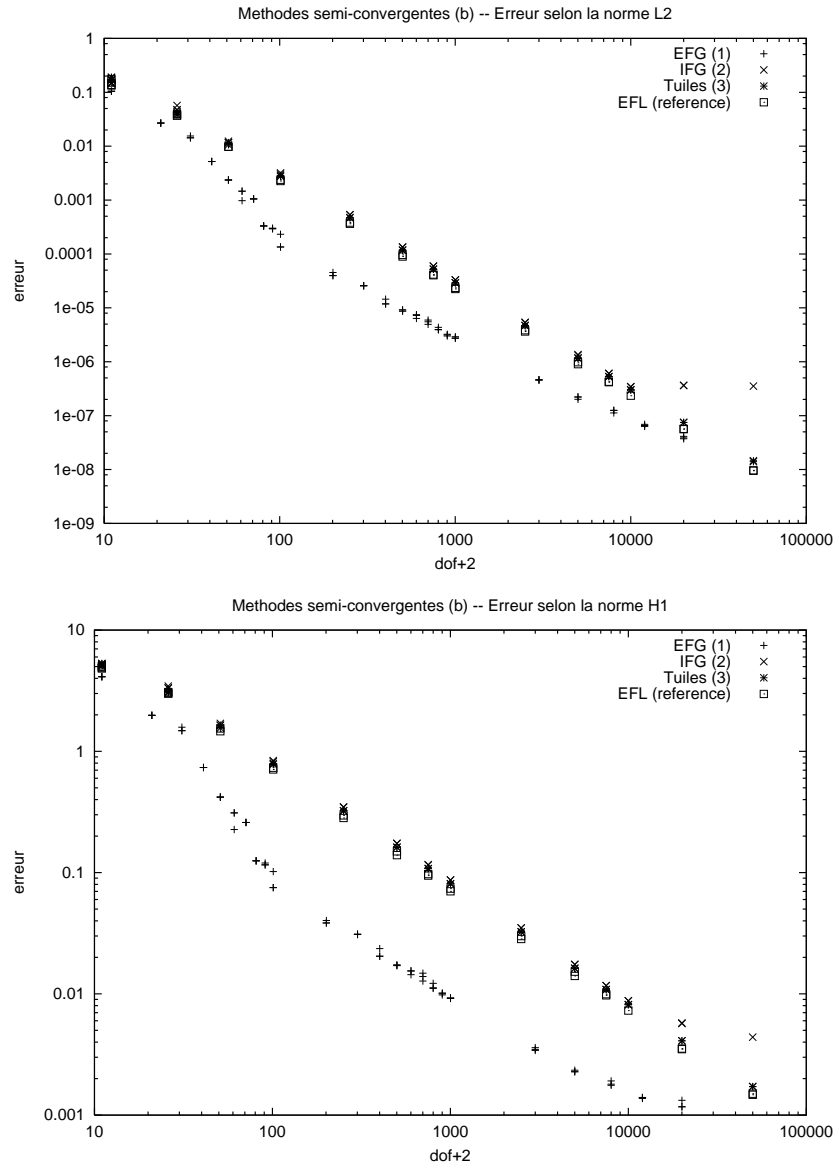


FIG. E.32: Méthodes 1, 2 et 3 (Dérivée réelle, Gauss-Legendre) sur le problème 6. Analyse de la convergence.

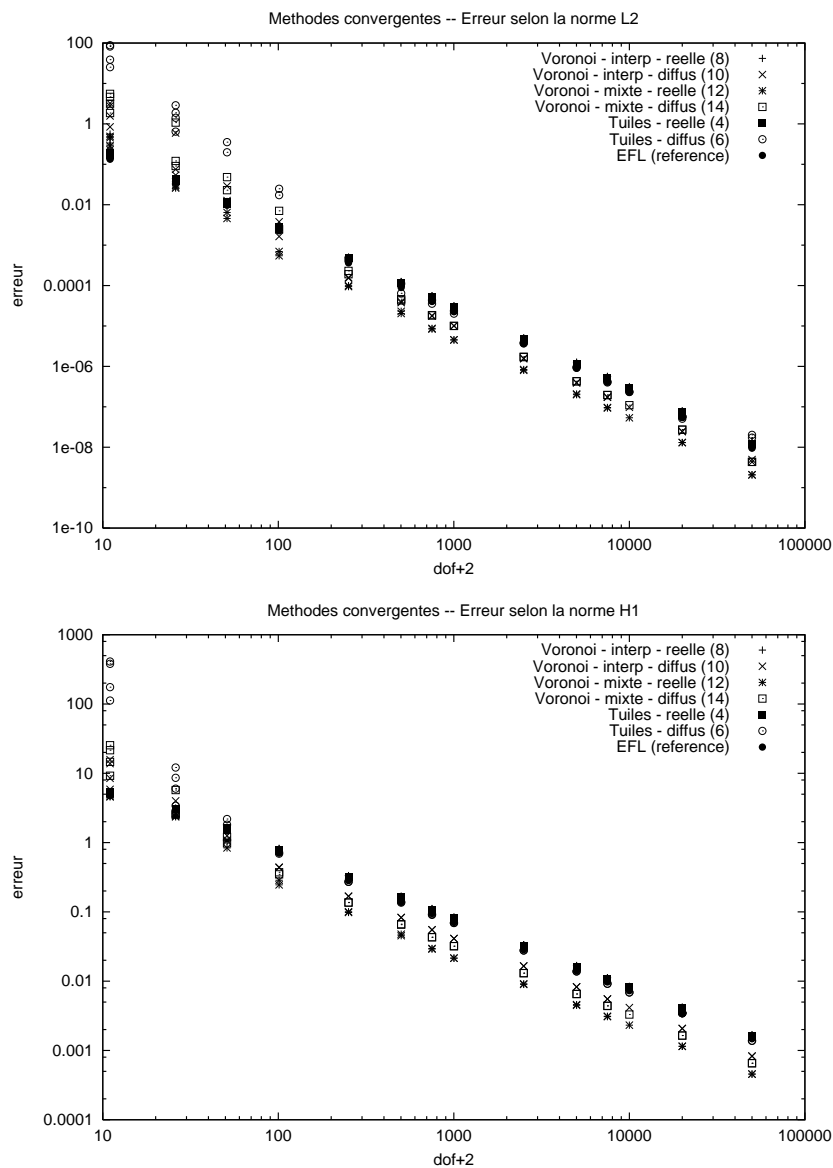


FIG. E.33: Méthodes convergentes sur le problème 6.

## E.2.8 Valeurs Propres

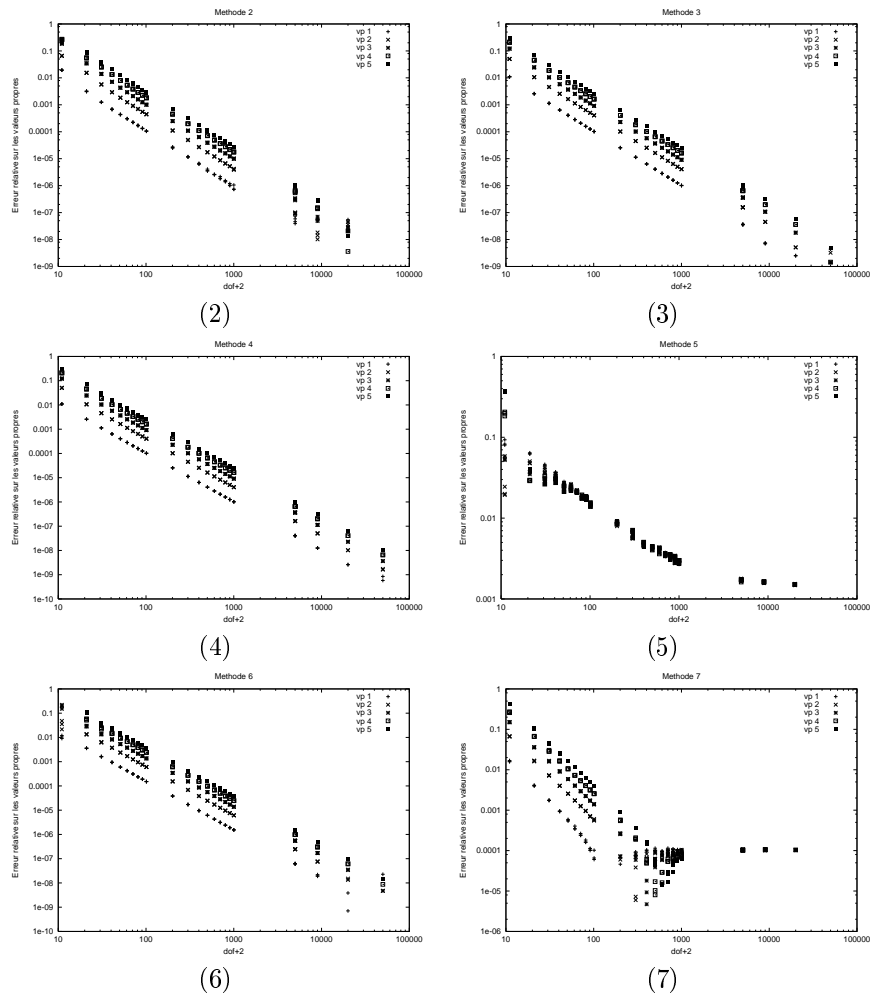


FIG. E.34: Erreurs relatives sur les cinq premières fréquences propres pour les méthodes 2 à 7.

