

Proposition de projet LIP6

Algorithmes symbolique-numériques pour la résolution de systèmes polynomiaux

Christophe Denis Jean-Charles Faugère Stef Graillat
Fabienne Jézéquel Fabien Rico Philippe Trébuchet

1^{er} mars 2007

Ce document présente une proposition de projet LIP6 sur le thème de l'algorithmique *symbolique-numérique*. Le contexte du projet est présenté en section 1. La description du travail est présentée en section 2. La section 3 décrit succinctement les résultats attendus et les domaines d'application de ce travail. Enfin, la section 4 est consacrée au budget prévisionnel du projet.

1 Contexte du projet

Les deux équipes (PEQUAN et SPIRAL) du département Calcul Scientifique du LIP6 sont impliquées dans une action commune où interviennent des algorithmes symboliques et numériques. Le but de cette action est de développer des méthodes de calcul combinant les deux types d'algorithmes. Les algorithmes symboliques fournissent une représentation exacte du résultat mais peuvent être lents. Les algorithmes numériques (exécutés en précision finie) sont, quant à eux, plus rapides en général mais fournissent des résultats approchés. Profiter des performances des unités d'arithmétique flottante est un des enjeux actuels du calcul symbolique, motivé par l'augmentation de la complexité des problèmes algébriques traités. La principale difficulté consiste à déterminer quelle partie d'un algorithme symbolique peut être effectuée en précision finie sans que les erreurs d'arrondi n'entachent la qualité du résultat.

La recherche dans le domaine des algorithmes symbolique-numériques est en pleine expansion ; on peut citer par exemple la conférence SNC'07 (International Workshop Symbolic-Numeric Computation) qui aura lieu en juillet au Canada ou bien le numéro spécial de la revue *Mathematics in Computer Science* intitulé « Symbolic and Numeric Computation ».

2 Description du travail

Le but du projet est de proposer de nouveaux algorithmes symbolique-numériques pour la résolution de systèmes polynomiaux. À l'heure actuelle, la technique utilisée consiste à

transformer le problème de la recherche des racines d'un système polynomial en un problème de calcul de valeurs propres d'une matrice. Cette transformation se fait de manière symbolique. Pour le calcul des valeurs propres, deux méthodes ont été proposées. La première [4, 5] consiste à calculer le polynôme caractéristique de la matrice (en symbolique) puis à calculer les racines de ce polynôme en arithmétique d'intervalles. Il s'agit donc bien de calcul symbolique suivi de calcul numérique validé : le résultat est certifié. La seconde approche [7] consiste à transformer la matrice (dont les coefficients sont rationnels) en une matrice à coefficients flottants. Il est alors clair que l'on fait déjà une approximation. Les valeurs propres de cette matrice sont alors calculées numériquement par des algorithmes classiques comme ceux inclus dans la bibliothèque LAPACK (ici décomposition de Schur). Le problème principal est que les erreurs d'arrondi dues au calcul peuvent dégrader la qualité du résultat voire même renvoyer un résultat n'ayant aucun chiffre en commun avec le résultat exact. Pour pouvoir juger de la qualité numérique du résultat, on utilise des estimateurs du conditionnement (inclus dans LAPACK) qui permettent d'obtenir une approximation du nombre de chiffres significatifs exacts. Malheureusement, les estimateurs de conditionnement peuvent eux aussi donner des résultats incorrects, ce qui peut faire croire que le résultat est très précis alors que ce n'est pas le cas.

Dans ce projet, nous allons nous consacrer à la seconde approche, le but étant d'obtenir une certification de la précision du résultat tout en calculant avec une arithmétique flottante [3, 6]. Le projet a pour but de réaliser les objectifs suivants :

1. la première étape consiste à contrôler la validité numérique du code à l'aide de la bibliothèque CADNA [1] afin de détecter de possibles instabilités numériques.
2. la deuxième étape consiste à utiliser la bibliothèque MPFR [2] pour pouvoir utiliser le code avec une précision arbitraire fixée ; la stratégie optimale étant de doubler à chaque étape la précision de calcul jusqu'à obtenir la le résultat avec la précision désirée.
3. la troisième étape consiste à valider la précision du résultat tout en travaillant en double précision IEEE. En effet, l'utilisation de la double précision permet de maximiser les performances des calculs. Néanmoins, on n'a alors aucune information sur la précision du résultat. Notre but est d'utiliser les méthodes « auto-validantes » (self-validating methods [8]) pour garantir en précision finie la qualité numérique du résultat. Si le résultat n'est pas assez précis, nous exécuterons alors une ou plusieurs étapes de raffinement itératif [9] pour atteindre la précision désirée (qui sera ensuite validée par les méthodes « auto-validantes »).

3 Résultats attendus et domaines d'application

Nous espérons obtenir un algorithme de calcul numérique-symbolique pour le calcul de valeurs propres plus rapide que celui de la première approche décrite précédemment. L'idée est de tirer pleinement parti des performances des unités flottantes actuelles.

Le principal domaine d'application de ce travail est la *robotique*. En effet, la conception et l'étude des robots parallèles nécessitent la résolution de modèles géométriques (notamment le calcul des coordonnées des bras du robot et de la plateforme) qui eux-mêmes nécessitent la résolution de systèmes polynomiaux de tailles conséquentes. C'est donc un enjeu majeur que de pouvoir résoudre efficacement de tels systèmes.

4 Budget prévisionnel

Le budget prévisionnel s'élève à 6000 € répartis comme suit.

rémunération d'un stagiaire	2100 € (6×350 €)
2 conférences internationales	3000 €
divers (matériel informatique)	900 €

Références

- [1] *The CADNA library*. Disponible à l'URL = <http://www.lip6.fr/cadna/>.
- [2] *The MPFR library*. Disponible à l'URL = <http://www.mpfr.org>.
- [3] Françoise Chaitin-Chatelin and Valérie Frayssé. *Lectures on finite precision computations*. Software, Environments, and Tools. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1996.
- [4] J. C. Faugère, P. Gianni, D. Lazard, and T. Mora. Efficient computation of zero-dimensional Gröbner bases by change of ordering. *J. Symbolic Comput.*, 16(4) :329–344, 1993.
- [5] Jean-Charles Faugère. A new efficient algorithm for computing Gröbner bases without reduction to zero (F5). In *Proceedings of the International Symposium on Symbolic and Algebraic Computation*, pages 75–83. ACM Press, 2002.
- [6] Nicholas J. Higham. *Accuracy and stability of numerical algorithms*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second edition, 2002.
- [7] Bernard Mourrain and Philippe Trébuchet. Generalized normal forms and polynomial system solving. In M. Krauers, editor, *International Symposium on Symbolic and Algebraic Computation*, pages 253–260. ACM Press, 2005.
- [8] Shin'ichi Oishi. Fast enclosure of matrix eigenvalues and singular values via rounding mode controlled computation. *Linear Algebra Appl.*, 324(1-3) :133–146, 2001. Special issue on linear algebra in self-validating methods.
- [9] Françoise Tisseur. Newton's method in floating point arithmetic and iterative refinement of generalized eigenvalue problems. *SIAM J. Matrix Anal. Appl.*, 22(4) :1038–1057 (electronic), 2001.