Polynômes : du discret (codes correcteurs et cryptographie basée sur les codes) et du continu (autour des trajectoires optimales)

Olivier Ruatta

Habilitation à dirigé les recherches de l'Université de Limoges

Polynômes : du discret (codes correcteurs et cryptographie basée sur les codes) et du continu (autour des trajectoires optimales)

Olivier Ruatta

XLIM UMR 7252 Université de Limoges - CNRS
Soutenue le 29 juin 2022

Jury :

Philippe Gaborit, Professeur à l'Université de Limoges (examinateur)
François Jouve, Professeur à l'Université Paris Cité (rapporteur)
Bernard Mourrain, Directeur de recherche à l'INRIA (examinateur)
Jean-Pierre Tillich, Directeur de recherche à l'INRIA (rapporteur)
Félix Ulmer, Professeur à l'Université de Rennes 1 (rapoorteur)
Jean-Claude Yakoubsohn, Professeur émérite à l'Université de Toulouse III (Président)
Jacques-Arthur Weil, Professeur à l'Université de Limoges (examinateur)

Chapitre 1
Parcours et introduction

1.1. Parcours

Dans ce chapitre, je décris mon parcours scientifique et professionnel afin d'expliquer ce qui me conduit à rédiger une habilitation à diriger les recherches. Je commence d'abord par mes activités de recherche et je continuerai par celles d'enseignement bien que ces activités aient été souvent étroitement liées.

J'ai effectué ma thèse à l'INRIA au titre de l'Université d'Aix-Marseille II sous la direction de Bernard Mourrain. Ma thèse, soutenue en septembre 2001, s'intitule « Dualité algébrique, structures et applications » [44] et est constituée des travaux sur les applications des représentations des algèbres de dimension 0 et résidus algébriques de celles-ci avec des applications à l'interpolation algébrique multivariée et le calcul de racines. Les formules théoriques obtenues avaient pour objectif de concevoir des algorithmes pour calculer des objets permettant de résoudre des systèmes algébriques ou caractérisant des ensembles algébriques. Les formules d'interpolation et l'application à la méthode de Weierstrass pour le calcul simultané des racines d'une variété de dimension 1 et les méthodes d'homotopie qui en découlent constituent les contributions qui me semblent les plus importantes dans ces travaux.

Je suis alors parti un an en Italie en post-doctorat à l'université de Pise, d'abord financé par le réseau européen (European training network) Real Algebraic and Analytic Geometry, puis par un programme de l'université de Pise sur l'algèbre commutative effective. Alors que jusque là je me consacrais pour l'essentiel à la géométrie complexe, j'ai appris beaucoup de choses sur la géométrie réelle.

Je suis rentré en France sur un support d'ATER à l'université de Nice-Sophia Antipolis en continuant à travailler avec le groupe GALAAD à l'INRIA.

Enfin, avant mon recrutement comme maître de conférences à l'université de Limoges, j'ai été post-doctorant pour le projet européen AIM@SHAPE à l'INRIA Sophia-Antipolis. Durant ces années post-doctorales, j'ai continué à travailler sur les sous-résultants et la méthode de Weierstrass multivariés avec Agnes Szanto pour finir des années plus tard avec un article à l'historique de publication rocambolesque [46].

J'ai commencé à Limoges le 1er février 2005 dans l'équipe de Calcul Formel et j'y ai réalisé la plupart de ma carrière jusqu'à ce que je rejoigne l'équipe CRYPTIS récemment.

J'ai commencé sur un poste pour lequel j'enseignais pour le département Technologie de l'Information et de la Communication (PHP, des bases de données, de la gestion de projets, de la programmation javascript, de l'algorithmique pour des formations professionnalisantes) et pour le département de Mathématiques pour lequel j'ai dispensé essentiellement des cours de DEA, de Master pour le Master CRYPTIS puis ACSYON en parallèle.