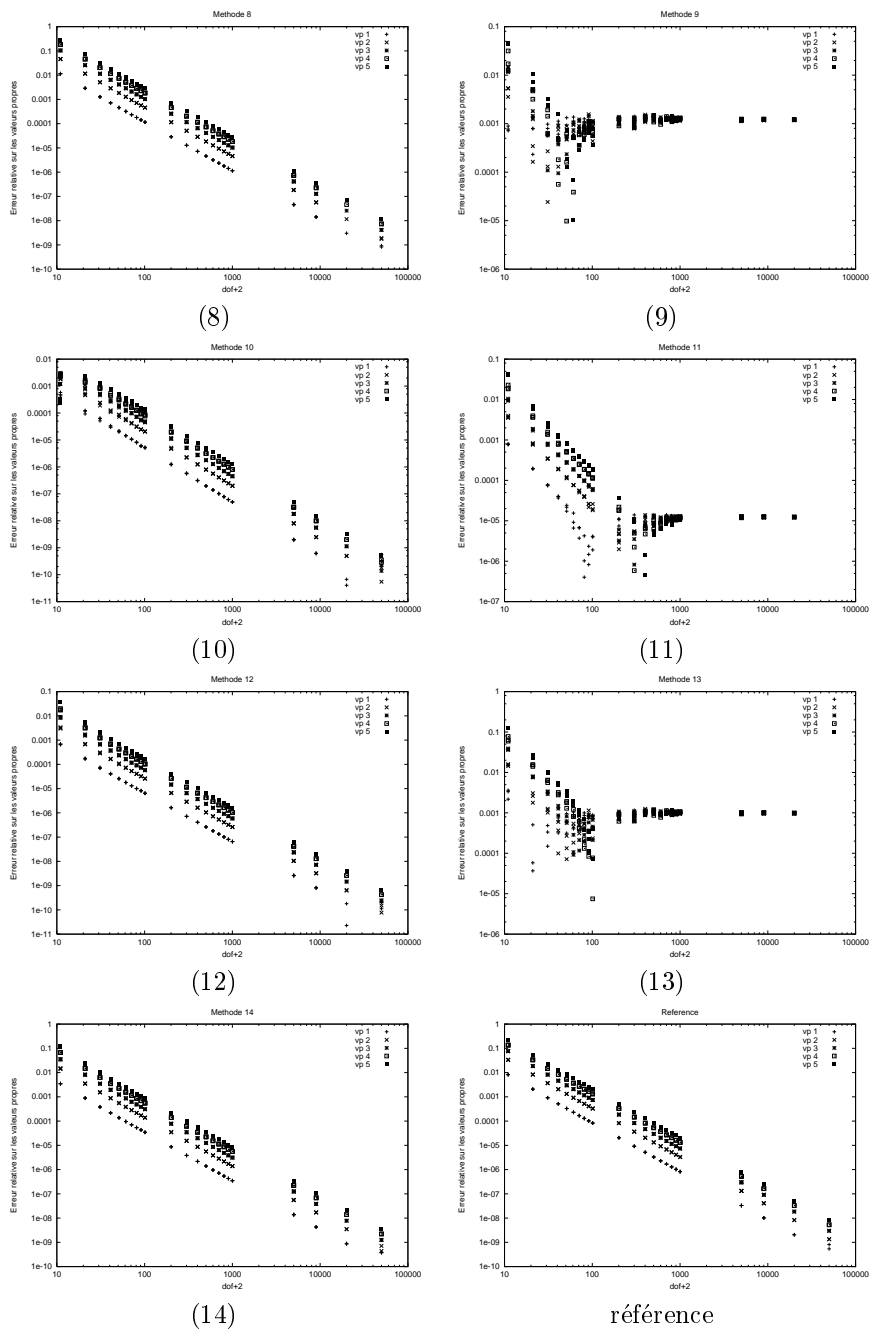


FIG. E.35: Erreurs relatives sur les cinq première fréquences propres pour les méthodes 8 à 14 et les éléments finis linéaires.





## Annexe F

# Analyse des résultats numériques du patch-test

Nous essayons ici de comprendre l'augmentation de l'erreur avec le nombre de nœuds observée pour le patch-test. Lorsqu'on résout un problème simple

$$\begin{cases} u'' = 0 \text{ sur } [1, l + 1] \\ u(0) = u_0 \\ u(l) = u_l \end{cases}$$

par élément finis linéaire ou méthode sans maillage construite avec l'intégration numérique adaptée, on obtient le type de courbe de la figure F.1 qui montre que pour ce problème (le patch-test) l'erreur augmente avec la discrétisation!

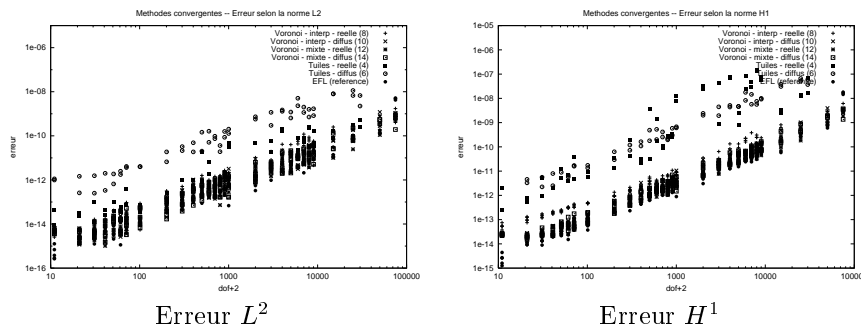


FIG. F.1: « Divergence » pour le patch test numérique (voir annexe précédente).

Comme la solution calculée est théoriquement exacte (car la solution est contenue dans l'espace de recherche et que nous utilisons l'intégration adaptée), l'augmentation de l'erreur est certainement due à des difficultés numériques liées au pas d'intégration  $h$ . Nous allons analyser le cas des éléments finis linéaires pour mieux comprendre ce phénomène.

### F.1 Introduction

Le pas de discrétisation  $h$  ( $= \frac{1}{n-1}$ ,  $n$  le nombre de nœuds) intervient dans plusieurs facteurs numériques et en particulier :

- Sur les positions des nœuds  $x_i = ih + \epsilon_i$  ;



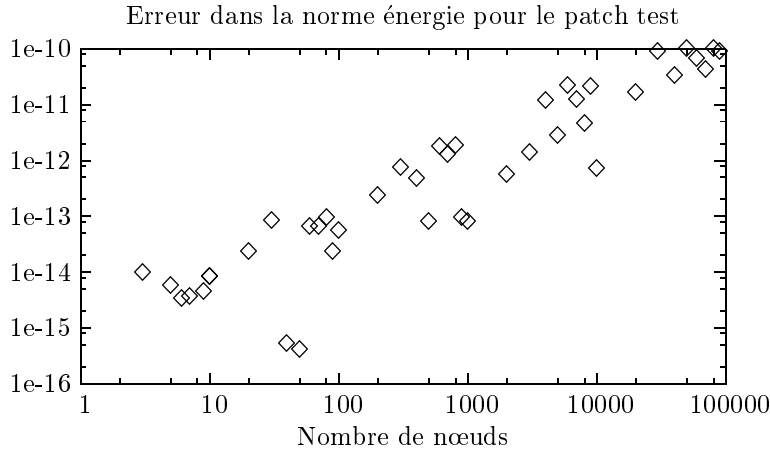


FIG. F.2: *Calcul de  $U^T KU$  dans le cas homogène.*

### F.2.2 Cas hétérogène

Cependant, la situation que nous venons de tester est particulièrement favorable et une partie des calculs nécessaires dans le cas général est faite « à la main ». Ceci concerne l'assemblage en tout premier lieu. Nous faisons le même calcul que précédemment, mais la matrice de raideur est calculée par assemblage des matrices élémentaires.

Nous pouvons tester le cas homogène (i.e.  $x_i = ih + 1$ ), comme le cas hétérogène. On trouve aussi une erreur qui augmente lorsque la discrétisation diminue.

### F.2.3 Test du solveur de système linéaire

Pour tester le solveur linéaire (routines LAPACK, `dgbco` et `dgbsl`), nous reprenons la matrice de raideur modifiée pour prendre en compte les conditions aux limites (pour avoir une matrice inversible). Nous calculons alors le produit  $KU$  dans  $F$  où  $U$  contient une solution linéaire et résolvons  $KU_{calc} = F$  pour différents nombres de nœuds. L'erreur est la norme  $l^2$  de  $U - U_{calc}$ .

La courbe figure F.4 montre aussi une erreur qui augmente avec la discrétisation.

## F.3 Analyse

### F.3.1 Norme énergie

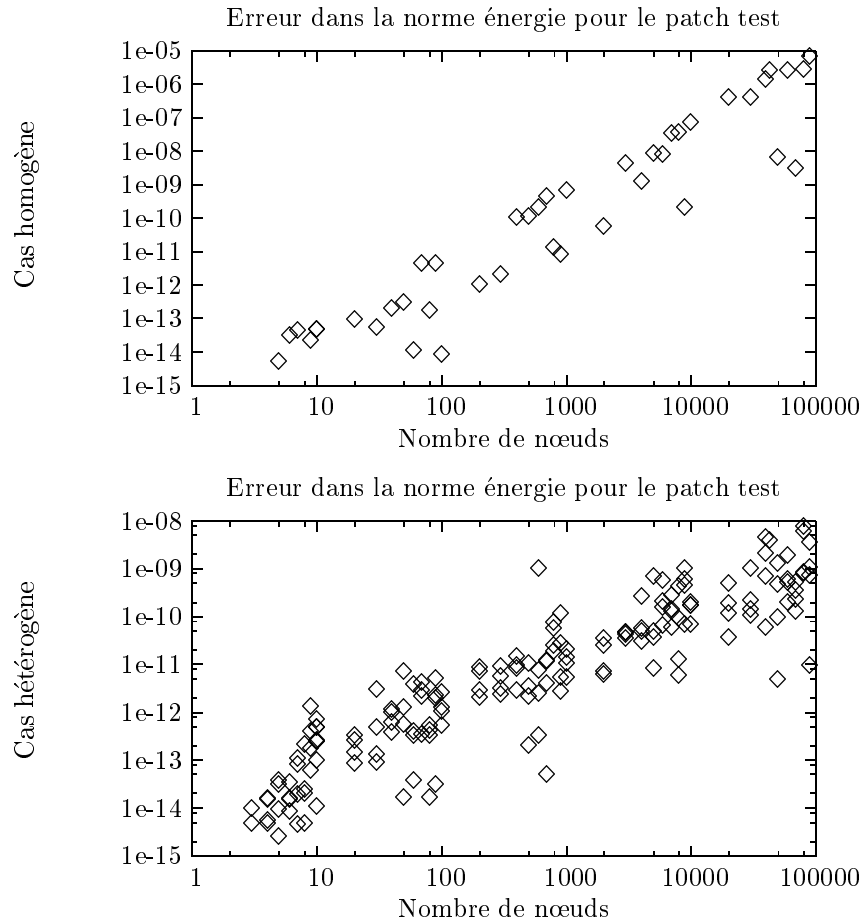
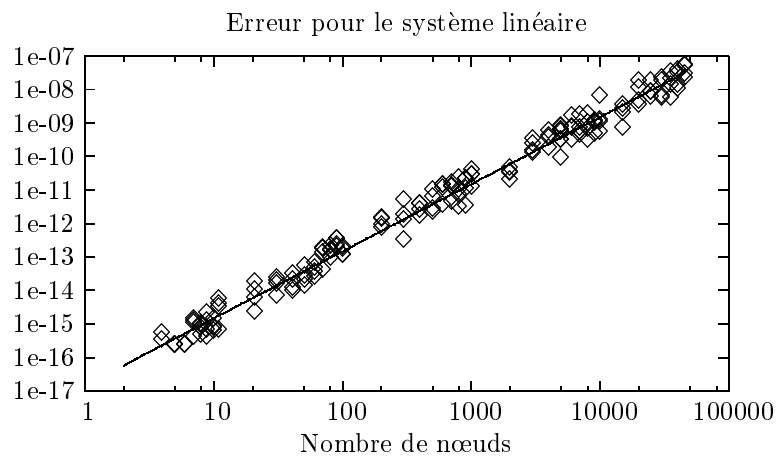
Nous nous plaçons ici dans le cas sans assemblage. Soit  $V = KU$ , ses coordonnées sont de la forme (on ne prend en compte que les nœuds intérieurs)

$$\begin{aligned} V_i &= \frac{2U_i - U_{i-1} - U_{i+1}}{h} \\ &= \frac{2b - b - b + 2aih - aih - aih + ah - ah}{h} \end{aligned}$$

Soit  $\delta$  une erreur de la magnitude de l'epsilon machine (i.e. le plus grand réel tel que  $1 + \epsilon = 1$  selon l'arithmétique de la machine, pour la double précision IEEE, on a  $\epsilon \approx 2.2210^{-16}$ ). On a

$$V_i \approx \frac{b\delta + aih\delta + a\delta h}{h}$$

Et donc l'erreur sur une coordonnée de  $V$  est proportionnelle à  $\frac{\epsilon}{h}$ , elle augmente lorsque  $h$  diminue. Ceci semble conforme à l'expérience.

