(info confidentielle) Arthur

MPI

Procédure de primitivation de fonctions rationnelles et logarithmiques

Les calculatrices modernes sont des outils utilisés pour contourner les difficultés calculatoires. Un exemple typique est la détermination de primitives explicites de fonctions, dont le calcul manuel est souvent pénible. Je me suis donc intéressé à la manière dont ces machines réalisent leurs opérations, en particulier leur procédure de primitivation.

La recherche d'une primitive explicite est normalement une procédure réalisée manuellement et dont le résultat est issu de choix souvent arbitraires. Dans le but d'éliminer des incertitudes, on cherche par le biais d'un algorithme qui réalisera diverses transformations à obtenir une primitive à partir d'une fonction qui sera convertie informatiquement.

Positionnement thématique (ÉTAPE 1) :

- *INFORMATIQUE (Informatique Théorique)*
- *MATHEMATIQUES (Algèbre)*
- *INFORMATIQUE (Informatique pratique)*

Mots-clés (ÉTAPE 1) :

Mots-clés (en français)	**Mots-clés (en anglais)**
Primitive	*Antiderivative*
Fraction rationnelle	*Rational function*
Décomposition de fraction	*Fraction decomposition*
Factorisation en polynôme sans carré	*Square-free factorisation*
Algèbre différentielle	*Differential algebra*

Bibliographie commentée

La dérivation et l'intégration sont des opérations qui apparaissent dans une multitude d' équations traduisant des phénomènes physiques et mathématiques. La faculté de primitiver une fonction permet parfois ainsi d'apporter une solution simple à un problème.

L'étude de la dérivation par Leibniz et Newton au XVII siècle a mené à la recherche de fonctions dont la dérivée est une autre fonction particulière, de tels fonctions sont appelées des primitives. Les travaux sur les corps différentiels réalisés par Liouville au XIX siècle permet d'exprimer une condition nécessaire à l'obtention d'une primitive explicite d'une fonction [1]. Une méthode générale reposant sur la manipulation des expressions, et la résolution d'équations différentielles permet d'obtenir une primitive s'il en existe une ou dans le cas échéant de fournir une preuve qu'il n'existe pas de primitives élémentaires à telle fonction.

Ainsi lors de l'avènement des ordinateurs, lorsque les fonctions mathématiques ont été implémentées, s'est posé le problème de la primitivation par une procédure qui est assurée d'aboutir, à l'inverse des méthodes mathématiques utilisées pour chercher des primitives manuellement [2]. Des algorithmes comme ceux de Hermite [4], et de Trager, Rioboo, Lazard permettent de primitiver des fonctions rationnelles sans avoir besoin d'en déterminer les pôles explicitement de manière exacte ce qui est généralement impossible de faire avec un ordinateur.

En 1968 Robert Risch propose un algorithme récursif, permettant d'intégrer des fonctions comportant une fraction rationnelle, des logarithmes, des radicaux, et des exponentielles. En 1970, il fournit une preuve contredisant la conjecture d'Hardy, qui prévoit qu'il ne puisse y avoir de procédure d'intégration [5]. Cependant l'algorithme de Risch n'est pas encore fini en tant que tel, certaines extensions des corps de fonctions différentielles ne peuvent donc pas être intégrées par les premières et dernières versions de l'algorithmes [2]. Cependant des améliorations sont encore réalisées sur les cas que sait déjà traiter l'algorithme comme le cas des fonctions aux extensions exponentielles [6], perfectionné par Manuel Bronstein.

Problématique retenue

Dans quelle mesure la mise en place d'un algorithme de recherche de primitive automatique est-il pertinent ?

Objectifs du TIPE du candidat

On se propose d'implémenter une partie de l'algorithme de Risch prenant en charge l'ensemble des fractions rationnelles étendues par des extensions logarithmiques. Cette implémentation personnelle sera comparée sur un ensemble de fonctions variées à d'autres systèmes de primitivation. Une partie de cette implémentation comprendra une interface permettant de convertir les fonctions vers la représentation symbolique utilisée par l'algorithme avant de la reconvertir en fonction connue.

Références bibliographiques (ÉTAPE 1)

[1] K. O. GEDDES, S. R. CZAPOR, G. LABAHN : Algorithms for Computer Algebra *: Springer, 1992, ISBN : 9780792392590*

[2] MANUEL BRONSTEIN : Symbolic Integration I, transcendental functions *: Springer, 1997, ISBN : 9783662033869*

[3] COLORADO STATE UNIVERSITY : Risch's algorithm for integration *: https://www.math.colostate.edu/~hulpke/lectures/m676ca/maplerisch.pdf*

[4] SAM BLAKE, WOLFRAM DEMONSTRATIONS PROJECT : Integration using Hermite Reduction *: https://demonstrations.wolfram.com/IntegrationUsingHermiteReduction/*

[5] MANUEL BRONSTEIN INRIA SOPHIA ANTIPOLIS : SYMBOLIC INTEGRATION TUTORIAL *: https://www-sop.inria.fr/cafe/Manuel.Bronstein/publications/issac98.pdf*

[6] MANUEL BRONSTEIN : The transcendental Risch differential equation *: Journal of Symbolic Computation 9 (1990), 49-60*

DOT

[1] *: Mars 2024 : Acquisition de ressources en ligne notamment les références [1] et [2] pour comprendre le fonctionnement de l'algorithme de Risch. Abandon de la compréhension du seul code trouvé de l'algorithme implémenté dans sympy.*

[2] *: Mai à Octobre 2024 : Définition et implémentation des types ainsi que les opérations de bases qui seront nécessaires.*

[3] *: Novembre 2024 : Résolution de nombreux bugs et passage des nombres complexes aux nombres rationnels suite à des problèmes d'approximations sur les floats.*

[4] *: Janvier 2025 : Première intégration rationnelle réussite suite à l'implémentation des algorithmes nécessaires. Cependant certaines fonctions provoques encore des erreurs.*

[5] *: Février à Mars : Nouvelle affichage plus concis pour vérifier les résultats, résolution de nombreux problèmes liés à l'intégration rationnelle. Première intégration logarithmique réussite avec cette fois aussi quelques problèmes.*

[6] *: Début Mai 2025 : Résolution des problèmes précédents liés à une mauvaise compréhension des explications donnés par [1]. Recherche automatique des extensions nécessaires et implémentation de la fonction principal.*

[7] *: Fin Mai 2025 : Implémentation d'une interface utilisateur rudimentaire qui ne sera pas plus développée suite à des difficultés d'utilisation des modules Ocaml. Tests sur de multiples fonctions à l'aide du produit fini.*