Dès le début, le profil recherche du poste sur lequel j'ai été recruté explicitait la volonté d'interaction entre « calcul formel » et « codes et cryptographie » et plus généralement avec le reste du laboratoire, en effet c'était la création du laboratoire XLIM qui rassemblait des laboratoires de mathématiques, d'informatique, d'électronique et d'optique. C'était le début d'une très longue collaboration avec Philippe Gaborit et Thierry Berger sur les codes correcteurs et les les algorithmes de décodage puis les applications à la cryptographie. J'ai toujours continué à avoir des activités « en caractéristique zéro » en parallèle. Suivant des travaux commencés durant mes années de post-doctorat, j'ai co-encadré avec Moulay Barkatou et Bernard Mourrain la thèse de Daouda Niang Diatta [19] sur le calcul de topologie de variétés algébriques réelles de petites dimensions (courbes et surfaces). Daouda Niang Diatta a soutenu sa thèse en 2009. Je commençais alors une recherche en relation entre mes travaux sur les courbes paramétrées et l'optimisation de formes avec des application à la physique et j'ai mené avec Olivier Prot un projet « Carnot » montrant la faisabilité de l'approche que je proposais. C'est le début de mes travaux sur les contours libres pour l'optimisation de formes.

Ensuite j'ai commencé à travailler avec Paola Boito, alors post-doctorante à Toulouse, sur des algorithmes symboliques-numériques et plus spécialement sur le calcul de PGCD approchés. C'est un domaine où la structure des anneaux polynomiaux quotients est très liée aux structures des matrices qui interviennent et nous avons proposé un algorithme basé sur les matrices de multiplication pour calculer des PGCD approchés [8].

Pendant ce temps, avec Philippe Gaborit, nous commencions à travailler sur les codes en métrique rang. Initialement, nous cherchions un algorithme de décodage en liste pour les codes de Gabidulin après avoir travaillé sur le décodage en métrique de Hamming dans [23] et [25]. Ce fut notre porte d'entrée sur une thématique féconde sur laquelle nous avons beaucoup travaillé depuis avec des étudiants et de nombreux collègues. Ce qui m'intéressait particulièrement était le rôle des polynômes linéarisés vis-à-vis de ces codes. Ce point de vue allait se révéler productif car même sur des sujets pour lesquels ils ont disparu, ils ont été à l'origine de beaucoup d'idées nouvelles notamment sur les codes en métrique rang et leur décodage. Deux thèses ont commencée presque en parallèle sur cette thématique : la thèse de Gaëtan Murat, très orienté sur les polynômes de Öre, et la thèse de Julien Schrek, encadrée par Philippe Gaborit, beaucoup plus orientée sur le décodage et les applications des codes en métrique rang à la cryptographie. Durant la thèse de Gaëtan Murat, nous définissons les codes « Low Rank Parity Check » - LRPC - et proposons un premier algorithme de décodage basé sur une formulation à base de polynômes linéarisés. Gaëtan Murat a soutenu sa thèse en 2014 [40].

En parallèle, je travaillais dans l'axe transverse CAO du laboratoire pour apporter une compétence mathématique aux travaux qui regroupaient des acteurs de tous les axes d'XLIM avec toujours l'optimisation de formes en ligne de mire, ce qui m'a conduit à exposer un principe de déformation de contours libres pour l'optimisation de formes. Cette idée a déjà émergé plusieurs fois, mais sans donner de développements systématiques. Nous continuions à travailler sur le décodage des codes en métrique rang et les applications à la cryptographie et plus particulièrement sur les attaques sur les systèmes basés sur les codes en métrique rang avec la thèse de Adrien Hauteville [29] qu'il a soutenue en 2017 en co-encadrement avec Philippe Gaborit, Jean-Pierre Tillich et moi-même.

Sous l'impulsion de Philippe Gaborit, nous avons participé à des soumissions pour l'appel à propositions du NIST sur les systèmes cryptographiques « post-quantiques » sur lequel nous avons pu travailler grâce à l'obtention du financement du projet ANR CBCRYPT. Nous avons notamment amélioré des systèmes cryptographiques et les études des attaques de ces systèmes.

En co-direction avec Paul Armand et Stéphane Bila, j'ai commencé à travailler avec Pierre Bonnélie pour formaliser l'approche « contours libres », m'intéresser à la partie algorithmique et à des applications, avec en ligne de mire l'optimisation de filtres électromagnétiques [9]. En étudiant la partie algorithmique et après les travaux réalisés en collaboration avec Pauline Merveilleux et Ouiddad Labbani-Igbida [31] dans le cadre de la thèse de Pauline Merveilleux sur la segmentation d'images en temps réel pour la robotique [38], nous avons pu montrer, avec Pierre Bonnélie, Fabien Caubet et Loïc Bourdin, que l'approche proposée permettait une alternative intéressante aux méthodes de gradients topologiques ou de lignes de niveau pour l'optimisation de formes qui ne sont pas connexes [10]. Pierre Bonnélie a soutenu sa thèse en 2017. Parallèlement, avec Stéphane Bila, j'ai co-encadré Satafa Sanogo en post-doctorat sur l'optimisation de composants électromagnétiques.