FIG. F.3: *Calcul de  $U^T KU$  avec assemblage.*FIG. F.4: *Calcul de l'erreur sur la résolution du système linéaire.*

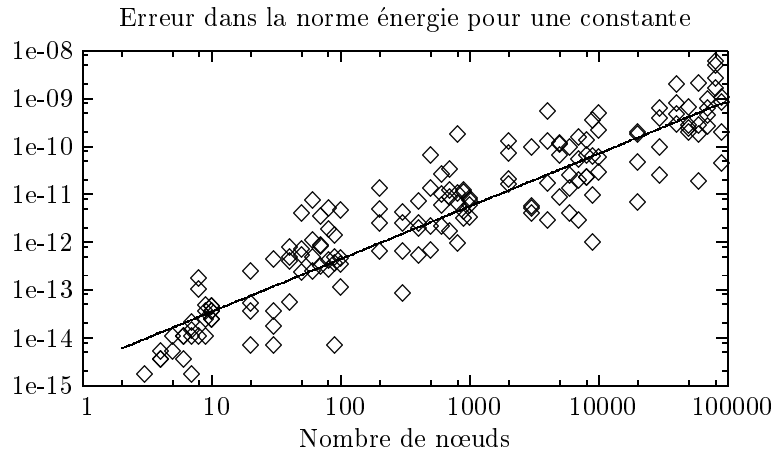


FIG. F.5: Erreur énergie avec une matrice assemblée pour le terme constant.

### F.3.2 Influence des paramètres

**Terme constant** D'après l'analyse précédente, l'erreur due au terme constant est  $\frac{b\delta}{h}$ . Les tests sans assemblage montrent une erreur nulle. Ce résultat n'est pas choquant car on fait alors du calcul « presque entier ». L'évolution de l'erreur avec le nombre de nœuds dans le cas d'une matrice de raideur assemblée est montrée figure F.5.

Dans ce cas, ce n'est pas l'erreur due au calcul de  $2b - b - b$ , mais à celle de l'assemblage de la matrice de raideur. Nous voyons apparaître une nouvelle source d'erreur numérique. La droite ajustée aux données dans le diagramme log – log montre cependant que l'erreur évolue bien en  $\frac{1}{h}$ .

**Terme linéaire** L'erreur due au terme linéaire est composée de deux parties: d'une part un terme constant en  $a\delta$  et d'autre part un terme qui varie avec la position du nœud en  $\frac{aih\delta}{h}$ . La seconde partie de l'erreur pour  $a$  proche de 1, varie donc en  $\frac{1}{h}$  pour les derniers termes du vecteur  $U$  (car on a alors  $ih \approx 1$ ). On doit donc avoir la même évolution pour l'erreur  $l^2$  que précédemment. La figure F.6 montre l'évolution de l'erreur pour une matrice de raideur calculée sans assemblage.

Pour vérifier notre hypothèse de répartition de l'erreur dans le vecteur  $U$  nous traçons  $|U_i|$  en fonction de  $i$  (calcul sans assemblage, figure F.7). La première droite d'erreurs non nulles s'écrit  $d(i) = \alpha i + \beta$ , les suivantes  $2d(i)$ ,  $3d(i)$ , ... Ceci confirme notre analyse (une erreur de *roundoff*, deux, trois, ...).

### F.3.3 Système linéaire

Le même type de raisonnement peut être mené pour la résolution du système linéaire. Je renvoie à la littérature spécialisée pour plus de détails. Notons que sur la courbe F.4 l'erreur augmente en  $\frac{1}{h^2}$ . Ceci est certainement du à la technique utilisée qui génère deux erreurs:

1. La première pour calculer  $KU$  (erreur en  $\frac{1}{h}$ )
2. La seconde pour calculer la solution du système linéaire.

La résolution elle-même présente donc une erreur qui évolue en  $\frac{1}{h}$  (les erreurs sont multiplicatives).

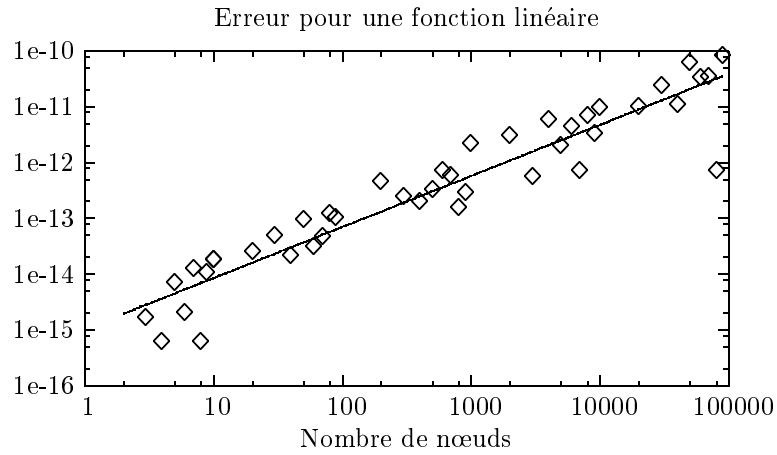


FIG. F.6: Erreur énergie avec une matrice assemblée pour une fonction linéaire.

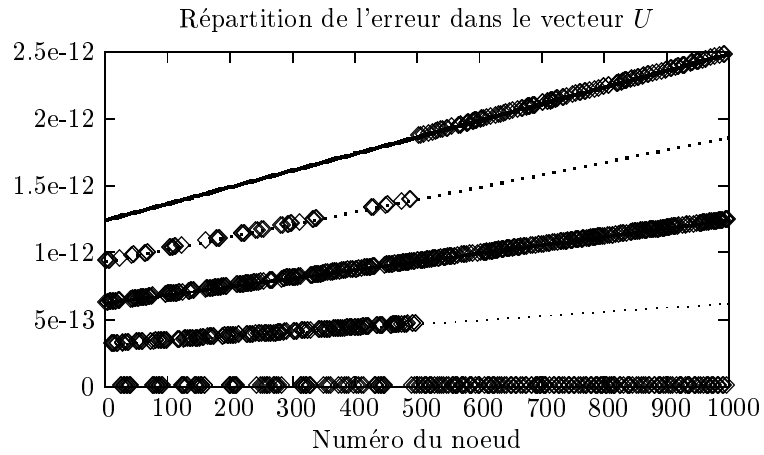


FIG. F.7: Répartition de l'erreur dans le vecteur  $U$ .

## F.4 Conclusion

Il est important de noter que les « taux de divergence » observés pour la norme énergie et la résolution de  $KU = F$  sont du même ordre que celui du patch test numérique. Nous pensons donc que ces deux facteurs sont essentiels pour l'expliquer. Il ne faut pas non plus négliger l'effet de l'assemblage qui agit principalement sur les constantes. L'utilisation d'une méthode itérative pour la résolution du système linéaire  $KU = F$  n'apporte (certainement) pas de solution car on évalue des produits  $KU$  (gradients conjugués).





## Annexe G

# Patch test pour une poutre de courbure et torsion constantes

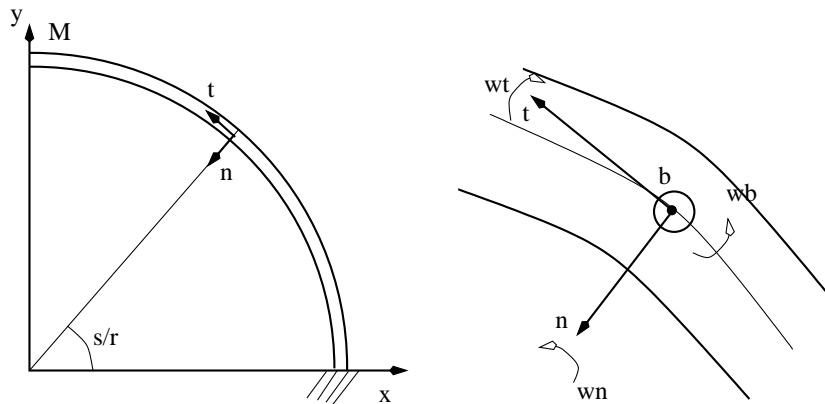
### G.1 Cas d'une poutre en arc de cercle

#### G.1.1 Problème posé

On considère une poutre annulaire ayant la forme d'un arc de cercle et on cherche l'état de contraintes constantes.

**Géométrie** Soit l'arc de cercle défini par (rayon  $r$ , courbure  $\rho = \frac{1}{r}$ , centre à l'origine)

$$\vec{x}(s) = \begin{pmatrix} r \cos(\rho s) \\ r \sin(\rho s) \\ 0 \end{pmatrix}$$



Le repère de Frenet sur le quart de cercle est défini par

$$\vec{t}(s) = \begin{pmatrix} -\sin(\rho s) \\ \cos(\rho s) \\ 0 \end{pmatrix}, \vec{n}(s) = \begin{pmatrix} -\cos(\rho s) \\ -\sin(\rho s) \\ 0 \end{pmatrix}, \vec{b}(s) = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

**Grandeur vectorielle sur le cercle** Soit  $\vec{u}$  un champ vectoriel défini sur le cercle de composantes  $U = \langle U_t, U_n, U_b \rangle^\top$  sur  $(\vec{t}, \vec{n}, \vec{b})$ . La propriété du repère de Frenet

$$\begin{Bmatrix} \vec{t}' \\ \vec{n}' \\ \vec{b}' \end{Bmatrix} = \begin{bmatrix} 0 & \rho & 0 \\ -\rho & 0 & \omega \\ 0 & -\omega & 0 \end{bmatrix} \begin{Bmatrix} \vec{t} \\ \vec{n} \\ \vec{b} \end{Bmatrix}$$

implique dans le cas particulier où le champ  $\vec{u}$  est uniforme que

$$\begin{cases} U_t' - \rho U_n & = 0 \\ U_n' + \rho U_t & = 0 \\ U_b' & = 0 \end{cases}$$

### G.1.2 Solution analytique

**Moment** La première condition est  $\vec{M}' = \vec{0}$ , le vecteur  $\langle M_t, M_n, M_b \rangle^\top$  est donc solution du système différentiel

$$\begin{cases} M_t' - \rho M_n & = 0 \\ M_n' + \rho M_t & = 0 \\ M_b' & = 0 \end{cases}$$

La solution est donc

$$\begin{cases} M_t(s) & = M_1 \cos(\rho s) + M_2 \sin(\rho s) \\ M_n(s) & = M_2 \cos(\rho s) - M_1 \sin(\rho s) \\ M_b(s) & = M_3 \end{cases} \quad (\text{G.1})$$

et les constantes  $M_1, M_2, M_3$  sont définies par le chargement à l'extrémité de la poutre  $(+M_2, -M_1, M_3)$  (en coordonnées cartésiennes). Comme la courbure est non nulle, on a aussi  $\mathcal{T} = \vec{0}$ .

