

Une approche tensorielle pour la discrétisation EF d'EDP

Implantation dans un environnement à objets

R. Saad¹, D. Eyheramendy²

¹ LMA, CNRS UPR 7051, Marseille, France, saad@lma.cnrs-mrs.fr

² LMA, CNRS UPR 7051, Ecole Centrale Marseille, France, dominique.eyheramendy@centrale-marseille.fr

Résumé — Les problèmes de mécanique deviennent aujourd'hui de plus en plus complexes. Ils intègrent par exemples des phénomènes physiques annexes ayant des conséquences mécaniques non négligeables. Cela justifie depuis quelques années, l'utilisation de logiciels de manipulations algébriques appliqués à la dérivation systématique de modèles de discrétisation, de type élément finis par exemple. Dans ce papier nous présentons un environnement générique basé sur un concept objet pour automatiser la méthode des éléments finis et basé sur l'analyse tensorielle appliquée à la discrétisation par éléments finis.

Mots clefs — Eléments finis, orienté objet, analyse tensorielle, discrétisation automatique d' EDP.

Les programmes d'élément finis se composent de plusieurs centaines de milliers de lignes de code, exclusivement écrit en FORTRAN avant les années 90. Aujourd'hui la résolution de problèmes de mécanique devient de plus en plus complexe. Ces problèmes impliquent de plus en plus souvent différentes physiques à plusieurs échelles de temps et d'espace... Dans les années 90, Fenves [1] a été l'un des premiers à décrire les avantages de la programmation orientée-objet pour le développement du génie logiciels afin d'appréhender la complexité des codes de calculs. Parmi eux, on peut citer Miller [2] et [Zimmermann 3] pour les pionniers, et sur les approches avancées Mackerle [4] et Eyheramendy [5, 9, 10], Nikishkov [6], Makie [7] et Vanderheyden [8]...

L'utilisation de logiciels de manipulation algébrique a toujours été un point d'intérêt pour le développement des éléments finis Luft [11], Noor [12], Yagawa [13] et Eyheramendy [14].

Récemment, Korelc [15] a présenté une approche Mathematica qui contient le calcul symbolique, la génération automatique du code et l'interfaçage du code élément. De même Logg [16] a présenté le projet FENICS dont le but explicite est de développer des logiciels pour l'automatisation de la modélisation mathématique de calcul, inclus une automatisation de la méthode des éléments finis.

Dans notre approche Saad [17,18], nous visons à développer un environnement générique basé sur un concept objet pour automatiser la méthode des éléments finis. Pour atteindre cet objectif, nous fondons notre approche sur l'analyse tensorielle appliquée à la discrétisation par éléments finis. La représentation indicielle des contributions élémentaires est obtenue à partir des données des opérateurs différentiels et des opérations tensorielles. Ainsi la discrétisation de la forme variationnelle est générée automatiquement sous forme symbolique dans un formalisme de haut niveau d'abstraction. L'intérêt de cette approche est la généralité de la description qui peut être étendue naturellement à tout modèle de discrétisation (spatiale ou temporelle). Ce caractère générique est conservé grâce à l'approche logicielle choisie qui orientée objet en Java. Nous proposons donc un cadre orienté objet basé sur des concepts symboliques capables de gérer des codes traditionnels d'éléments finis. Dans cette présentation, nous décrivons la base du cadre orientée décrivant un schéma de discrétisation par éléments finis.

L'approche comporte deux étapes, la première consiste à réduire une formulation variationnelle quelconque à une forme bilinéaire. Cette forme peut s'exprimer sous la forme d'une intégrale et sous la forme de produit de tenseurs. On obtient ainsi après contraction de certains indices des tenseurs, les contributions élémentaires du système. La deuxième étape est la pré-compilation automatique de ces formes élémentaires afin de les intégrer dans un code de calcul classique.

Références

- [1] G.L. Fenves, *Object-Oriented programming for engineering software development*, Engineering with Computer, vol. 6, page1-page15, 1990.
- [2] G.R Miller, *A LISP-Based Object-Oriented approach to structural analysis*, Engineering with Computer, vol. 4, page197-page203, 1988.
- [3] Y. Dubois-Pèlerin and Th. Zimmermann, *Object-oriented finite element programming: II. A prototype in smalltalk*, Computer Methods in Applied Mechanics and Engineering; Vol. 108, page165-page183, 1993.
- [4] J. Mackerle, *Object-oriented programming in FEM and BEM: a bibliography (1990-2003)*, Advances in Engineering Software, vol. 35, page325-page336, 2004.
- [5] D. Eyheramendy, Th. Zimmermann, *Object-oriented finite elements: III. Theory and application of automatic programming*, Computer Methods in Applied Mechanics and Engineering, vol 154, page41-page68, 1998.
- [6] G.P. Nikishkov, *Object-oriented design of a finite element code in Java*, Computer Modeling in Engineering and Sciences, vol. 11, page81-page90, 2006.
- [7] R.I. Mackie, *Object oriented implementation of distributed finite element analysis in .NET*, Advances in Engineering Software, vol. 38, page726-page737, 2007.
- [8] W.B. VanderHeyden, E.D. Dendy, N.T. Padial-Collins, *CartaBlanca– A Pure-Java, Component-based Systems Simulation Tool for Coupled Non-linear Physics on Unstructured Grids*, Joint ACM Java Grande - ISCOPE 2001 Conference Stanford, California June 2-4, 2001.
- [9] D. Eyheramendy and F. Oudin, *Advanced object-oriented techniques for coupled multiphysics*, In Civil engineering computations: Tools and Techniques Ed. B.H.V. Topping, ©Saxe-Cobourg Publications, Chap. 3, page37-page60, 2007.
- [10] D. Eyheramendy, *Advanced object models for mathematical consistency enforcement in scientific computing*, WSEAS Transactions on Mathematics, vol. 4, N° 4, page457-page463, 2005.
- [11] R.W. Luft, J.M. Roesset and J.J. Connor, *Automatic generation of finite element matrices*, Struct. Div., Proceedings of ASCE, page349-page361, 1971.
- [12] A.K. Noor, C.M. Andersen, *Computerized symbolic manipulation in nonlinear finite element analysis*, Computers & Structures vol. 13, page379-page403, 1981.
- [13] G. Yagawa, G.-W. Ye and S. Yoshimura, *A numerical integration scheme for finite element method based on symbolic manipulation*, Internat. J. Numer. Methods Engrg., vol. 29, page1539-page1549, 1990.
- [14] D. Eyheramendy, Th. Zimmermann, *The Object-oriented finite elements: II. A symbolic environment for automatic programming*, Computer Methods in Applied Mechanics and Engineering, vol. 32, page259-page276, 1996.
- [15] J. Korelc, *Multi-language and Multi-environment Generation of Nonlinear Finite Element Codes*, Engineering with Computers, vol. 18, page312-page327, 2002.
- [16] A. Logg, *Automating the Finite Element Method*, Arch Comput Methods Eng., vol. 14, page93-page138, 2007.
- [17] R. Saad, *An Object-Oriented Framework for Automated Computer Aided Finite Element Derivation*, Proceedings of the Seventh International Conference on Engineering Computational Technology, Civil-Comp Press, Valencia, Spain, 2010.
- [18] R. Saad, *Generalized finite element discretization: An object-oriented analysis*, IV European Conference on Computational Mechanics, Paris, France, 2010.