Suite à mes échanges réguliers avec le Vietnam et à l'encadrement de nombreux mémoires de M2 pour le Master ACSYON, avec Jacques-Arthur Weil, j'ai commencé à co-encadrer Hoang Van Duc grâce à un financement du programme 911 (qui consistait à faire soutenir des thèses à des enseignants non docteurs des universités vietnamiennes). Hoang Van Duc, comme Pierre Bonnélie, a réalisé son mémoire de M2 ACSYON sous ma direction en 2015. Les bourses 911 étaient des financements sur 4 ans. Hoang Van Duc a soutenu sa thèse en 2020 [30]. En plus de continuer sur la thématique de l'optimisation de formes, nous avons introduit une nouvelle thématique qui m'a initialement été inspirée par les méthodes d'homotopie : les trajectoires optimales. L'intérêt pour les trajectoires optimales était d'approcher rapidement de chemins qui permettent d'améliorer la complexité des méthodes d'homotopie, dites aussi de prédiction-correction : c'est l'object de l'approximation de géodésiques pour la métrique du conditionnement. Nous nous sommes alors rendu compte que cette approche par « trajectoires optimales » permettait d'avoir une approche géométrique intéressante pour les systèmes différentiels et le contrôle, surtout si on ne s'intéresse

pas à des solutions exactes et qu'on souhaite avoir une bonne approximation des trajectoires. Les travaux ne sont pas complètement aboutis dans la thèse et j'ai commencé à travailler sur une nouvelle famille de codes correcteurs, ce qui a retardé ce sujet. J'évoquerai les perspectives de ces sujets à la fin des chapitres correspondants. Enfin, avec Stéphane Bila, Christophe Durousseau et Cyrille Menudier, nous avons continué à travailler sur l'optimisation de formes pour la conception de composants dans le cadre d'une action avec le CNES et j'ai participé à l'encadrement d'Ali Dia qui devrait soutenir cette année une thèse de physique sur l'utilisation des courbes de Bézier pour l'optimisation de dispositifs électromagnétiques. Cette thèse a donné lieu à la réalisation de prototypes confirmant l'intérêt de la méthode.

Je décris maintenant mes activités d'enseignement.

À partir de mon recrutement et jusqu'à ma mutation à l'INSPÉ en 2017, je partagais mon temps d'enseignement, d'encadrement et d'administration entre les départements TIC, mathématiques et plus marginalement informatique de la Faculté des sciences et techniques de l'Université de Limoges.

Dans le département TIC j'ai rapidement pris des responsabilité de formation (DEUST Webmaster et DU Économie de l'Immatériel) et la responsabilité des unités d'enseignement dans lesquelles j'intervenais. J'enseignais principalement la conception et l'utilisation de bases de données (bases de données relationnelles et SQL), le web dynamique (PHP + bases de données), Javascript et la gestion de projets.

Pour le département de mathématiques, j'intervenais dans le DEA (j'assurais le cours de calcul formel), puis dans le Master CRYPTIS et enfin dans le Master ACSYON. J'ai participé à la conception de la maquette et introduit beaucoup de nouveaux cours dans le Master 1 CRYPTIS (calculabilité et complexité, systèmes polynomiaux en M1) et je suis intervenu dans beaucoup d'autres unités (codes correcteurs, algorithmique des corps finis - dans lequel j'interviens toujours - par exemple) mais aussi en M1 ACSYON (analyse algébrique). En M2 CRYPTIS, j'ai assuré longtemps le cours de programmation en C et une partie du cours de calcul formel, puis dans la nouvelle maquette j'ai enseigné une partie des outils mathématiques émergeants pour la cryptographie pour laquelle j'interviens toujours. J'interviens aussi pour l'enseignement de la cryptographie symétrique. J'ai créé trois unités d'enseignement pour le Master 2 ACSYON : optimisation semi-algébrique, conception géométrique assistée par ordinateur et calcul symbolique. J'ai assuré ce dernier cours jusqu'à l'an dernier au changement de maquette, mais mes activités à l'INSPÉ avec un changement de maquette ne me permettant plus de proposer un nouveau cours dans l'esprit de la nouvelle maquette orientée vers les sciences des données.