**Rotation de la section droite** De la même façon que précédemment, on écrit le système différentiel des coordonnées de  $\vec{\omega}, \Omega$  dans la base de Frenet

$$\begin{cases} EI(\Omega_t' - \rho\Omega_n) & = (1 + \nu)M_t \\ EI(\Omega_n' + \rho\Omega_t) & = M_n \\ EI\Omega_b' & = M_3 \end{cases}$$

On reconnaît la même équation différentielle que précédemment mais avec un second membre: faisons varier la constante

$$\begin{cases} \Omega_1'(s) \cos(\rho s) + \Omega_2'(s) \sin(\rho s) & = \frac{1+\nu}{EI} \left\{ M_1 \cos(\rho s) + M_2 \sin(\rho s) \right\} \\ \Omega_2'(s) \cos(\rho s) - \Omega_1'(s) \sin(\rho s) & = \frac{1}{EI} \left\{ M_2 \cos(\rho s) - M_1 \sin(\rho s) \right\} \end{cases}$$

d'où on tire

$$\begin{cases} \Omega_1'(s) & = \frac{M_1}{EI} + \frac{\nu}{EI} \left\{ M_1 \cos^2(\rho s) + M_2 \sin(\rho s) \cos(\rho s) \right\} \\ \Omega_2'(s) & = \frac{M_2}{EI} + \frac{\nu}{EI} \left\{ M_1 \sin(\rho s) \cos(\rho s) + M_2 \sin^2(\rho s) \right\} \end{cases}$$

pour intégrer on utilise les formules de trigonométrie des arcs doubles

$$\cos^2(\rho s) = \frac{1 + \cos(2\rho s)}{2}, \quad \sin^2(\rho s) = \frac{1 - \cos(2\rho s)}{2}, \quad \cos(\rho s) \sin(\rho s) = \frac{\sin(2\rho s)}{2}$$

pour obtenir

$$\begin{cases} \Omega_1(s) = \Omega_1^0 + \frac{M_1}{EI}s + \frac{\nu}{EI} \left\{ M_1 \left[ \frac{s}{2} + \frac{\sin(2\rho s)}{4\rho} \right] - M_2 \frac{\cos(2\rho s) - 1}{4\rho} \right\} \\ \Omega_2(s) = \Omega_2^0 + \frac{M_2}{EI}s + \frac{\nu}{EI} \left\{ -M_1 \frac{\cos(2\rho s) - 1}{4\rho} + M_2 \left[ \frac{s}{2} - \frac{\sin(2\rho s)}{4\rho} \right] \right\} \end{cases}$$

On injecte maintenant dans  $\Omega_t, \Omega_n$  pour retrouver les arcs simples

$$\begin{cases} \Omega_t(s) = \left( \Omega_1^0 + \frac{M_1}{EI}s + \frac{\nu M_1}{EI} \frac{s}{2} + \frac{\nu M_2}{4EI\rho} \right) \cos(\rho s) \\ \quad + \left( \Omega_2^0 + \frac{M_2}{EI}s + \frac{\nu M_2}{EI} \frac{s}{2} + \frac{\nu M_1}{4EI\rho} \right) \sin(\rho s) \\ \quad + \frac{\nu}{4EI\rho} \left\{ M_1 \underbrace{\left( \sin(2\rho s) \cos(\rho s) - \cos(2\rho s) \sin(\rho s) \right)}_{\sin(2\rho s - \rho s)} \right. \\ \quad \quad \left. - M_2 \underbrace{\left( \cos(2\rho s) \cos(\rho s) + \sin(2\rho s) \sin(\rho s) \right)}_{\cos(2\rho s - \rho s)} \right\} \\ \Omega_n(s) = \left( \Omega_2^0 + \frac{M_2}{EI}s + \frac{\nu M_1}{4EI\rho} + \frac{\nu M_2}{EI} \frac{s}{2} \right) \cos(\rho s) \\ \quad - \left( \Omega_1^0 + \frac{M_1}{EI}s + \frac{\nu M_1}{EI} \frac{s}{2} + \frac{\nu M_2}{4EI\rho} \right) \sin(\rho s) \\ \quad + \frac{\nu}{4EI\rho} \left\{ -M_1 \underbrace{\left( \cos(2\rho s) \cos(\rho s) + \sin(2\rho s) \sin(\rho s) \right)}_{\cos(2\rho s - \rho s)} \right. \\ \quad \quad \left. + M_2 \underbrace{\left( \cos(2\rho s) \sin(\rho s) - \sin(2\rho s) \cos(\rho s) \right)}_{\sin(\rho s - 2\rho s)} \right\} \end{cases}$$

Et tous calculs faits, on obtient la rotation de la section droite (note: la condition à l'extrémité  $\vec{\omega}(0) = \vec{0}$  impose  $\Omega_1^0 = \Omega_2^0 = 0$ )

$$\begin{cases} \Omega_t(s) = \frac{(1+\nu/2)M_1}{EI}s \cos(\rho s) + \left( \frac{\nu M_1}{2EI\rho} + \frac{(1+\nu/2)M_2}{EI} \right) s \sin(\rho s) \\ \Omega_n(s) = \frac{(1+\nu/2)M_2}{EI}s \cos(\rho s) - \left( \frac{\nu M_2}{2EI\rho} + \frac{(1+\nu/2)M_1}{EI} \right) s \sin(\rho s) \\ \Omega_b(s) = sM_3 \end{cases} \quad (G.2)$$

**Déplacements** Comme  $\rho \neq 0$  et  $\vec{u}' \cdot \vec{t} = 0$  (car  $\vec{T} \cdot \vec{t} = 0$ ) les déplacements sont solutions de  $\vec{u}' = -\vec{t} \wedge \vec{\omega}$ , soit

$$\begin{cases} U_t' - \rho U_n = 0 \\ U_n' + \rho U_t = -\Omega_b = -sM_3 \\ U_b' = \Omega_n \end{cases}$$

Calculons pour commencer le couple  $(U_t, U_n)$ , la solution homogène du système différentiel est

$$\begin{cases} U_t(s) = U_1 \cos(\rho s) + U_2 \sin(\rho s) \\ U_n(s) = U_2 \cos(\rho s) - U_1 \sin(\rho s) \end{cases}$$

et une solution particulière

$$\begin{cases} U_t(s) = -\frac{M_3}{\rho} s \\ U_n(s) = -\frac{M_3}{\rho^2} \end{cases}$$

avec les conditions aux limites, il vient finalement

$$\begin{cases} U_t(s) = \frac{M_3}{\rho^2} (\sin(\rho s) - \rho s) \\ U_n(s) = \frac{M_3}{\rho^2} (\cos(\rho s) - 1) \end{cases}$$

Le calcul de  $U_b$  utilise les primitives suivantes (par intégration par parties)

$$\begin{cases} \int x \sin(\omega x) dx &= \frac{\sin(\omega x)}{\omega^2} - \frac{x \cos(\omega x)}{\omega} \\ \int x \cos(\omega x) dx &= \frac{\cos(\omega x)}{\omega^2} + \frac{x \sin(\omega x)}{\omega} \end{cases}$$

Et finalement les déplacements projetés sur  $(\vec{t}, \vec{n}, \vec{b})$  sont donnés par

$$\begin{cases} U_t(s) &= \frac{M_3}{\rho^2} (\sin(\rho s) - \rho s) \\ U_n(s) &= \frac{M_3}{\rho^2} (\cos(\rho s) - 1) \\ U_b(s) &= \left( \frac{(1+\nu)M_2}{EI\rho^2} + \frac{(1+\nu/2)M_1}{EI\rho} s \right) \cos(\rho s) \\ &+ \left( \frac{(1+\nu/2)M_2}{EI\rho} s - \frac{(1+\nu/2)M_1}{EI\rho^2} \right) \sin(\rho s) \\ &- \frac{(1+\nu)M_2}{EI\rho^2} \end{cases} \quad (\text{G.3})$$

### G.1.3 Conclusion

Pour que les états de contraintes constantes sur une poutre en arc de cercle soient reproduits exactement par une interpolation, l'espace de recherche doit contenir les fonctions suivantes pour l'interpolation de  $\vec{\omega}$  et  $U$  (i.e. les coordonnées de  $\vec{\omega}$ ,  $\vec{u}$  dans la base de Frenet)

$$1, s, \cos(\rho s), \sin(\rho s), s \cos(\rho s), s \sin(\rho s).$$

## G.2 Cas d'une poutre en hélice

On suppose dans la suite  $\nu = 0$  pour simplifier les calculs.

### G.2.1 Géométrie

Une hélice est une courbe de courbure  $\rho$  et torsion  $\tau$  constantes. Soit  $\vec{x}_0, \vec{t}_0, \vec{n}_0, \vec{b}_0$  le repère de Frenet en un point initial. La courbe est donnée par

$$\gamma \vec{x}(s) = [A^2 \sin(\gamma s) + B^2 \gamma s] \vec{t}_0 + A[1 - \cos(\gamma s)] \vec{n}_0 + AB[\gamma s - \sin(\gamma s)] \vec{b}_0 + \gamma \vec{x}_0$$

avec

$$A = \frac{\rho}{\gamma}, \quad B = \frac{\rho}{\gamma}, \quad \gamma = \sqrt{\rho^2 + \tau^2}$$

on pose aussi dans la suite

$$\begin{cases} \vec{e} = \tau \vec{t} + \rho \vec{b} \\ \vec{f} = -\rho \vec{t} + \tau \vec{b} \\ \sigma = \gamma s \end{cases}$$

(note: le vecteur  $\vec{e}$  joue un rôle important dans la géométrie de l'hélice.).  $(\vec{e}, \vec{n}, \vec{f})$  est une base.

### G.2.2 Résultat

Nous nous contentons dans ce cas de donner les résultats sans démonstration. Le moment constant est (l'état de contraintes est caractérisé par  $\vec{T} = \vec{0}$  et  $\vec{M}$  constant)

$$\vec{M} = M_e \vec{e}_0 + M_n \vec{n}_0 + M_f \vec{f}_0$$

Les déplacements et rotations de la section droite pour cet état de contraintes sont donnés par

$$\begin{cases} \gamma EI \Omega_t &= \tau \sigma M_e + \rho \sigma \sin(\sigma) M_n - \rho \sigma \cos(\sigma) M_f \\ \gamma EI \Omega_n &= \sigma \cos(\sigma) M_n + \sigma \sin(\sigma) M_f \\ \gamma EI \Omega_b &= \rho M_e - \tau \sin(\sigma) + \tau \cos(\sigma) \end{cases} \quad (\text{G.4})$$

et

$$\begin{cases} \gamma^2 EIU_t &= [\tau^2(\sigma - \sin(\sigma))]M_e \\ &+ [\tau\rho(1 - \cos(\sigma) - \sigma \sin(\sigma) + \frac{1}{2}\sigma^2 \cos(\sigma))]M_n \\ &+ [\tau\rho(\sigma \cos(\sigma) - \sin(\sigma) + \frac{1}{2}\sigma^2 \sin(\sigma))]M_f \\ \gamma^2 EIU_n &= [\rho(1 - \cos(\sigma))]M_e \\ &- \frac{\tau}{2}\sigma^2 \sin(\sigma)M_n \\ &+ \frac{\tau}{2}\sigma^2 \cos(\sigma)M_f \\ \gamma^2 EIU_b &= [\tau\rho(\sin(\sigma) - \sigma)]M_e \\ &+ [\rho^2(1 - \cos(\sigma) - \sigma \sin(\sigma)) - \frac{\tau^2}{2}\sigma^2 \cos(\sigma)]M_n \\ &+ [\rho^2(\sigma \cos(\sigma) - \sin(\sigma)) - \frac{\tau^2}{2}\sigma^2 \sin(\sigma)]M_f \end{cases} \quad (\text{G.5})$$

### G.2.3 Conclusion

Pour que les états de contraintes constantes sur une poutre en arc d'hélice soient reproduits exactement par une approximation, l'espace de recherche doit contenir les fonctions suivantes pour l'interpolation de  $\Omega$  et  $U$

$$1, s, s^2, \cos(\gamma s), \sin(\gamma s), s \cos(\gamma s), s \sin(\gamma s), s^2 \cos(\gamma s), s^2 \sin(\gamma s).$$



## Annexe H

# Approximation diffuse trigonométrique

### H.1 Approximation diffuse – cas général

Soient des données  $(u_i)_{i \in I}$  aux points  $(x_i)_{i \in I}$  dans un intervalle  $]a, b[$ . On associe au point  $x_i$  la fonction de pondération  $w_i$ , et on définit  $I(x) = \{i \in I, w_i(x) \neq 0\}$ . On suppose dans la suite la propriété suivante vérifiée

$$\forall x \in ]a, b[, n_x = \text{card}(I(x)) > m$$

Soient des fonctions  $(f_l)_{l \in [1, m]}$  linéairement indépendantes sur  $]a, b[$  (cette propriété doit être vraie au sens **fort**, i.e. linéairement indépendantes sur les  $(x_i)_{i \in I}$ ).

Soit  $\bar{u}_x(t) = \sum_l a_l(x) f_l(t)$  où les  $(a_l)_l$  sont calculés par minimisation de l'erreur quadratique pondérée

$$\mathcal{E}_x(\{a\}) = \sum_{i \in I(x)} w_i(x) (u_i - \bar{u}_x(x_i))^2 \quad (\text{H.1})$$

L'approximation diffuse est définie par

$$\tilde{u}(x) = \bar{u}_x(x), \text{ avec } \{a\} \text{ minimisant } \mathcal{E}_x \quad (\text{H.2})$$

Si on pose

$$P(x) = \begin{bmatrix} f_1(x_{i_1}) & \dots & f_l(x_{i_1}) \\ \vdots & & \vdots \\ f_1(x_{i_{n_x}}) & \dots & f_l(x_{i_{n_x}}) \end{bmatrix}, \quad W(x) = \text{diag}(\{w_i(x)\}_{i \in I(x)})$$

alors le vecteur  $\{a\}$  est solution du système

$$\begin{cases} A(x)\{a(x)\} = b(x) \\ A(x) = P^\top(x)W(x)P(x) \\ b(x) = P^\top(x)W(x)U \end{cases} \quad (\text{H.3})$$

On définit bien entendu la pseudo-dérivée de l'approximation diffuse par (et par analogie)

$$\frac{\delta^k \tilde{u}}{\delta x^k}(x) = \bar{u}_x^{(k)}(x) \quad (\text{H.4})$$

L'approximation diffuse construite ainsi admet les propriétés usuelles (continuité, consistance), c'est-à-dire:

- elle est aussi régulière que la moins régulière des fonctions  $w_i, f_l$  ;



- elle reproduit exactement toute combinaison linéaire des fonctions  $f_l$ .
- interpolation avec les fonctions d'atténuation singulières ;
- elle admet une construction « en fonction de forme ».

Penchons-nous plus particulièrement sur ce dernier point.

Soient les fonctions  $N_i$  définies en tout point  $x$  par

$$\left\{ \begin{array}{l} \min_{N_i} \sum_{i \in I(x)} \frac{N_i^2}{w_i(x)} \\ \sum_{i \in I(x)} N_i(x) f_l(x) = f_l(x_i), \forall l \in [1, m] \end{array} \right.$$

les multiplicateurs de Lagrange associés aux contraintes de consistance sont notés  $\mu_l$ , et les **fonctions de forme**  $N_i$  sont solutions du système

$$\left[ \begin{array}{cc} W(x)^{-1} & P(x) \\ P^\top(x) & 0 \end{array} \right] \left\{ \begin{array}{l} N(x) \\ \{\mu_l(x)\} \end{array} \right\} = \left\{ \begin{array}{l} 0 \\ \{f_l(x)\} \end{array} \right\} \quad (\text{H.5})$$

Et on démontre par la technique habituelle que l'approximation diffuse s'écrit

$$\tilde{u}(x) = \sum_{i \in I(x)} N_i(x) u_i \quad (\text{H.6})$$

On garde les mêmes propriétés avec les pseudo-dérivées des fonctions de forme. L'une des propriétés importantes de l'approximation diffuse « polynomiale » est la construction centrée qui est une étape importante pour démontrer la convergence et obtenir des méthodes de calcul efficaces des fonctions de forme. Nous devons cependant spécifier les fonctions  $f_l$  pour obtenir ce genre de résultat.

## H.2 Approximation diffuse « trigonométrique »

Nous utiliserons dans la suite la base de fonction suivante qui est composée de polynômes et de fonctions trigonométriques :

$$1, x, \frac{x^2}{2}, \frac{x^3}{6}, \dots, \frac{x^k}{k!}, \cos(\omega x), \sin(\omega x) \dots \frac{x^l}{l!} \cos(\omega x), \frac{x^l}{l!} \sin(\omega x) \quad (\text{H.7})$$

Les conditions de consistance de la construction en fonctions de forme s'écrivent

$$\left\{ \begin{array}{l} \sum_{i \in I(x)} N_i(x) = 1 \\ \vdots \\ \sum_{i \in I(x)} N_i(x) \frac{x_i^k}{k!} = \frac{x^k}{k!} \\ \sum_{i \in I(x)} N_i(x) \cos(\omega x_i) = \cos(\omega x) \\ \sum_{i \in I(x)} N_i(x) \sin(\omega x_i) = \sin(\omega x) \\ \vdots \\ \sum_{i \in I(x)} N_i(x) \frac{x_i^k}{k!} \cos(\omega x_i) = \frac{x^k}{k!} \cos(\omega x) \\ \sum_{i \in I(x)} N_i(x) \frac{x_i^k}{k!} \sin(\omega x_i) = \frac{x^k}{k!} \sin(\omega x) \end{array} \right.$$

On sait que les contraintes polynomiales sont équivalentes aux contraintes centrées. Étudions le cas des contraintes trigonométriques, la forme centrée s'écrit

$$\left\{ \begin{array}{l} \sum_{i \in I(x)} N_i(x) \cos(\omega(x - x_i)) = 1 \\ \sum_{i \in I(x)} N_i(x) \sin(\omega(x - x_i)) = 0 \\ \vdots \\ \sum_{i \in I(x)} N_i(x) \frac{(x-x_i)^l}{l!} \cos(\omega(x - x_i)) = 0 \\ \sum_{i \in I(x)} N_i(x) \frac{(x-x_i)^l}{l!} \sin(\omega(x - x_i)) = 0 \end{array} \right.$$

À l'ordre zéro, il vient

$$\left\{ \begin{array}{l} \sin(\omega x) \sum_{i \in I(x)} N_i(x) \sin(\omega x_i) + \cos(\omega x) \sum_{i \in I(x)} N_i(x) \cos(\omega x_i) = \delta_0^j \\ \cos(\omega x) \sum_{i \in I(x)} N_i(x) \sin(\omega x_i) - \sin(\omega x) \sum_{i \in I(x)} N_i(x) \cos(\omega x_i) = 0 \end{array} \right.$$

soit

$$\left\{ \begin{array}{l} \sum_{i \in I(x)} N_i(x) \cos(\omega x_i) = \cos(\omega x) \\ \sum_{i \in I(x)} N_i(x) \sin(\omega x_i) = \sin(\omega x) \end{array} \right.$$

qui est la forme non centrée des contraintes trigonométriques. La démonstration se poursuit par récurrence pour les degrés 1, ..., l. Ainsi le système de construction de l'approximation diffuse « trigonométrique » s'écrit de façon équivalente sous forme centrée :

$$\left\{ \begin{array}{l} \min_{N_i} \sum_{i \in I(x)} \frac{N_i}{w_i(x)} \\ \left\{ \begin{array}{l} \sum N_i(x) = 1 \\ \sum N_i(x) \frac{(x-x_i)^j}{j!} = 0 \quad j \in [1, k] \\ \sum N_i(x) \frac{(x-x_i)^j}{j!} \cos(\omega(x - x_i)) = \delta_0^j \quad j \in [0, l] \\ \sum N_i(x) \frac{(x-x_i)^j}{j!} \sin(\omega(x - x_i)) = 0 \quad j \in [0, l] \end{array} \right. \end{array} \right. \quad (\text{H.8})$$

Cette forme est cependant différente de la forme usuelle où l'on a **un seul** terme non nul au second membre. De plus, si  $\omega = 1$  et  $x_i = 2i\pi$ , alors la contrainte issue du cosinus est  $\sum N_i = 1$  et celle des sinus inopérante  $0 = 0$ . La présence des fonctions trigonométriques n'apporte dans ce cas aucune amélioration et pose des problèmes d'existence. De la même façon, lorsque l'on est proche des positions de résonance, le problème numérique est mal conditionné.

Par exemple, si la base est  $(1, x, \sin(kx))$  et les nœuds  $(-\pi, 0, \pi)$ , alors la matrice  $A(x)$  est

$$A(x) = w_1(x) \begin{bmatrix} 1 & -\pi & 0 \\ -\pi & \pi^2 & 0 \\ 0 & 0 & 0 \end{bmatrix} + w_2(x) \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + w_3(x) \begin{bmatrix} 1 & \pi & 0 \\ \pi & \pi^2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Sa dernière ligne est nulle, elle est singulière.



## Annexe I

# Convergence des éléments diffus Hermite

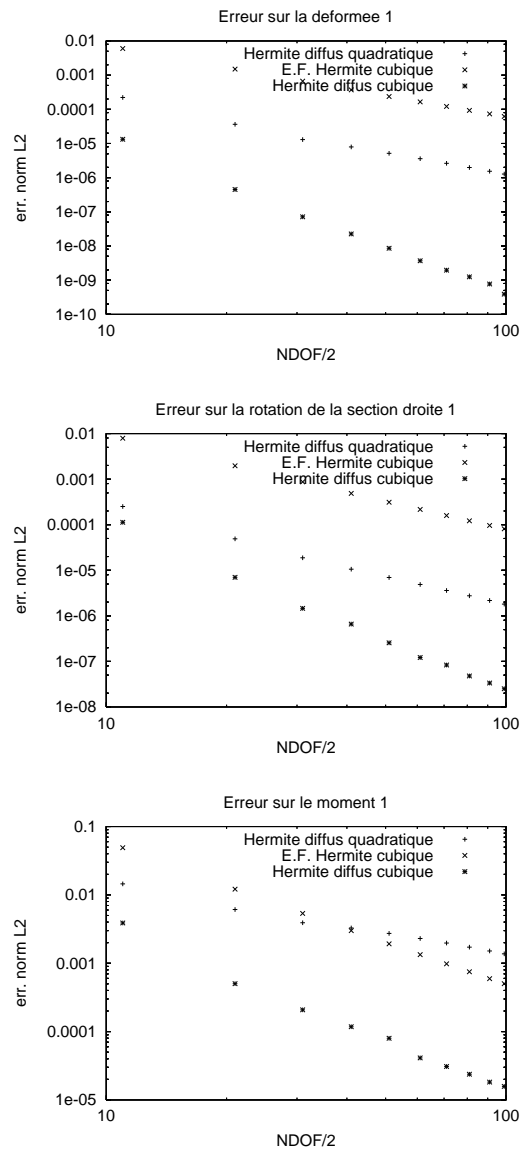


FIG. I.1: Convergence des éléments diffus Hermite pour le problème 1.

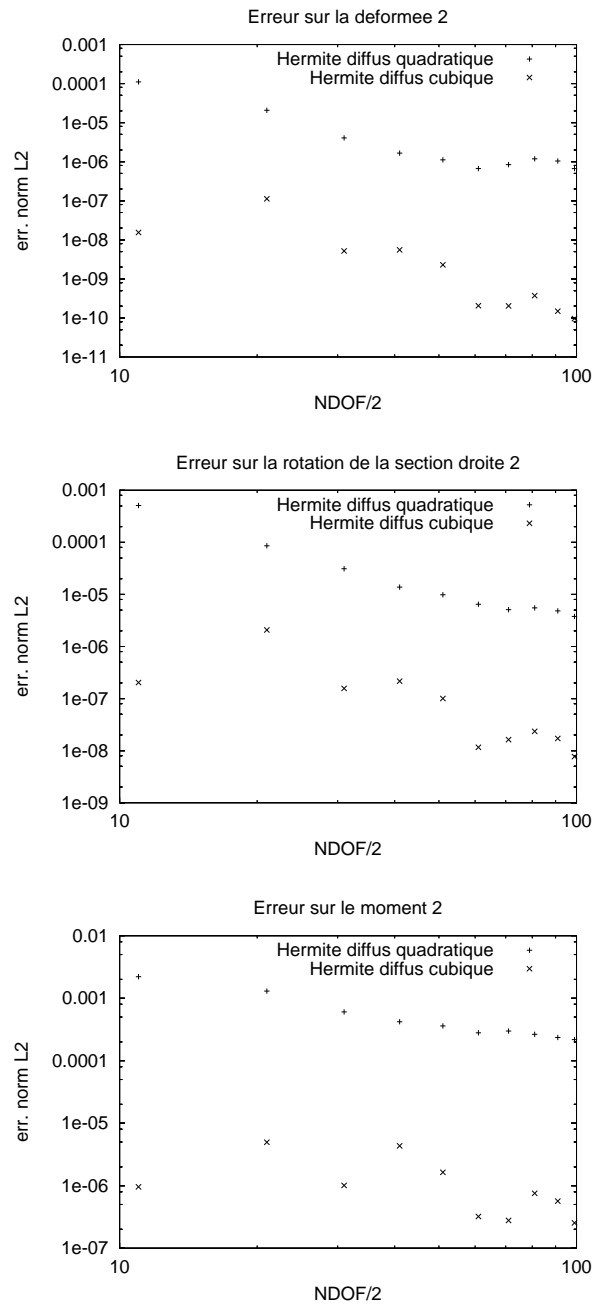


FIG. I.2: Convergence des éléments diffus Hermite pour le problème 2.

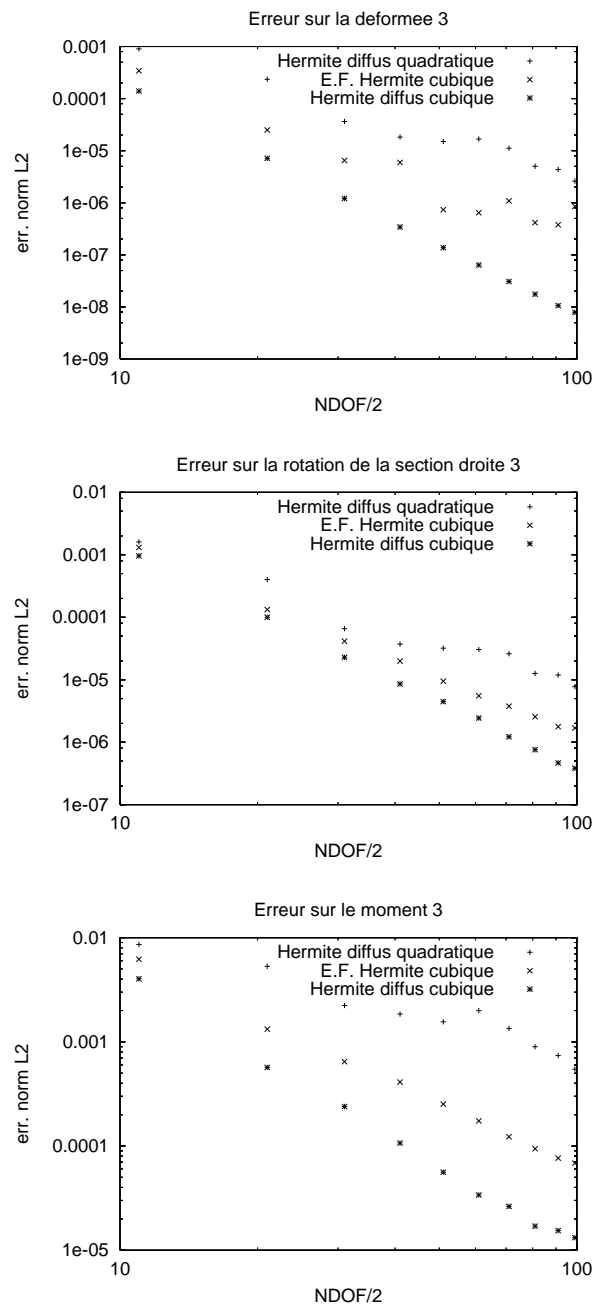


FIG. I.3: Convergence des éléments diffus Hermite pour le problème 3.

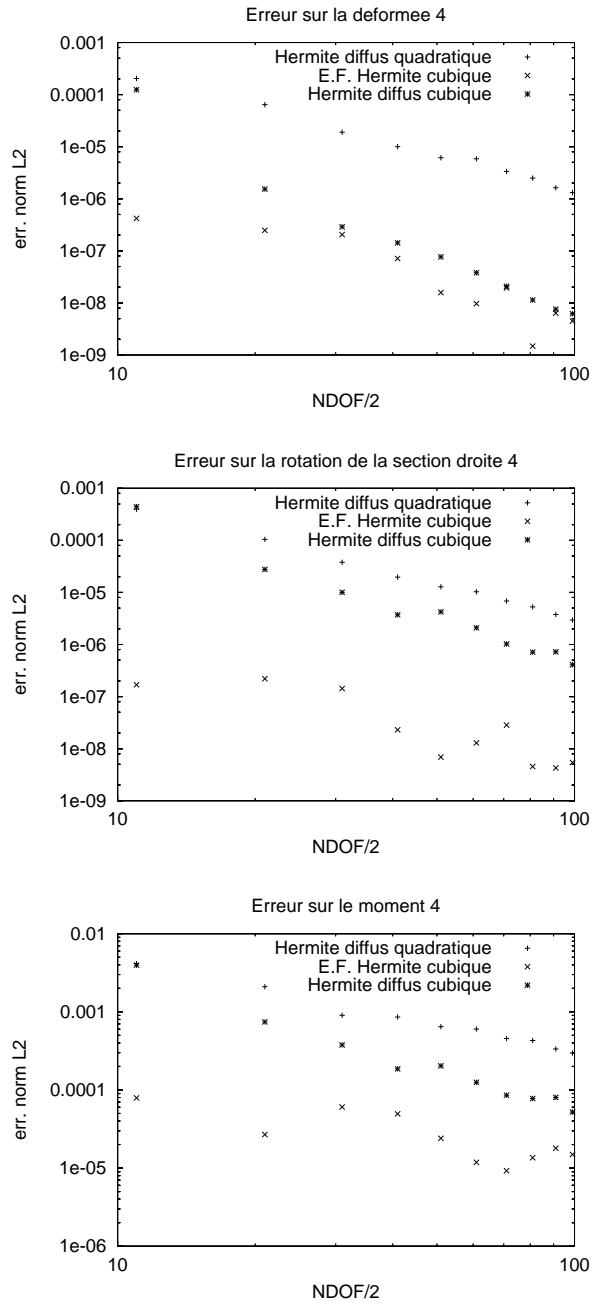


FIG. I.4: Convergence des éléments diffus Hermite pour le problème 4.





# Bibliographie

- [AC69] M. Albiges and A. Coin, *Résistance des matériaux appliquée (tome i)*, Eyrolles, 1969.
- [ACK99] S. N. Atluri, J. Y. Cho, and H.-G. Kim, *Analysis of thin beams using the meshless local petrov–galerkin method with generalized moving least squares interpolations*, *Comput. Mech.* **24** (1999), 334–347.
- [All93] J. Allison, *Multiquadratic radial basis functions for representing multidimensional high energy physics data*, *Computer Physics Communications* **77** (1993), 377–395.
- [Alu00] N. R. Aluru, *A point collocation method based on reproducing kernel approximation*, *International Journal for Numerical Methods in Engineering* **47** (2000), 1083–1121.
- [Aro50] N. Aronszajn, *Theory of reproducing kernels*, *Trans. Amer. Math. Soc.* **68** (1950), 337–404.
- [AZ00] S. N. Atluri and T. Zhu, *New concepts in meshless methods*, *International Journal for Numerical Methods in Engineering* **47** (2000), 537–556.
- [BB96] S. Beissel and T. Belytschko, *Nodal integration of the element–free galerkin method*, *Comput. Methods Appl. Mech. Engrg.* **139** (1996), 49–74.
- [BBAC99] H. Bung, M. Berthillier, S. Audic, and A. Combescure, *Traitement d’impact fluide–solide par la méthode particulière sph. application : ingestion d’oiseau par les turboréacteurs*, *Calcul des Structures*, vol. 1, 1999, Giens’99.
- [BJ85] P. J. Besl and R. C. Jain, *Three-dimensional object recognition*, *Computing Surveys* **17** (1985), no. 1, 75–145.
- [BK98] J. Bonet and S. Kulasegaram, *Correction and stabilization of smooth particule hydrodynamics methods with application in metal forming simulations*, Tech. report, Institute for Numerical Methods in Engineering, 1998.
- [BKO<sup>+</sup>96] T. Belytschko, Y. Krongauz, D. Organ, M. Fleming, and P. Krysl, *Meshless methods : An overview and recent developments*, *Comput. Methods Appl. Mech. Engrg.* (1996), 3–47.
- [BLG94] T. Belytschko, Y. Y. Lu, and L. Gu, *Element–free galerkin method*, *International Journal for Numerical Methods in Engineering* **37** (1994), 229–256.
- [Boi84] J. D. Boissonnat, *Geometric structures for 3d shape representation*, *ACM Transaction on Graphics* **3** (1984), no. 4, 266–286.

- [Bre83] H. Brezis, *Analyse fonctionnelle. théorie et applications*, Masson, 1983.
- [BTV99a] P. Breitkopf, G. Touzot, and P. Villon, *Explicit form and efficient computation of mls shape functions and their derivatives*, à paraître dans int. j. num. meth. engrg., 1999.
- [BTV99b] P. Breitkopf, G. Touzot, and P. Villon, *Méthodes alternatives aux éléments finis : collocation diffuse à double grille*, Actes du 4ième Colloque National en Calcul des Structures, 1999, pp. 549–554.
- [BTV99c] P. Breitkopf, G. Touzot, and P. Villon, *A robust integration strategy for meshless methods based on moving least squares approximation*, USNCCM'99, 1999.
- [BV99] P. Breitkopf and P. Villon, communication personnelle, 1999.
- [Car76] M.P Do Carmo, *Differential geometry of curves and surfaces*, Prentice Hall, 1976.
- [Cha94] P. Chauvet, *Aide mémoire de géostatistique linéaire*, École des Mines, Paris, 1994, Cahier de Géostatistique, Fascicule 2.
- [Cia98] P. G. Ciarlet, *Introduction à l'analyse numérique matricielle et à l'optimisation*, Dunod, 1998.
- [CMM99] M. K. Chati, S. Mukherjee, and Y. X. Mukerjee, *The boundary node method for three-dimensional linear elasticity*, International Journal for Numerical Methods in Engineering **46** (1999), 1163–1184.
- [CS97] L. H. T. Chang and H. B. Said, *A  $c^2$  triangular patch for the interpolation of scattered data*, Computer-Aided Design **29** (1997), no. 6, 407–412.
- [CS98] S. Couturier and H. Sadat, *A meshless method for the solution of incompressible flow equations*, Revue européenne des éléments finis **7** (1998), no. 7, 825–840.
- [Dau92] I. Daubechies, *Ten lectures on wavelets*, Soc. for Indust. and Appl. Math., 1992.
- [DB99] J. Dolbow and T. Belytschko, *Numerical integration of the galerkin weak form in meshfree methods*, Comput. Mech. **23** (1999), no. 3, 219–230.
- [DL98] B. Donning and W. K. Liu, *Meshless methods for shear-deformable beams and plates*, Comput. Methods Appl. Mech. Engrg. **152** (1998), 47–72.
- [DO96] C. A. Duarte and J. T. Oden, *H-p-clouds – an h-p meshless method*, Numer. Methods for Partial Differential Equations **16** (1996), 673–705.
- [DT84] G. Dhatt and G. Touzot, *Une présentation de la méthode des éléments finis*, Maloine S.A., 1984.
- [Dub81] O. Dubrule, *Krigeage et splines en cartographie automatique*, Ph.D. thesis, École des Mines, Paris, 1981.
- [Duc76] J. Duchon, *Interpolation des fonctions de deux variables suivant le principe de la flexion des plaques minces*, R.A.I.R.O Analyse Numérique **10** (1976), no. 12, 5–12.

- [EM94] H. Edelsbrunner and E. P. Mücke, *Three-dimensional*, ACM Transactions on Graphics **13** (1994), no. 1, 43–72.
- [Far90] G. Farin, *Surfaces over dirichlet tessellations*, Computer Aided Geometric Design (1990), 281–292.
- [FCMB97] M. Fleming, Y. A. Chu, B. Moran, and T. Belytschko, *Enriched element-free galerkin methods for crack tip fields*, International Journal for Numerical Methods in Engineering **40** (1997), 1483–1504.
- [FG96] J. Fan and I. Gijbels, *Local polynomial modeling and its applications*, Monographs on Statistics and Applied Propability, Chapman & Hall, 1996.
- [FHM95] C. A. Fellippa, B. Haugen, and C. Militello, *From the individual element test to finite element templates : Evolution of the patch test*, International Journal for Numerical Methods in Engineering **38** (1995), 199–229.
- [Gal15] B. G. Galerkin, *Series solution of some problems of elastic equilibrium of rods and plates*, Vest. Inzh. Tech. **19** (1915), 897–908, en russe, d’après Zienkiewicz/Taylor [ZT94].
- [GCK96] M. A. Golberg, C. S. Chen, and S. R. Karur, *Improved multiquadratic approximation for partial differential equations*, Engineering Analysis with Boundary Elements **18** (1996), 9–17.
- [GL89] G. H. Golub and C. F. Van Loan, *Matrix computations – second edition*, The Johns Hopkins University Press, 1989.
- [Gou97] F. Goulette, *Quelques outils de géométrie différentielle pour la construction automatique de modèles cao partir d’images télémétriques*, Ph.D. thesis, École des Mines, Paris, 1997.
- [Guy95a] F. Guyon, *Approximation diffuse*, Tech. report, Université de Technologie de Compiègne, 1995.
- [Guy95b] F. Guyon, *Interpolation diffuse. champs d’atténuation constant*, Tech. report, Université de Technologie de Compiègne, 1995.
- [HK97] G. Häusler and S. Karbacher, *Reconstruction of smoothed polyhedral surfaces from multiple range images*, 3D Image Analysis and Synthesis ’97, Infix Verlag, 1997, pp. 191–198.
- [HS92] R. M. Haralick and L. G. Shapiro, *Computer and robot vision*, vol. 1, Addison Wesley, 1992.
- [IL83] B. M. Irons and M. J. Loikkanen, *An engineers’ defence of the patch test*, International Journal for Numerical Methods in Engineering **19** (1983), 1391–1401.
- [IR72] B. M. Irons and A. Razzaque, *Experience with the patch test for convergence of finite elements*, Mathematical Foundations of the Finite Element Method (A. K. Aziz, ed.), Academic Press, 1972, pp. 557–587.
- [IW97] A. M. Mc Ivor and R. S. Walkenburg, *A comparison of local surface geometry estimation methods*, Tech. report, Industrial Research Limited, 1997, <http://www.irl.cri.nz/publications>.

- [IW98] A. M. Mc Ivor and P. T. Walkenburg, *Recognition of simple curved surface from 3d surface data*, ACCV'98, Hong Kong, vol. 1, 1998, pp. 434–441.
- [Jen72] P. S. Jensen, *Finite difference techniques for variable grids*, Comput. & Structures **2** (1972), 17–29.
- [JLB98] S. Jun, W. K. Liu, and T. Belytschko, *Explicit reproducing kernel particle methods for large scale deformation problems*, International Journal for Numerical Methods in Engineering **41** (1998), 137–166.
- [Kan90] E. J. Kansa, *Multiquadrics – a scattered data approximation scheme with applications to computational fluid–dynamics – i. surface approximation and partial derivative estimates*, Computers with mathematical applications **19** (1990), no. 8/9, 127–145.
- [Kan92] E. J. Kansa, *A strictly conservative spatial approximation scheme for the governing engineering and physics equations over irregular regions and inhomogeneously scattered nodes*, Computers Math. Appl. **24** (1992), no. 5/6, 169–190.
- [KB95] P. Krysl and T. Belytschko, *Analysis of thin plates by the element–free galerkin method*, Comput. Mech. **17** (1995), no. 1–2, 26–35.
- [KB96] P. Krysl and T. Belytschko, *Analysis of thin shells by the element–free galerkin method*, International journal of Solid and Structures **33** (1996), no. 20–22, 3057–3080.
- [KB98] Y. Krongauz and T. Belytschko, *Efg approximation with discontinuous derivatives*, International Journal for Numerical Methods in Engineering **41** (1998), 1215–1233.
- [Kro96] Y. Krongauz, *Applications of meshless methods to solid mechanics*, Ph.D. thesis, Northwestern University, 1996.
- [KS97] I. Kaljavic and S. Saigal, *An improved element free galerkin formulation*, International Journal for Numerical Methods in Engineering **40** (1997), 2953–2974.
- [Lao96] T. Laouar, *Contribution à l'étude de l'approximation diffuse: Auto-adaptativité en éléments diffus*, Ph.D. thesis, Université de Technologie de Compiègne, 1996.
- [LB00] W. K. Liu and P. Breitkopf, communication personnelle, 2000.
- [LBG94] Y. Y. Lu, T. Belytschko, and L. Gu, *A new implementation of the element free galerkin method*, Comput. Methods Appl. Mech. Engrg. **113** (1994), 397–414.
- [LC95] W. K. Liu and Y. Chen, *Wavelet and multiple scale reproducing kernel method*, International Journal for Numerical Methods in Fluids **21** (1995), 901–931.
- [LCUC96] W.K. Liu, Y. Chen, R. A. Uras, and C. T. Chang, *Generalized multiple scale reproducing kernel particle methods*, Comput. Methods Appl. Mech. Engrg. **139** (1996), 91–157.
- [LDT96] T. J. Liszka, C. A. M. Duarte, and W. W. Tworzydło, *hp-meshless cloud method*, Comput. Methods Appl. Mech. Engrg. **139** (1996), 263–288.

- [Lel90] A. Leluan, *Résistance des matériaux*, École Centrale Paris, 1990, support de cours.
- [Léo91] J. C. Léon, *Modélisation et construction de surfaces pour la CFAO*, Hermes, Paris, 1991.
- [LGB95] A. Leonardis, A. Gupta, and R. Bajcsy, *Segmentation of range images as the search for geometric parametric*, International Journal of Computer Vision **14** (1995), no. 3, 1995.
- [LHCG97] W. K. Liu, W. Hao, Y. Chen, and J. Gosz, *Multiresolution reproducing kernel particle methods*, Comput. Mech. **20** (1997), 295–309.
- [LHL00] S. Li, W. Hao, and W. K. Liu, *Numerical simulations of large deformation of thin shell structures using meshfree methods*, Computational Mechanics **25** (2000), 102–116.
- [Lis94] R. J. Lisle, *Detection of zones of abnormal strains in structures using gaussian curvature analysis*, AAPG Bulletin (1994), 1811–1819.
- [LJL<sup>+</sup>95] W. K. Liu, S. Jun, S. Li, J. Adee, and T. Belytschko, *Reproducing kernel particle methods for structural dynamics*, International Journal for Numerical Methods in Engineering **1995** (1995), 1655–1679.
- [LJZ95] W. K. Liu, S. Jun, and F. Zhang, *Reproducing kernel particle methods*, International Journal for Numerical Methods in Fluids **20** (1995), 1081–1106.
- [LL99a] S. Li and W. K. Liu, *Reproducing kernel hierarchical partition of unity, part i – formulation and theory*, International Journal for Numerical Methods in Engineering **45** (1999), 251–288.
- [LL99b] S. Li and W. K. Liu, *Reproducing kernel hierarchical partition of unity, part ii – applications*, International Journal for Numerical Methods in Engineering **45** (1999), 289–317.
- [LLB96] W. K. Liu, S. Li, and T. Belytschko, *Moving least square reproducing kernel method: Part 1: Methodology and convergence*, Comput. Methods Appl. Mech. Engrg. **143** (1996), 113–154.
- [LO80] T. Liszka and J. Orkisz, *The finite difference method at arbitrary irregular grids and its application in applied mechanics*, Comput. & Structures **11** (1980), 83–95.
- [LS80] P. Lancaster and K. Salkauskas, *Surfaces generated by moving least squares methods*, Math. Comp. **37** (1980), 141–158.
- [LS86] P. Lancaster and K. Salkauskas, *Curve and surface fitting; an introduction*, Academic Press, London, 1986.
- [LT90] P. Liang and J. S. Todhunter, *Representation and recognition of surface shapes in range images: A differential geometric approach*, Computer Vision, Graphics and Image Processing **52** (1990), 78–109.
- [Luc77] L. B. Lucy, *A numerical approach to the testing of the fission hypothesis*, Atron. Journ. **82** (1977), no. 12, 1013–1024.
- [Mal74] J.L. Mallet, *Présentation d'un ensemble de méthodes et techniques de la cartographie automatique numérique*, Fondation scientifique de la géologie et de ses applications, 1974.

- [Mao99] Z. Maouche, *Contribution à l'amélioration de la prédiction des systèmes rotary. couplage garniture-outil de forage*, Ph.D. thesis, Ecole des Mines de Paris, 1999.
- [Mat70] G. Matheron, *La théorie des variables régionalisées et ses applications*, École des Mines, Paris, 1970.
- [Mat73] G. Matheron, *The intrinsic random functions and their applications*, Adv. Appl. Probab. **5** (1973), 439–468.
- [Mat80] G. Matheron, *Splines et krigeage : leur équivalence formelle*, Tech. report, École des Mines, Paris, 1980.
- [Mel95] J. M. Melenk, *On generalized finite element methods*, Ph.D. thesis, University of Maryland, 1995.
- [Mey90] Y. Meyer, *Ondelettes et opérateurs i, ondelettes*, Hermann, Paris, 1990.
- [Mic86] C. A. Micchelli, *Interpolation of scattered data : Distance matrices and conditionally positive definite functions*, Constructive Approximation **2** (1986), 11–22.
- [MM97] Y. X. Mukherjee and S. Mukherjee, *The boundary node method for potential problems*, International Journal for Numerical Methods in Fluids **40** (1997), 797–815.
- [Mon92] J. J. Monaghan, *Smoothed particle hydrodynamics.*, Annu. Rev. Astron. Astrophys. **30** (1992), 543–574.
- [Mye92] D. E. Myers, *Kriging, cokriging, radial basis functions and the role of positive definiteness*, Computers Math. Applic. **24** (1992), no. 12, 139–148.
- [NTV92] B. Nayroles, G. Touzot, and P. Villon, *Generalizing the finite element method : Diffuse approximation and diffuse elements*, Comput. Mech. **10** (1992), 307–318.
- [OIZT96] E. Oñate, I. Idelsohn, O. C. Zienkiewicz, and R. L. Taylor, *A finite point method in computational mechanics. applications to convective transport and fluid flow*, International Journal for Numerical Methods in Engineering **39** (1996), 3839–3866.
- [Riv84] J. Rivoirard, *Le comportement des poids de krigeage*, Ph.D. thesis, École des Mines, Paris, 1984.
- [RSSV99] A. Rassineux, J. M. Savignat, O. Stab, and P. Villon, *Remaillage de surfaces par interpolation diffuse locale de type hermite*, Calcul des Structures, Giens'99, CSMA, 1999.
- [RSSV00] A. Rassineux, J-M. Savignat, O. Stab, and P. Villon, *Surface remeshing by local hermite diffuse interpolation*, International Journal for Numerical Methods in Engineering **49** (2000), 31–49.
- [RT98] P. A. Raviard and J. M. Thomas, *Introduction à l'analyse numérique des équations aux dérivées partielles*, Dunod, 1998.
- [Sal92] K. Salkauskas, *Moving least square interpolation with thin-plate splines and radial basis functions*, Computers Math. Applic. **24** (1992), no. 12, 177–185.

- [Sav98] J.M. Savignat, *Approximation sans maillage*, Tech. report, École des Mines, Paris, 1998.
- [Sav99] J-M. Savignat, *Construction géométrique de la trajectoire d'un puits de forage*, Tech. report, Ecole des Mines de Paris, 1999.
- [SB77] G. Sanders and P. Beckers, *The influence of the choice of connectors in the finite element method*, International Journal for Numerical Methods in Engineering **11** (1977), 1491–1505.
- [Sch99] R. Schaback, discussion informelle, 1999, Curves and Surface, Saint Malo '99.
- [SF73] G. Strang and G. J. Fix, *An analysis of the finite element method*, Prentice-Hall, inc., 1973.
- [Sib81] R. Sibson, *A brief description of natural neighbour interpolation*, Interpreting Multivariate Data (John Wiley & Sons, ed.), V. Barnett, 1981, pp. 21–36.
- [SM97] P. Samson and J. L. Mallet, *Curvature analysis of triangulated surfaces in structural geology*, Mathematical Geology **29** (1997), no. 3, 391–412.
- [SRD94] P. Shi, G. Robinson, and J. Duncan, *Myocardial motion and function assessment using 4d images*, IEEE conference on Visualization in Biomedical Computing, 1994.
- [SRSV99] J. M. Savignat, A. Rassineux, O. Stab, and P. Villon, *High resolution curvature analysis and its application to surface modeling*, to be published, 1999.
- [SSRV99] J. M. Savignat, O. Stab, A. Rassineux, and P. Villon, *Diffuse curvature computation for surface recognition*, Curve and Surface Design: Saint-Malo 1999 (Paul Sablonnière Pierre-Jean Laurent and Larry L. Schumaker, eds.), Vanderbilt University Press, Nashville, TN., 1999, pp. 363–370.
- [ST81] M. Syczewski and R. Tribillo, *Singularities of sets used in the mesh method*, Comput. & Structures **14** (1981), no. 5–6, 509–511.
- [Stu80] F. Stummel, *The limitations of the patch test*, International Journal for Numerical Methods in Engineering **15** (1980), 177–188.
- [Suk98] N. Sukumar, *The natural element method (nem) in solid mechanics*, <http://www.cqe.nwu.edu/~suku/nem/nem.html>, June 1998, Thèse, Northwestern University.
- [SVS99] J. M. Savignat, P. Villon, and O. Stab, *Construction of splines moving least square its properties and an application to differential equations*, to be published, 1999.
- [SVT<sup>+</sup>00] J-M. Savignat, P. Villon, M. Tijani, P. Breitkopf, and Z. Maouche, *éléments diffus hermite pour le calcul de poutre, application à un problème de forage*, IDMME'2000, 2000.
- [SVTB00] J-M. Savignat, P. Villon, M. Tijani, and P. Breitkopf, *Meshless methods for beams : Numerical integration and hermite diffuse elements*, ECCOMAS'2000, 2000.



- [TB95] E. Trucco and Fisher R. B., *Experiments in curvature-based segmentation of range data*, IEEE Transactions on Pattern Analysis and Machine Intelligence **17** (1995), no. 2, 177–182.
- [TBB94] M. Tabbara, T. Blacker, and T. Belytschko, *Finite element derivative recovery by moving least square interpolants*, Comput. Methods Appl. Mech. Engrg. **117** (1994), 211–223.
- [Tij78] M. Tijani, *Résolution numérique des problèmes d'elastoviscoplasticité*, Ph.D. thesis, Paris IV, P.M. Curie, 1978.
- [Tij99] M. Tijani, communication personnelle, 1999.
- [Ton69] P. Tong, *Exact solution of certain problems by finite-element method*, AIAA J. **7** (1969), no. 1, 178–180.
- [Tro93] F. Trochu, *A contouring program based on dual kriging interpolation*, Engineering with Computers **9** (1993), 160–177.
- [TSZC86] R. L. Taylor, J. C. Simo, O. C. Zienkiewicz, and A. C. H. Chan, *The patch test – a condition for assessing fem convergence*, International Journal for Numerical Methods in Engineering **22** (1986), 39–62.
- [Veu74] B. Fraeijs De Veubeke, *Variational principles and the patch test*, International Journal for Numerical Methods in Engineering **8** (1974), 783–801.
- [Vil91] P. Villon, *Contribution à l'optimisation*, Université de Technologie de Compiègne, 1991, Thèse d'état.
- [VMC97] T. Varady, R. R. Martin, and J. Cox, *Reverse engineering of geometric models*, Computer-Aided Design **29** (1997), no. 4, 255–268.
- [WA94] J. R. Williams and K. Amaratunga, *Introduction to wavelets in engineering*, International Journal for Numerical Methods in Engineering **37** (1994), 2365–2388.
- [Wah90] G. Wahba, *Spline model for observational data*, Soc. for Indust. and Appl. Math., 1990.
- [Wen98] H. Wendland, *Error estimates for interpolation by compactly supported radial basis functions of minimal degree*, Journal of approximation theory **93** (1998), no. 2, 258–272.
- [Wen99] H. Wendland, *Meshless galerkin methods using radial basis functions*, Math. Comp. **68** (1999), no. 228, 1521–1531.
- [YT94] T. Yoshimi and F. Tomita, *Robust curvature vectors calculation from range data using isl method*, MVA'94: IAPR Workshop on machine Vision Application, 1994, Kawasaki, Japan, pp. 506–509.
- [Zie00] O. C. Zienkiewicz, *Achievements and some unsolved problems of the finite element method*, International Journal for Numerical Methods in Engineering **47** (2000), 9–28.
- [ZPC98] M. Zerroukat, H. Power, and C. S. Chen, *A numerical method for heat transfer problems using collocation and radial basis functions*, International Journal for Numerical Methods in Engineering **42** (1998), 1263–1278.
- [ZT94] O. C. Zienkiewicz and R. L. Taylor, *The finite element method*, vol. 1 Basic Formulation and Linear Problems, Mc Graw-Hill, 1994.

- [ZT97] O. C. Zienkiewicz and R. L. Taylor, *The finite element patch test revisited: A computer test for convergence, validation and error estimates*, Comput. Methods Appl. Mech. Engrg. **149** (1997), no. 1–4, 223–254.