Depuis 2017, j'enseigne à l'INSPÉ (à l'époque ÉSPÉ) pour la formation des enseignants en mathématiques. J'enseigne sur les deux années du Master MEEF parcours mathématiques. J'ai été responsable du département de mathématiques de l'INSPÉ jusqu'en 2019. Depuis 2019, je suis responsable de la formation des enseignants de mathématiques et j'ai participé à la conception de la maquette du Master qu'a impliquée la réforme de la formation initiale des enseignants. J'interviens aussi dans la formation des enseignants du premier degré et j'encadre tous les ans des mémoires à ce titre. J'interviens en licence pour les étudiants se destinant à l'enseignement en premier degré à la Faculté des sciences et techniques (ce qui fut également une de mes contributions à la Faculté des lettres pendant trois années au cours desquelles j'ai dispensé une cinquantaine d'heures de cours).

En 2011, à la demande de Robert Cabanne, j'ai mis en place une formation des enseignants pour la création de l'option Informatique et Sciences du Numérique en Terminale. Le temps obtenu par un CRCT donné par le CNU en 2011 a été entièrement consommé par cette action qui a duré jusqu'en 2016 pour le Ministère de l'Éducation Nationale et jusqu'en 2018 pour l'Agence de l'Enseignement Français à l'Étranger. J'ai proposé un dispositif de formation à distance qu'il était facile d'hybrider et beaucoup de rectorats ont utilisé la plateforme (https://isn.unilim.fr) pour mettre en place des sessions de formation et de certification, et j'ai souvent assuré la formation et la certification à distance : ce fut le cas pour une cinquantaine de sessions, qui m'ont permis de certifier presque 2000 professeurs pour cet enseignement.

De 2013 à 2017, je suis intervenu au Vietnam à l'USTH dans le cadre du Master STIC pour assurer un cours de mathématiques pour ingénieurs (module de trente heures que je distribuais sur une période de 2 semaines à un mois selon le temps dont je disposais) et pour former sur place des assistants afin que ces derniers puissent par la suite assurer ces enseignements.

À partir de 2014, je suis devenu co-responsable du département TIC en cumulant ce rôle avec celui de chargé de mission au numérique de l'université de Limoges et de la Faculté des Sciences et Techniques. J'ai arrêté la charge de responsable du département TIC en 2017 et j'ai changé de poste par mutation interne en 2017 en rejoignant l'INSPÉ (à l'époque ÉSPÉ) de Limoges.

J'ai été chargé de mission au numérique de la Faculté des Sciences et Techniques puis de l'Université et j'ai assuré à ce titre la vice-présidence de l'Université numérique thématique des sciences UNICIEL pendant un mandat de quatre ans.

Je suis élu au CNU pour la 25ème section depuis 2015 et j'en ai été vice-président de 2016 à 2020.

1.2. Curriculum Vitae

Maître de conférences à l'université de Limoges depuis le 01/02/2005.

1.2.1. Titre

Thèse de doctorat spécialité Mathématiques-Informatique :
"Dualité algébrique, structures et applications"
soutenue publiquement à l'université d'Aix-Marseille II (INRIA Sophia Antipolis) le 23 septembre 2002. Directeur : Bernard Mourrain.

1.2.2. Expériences professionnelles

- Depuis le 01/02/2005 Maître de conférences à l'université de Limoges (à la FST de 2005 à 2016 et l'INSPÉ depuis).
- Du 01/10/2004 au 30/01/2005 Bourse post-doctorale de l'INRIA dans le cadre du réseau européen d'excellence Aim@Shape IST NoE 506766.
- 2003-2004 A.T.E.R. à l'Université de Nice-Sophia Antipolis.
- 2002-2003 Bourse post-doctorale à l'université de Pise (Italie) dans le cadre du réseau européen Real Analytic and Algebraic Geometry (6 mois) et d'un programme « Algèbre commutative effective » de l'université de Pise (6 mois).

1.2.3. Formation

- 1999-2002 Allocation de recherche
  Thèse de doctorat spécialité Mathématiques-Informatique :
  "Dualité algébrique, structures et applications".
- 1998-1999 DEA de Mathématiques discrètes et fondements de l'informatique à l'Université d'Aix-Marseille II - Luminy.
- 1997-1998 Maîtrise d'Ingénierie Mathématique, option Informatique, Université de Nice-Sophia Antipolis.
  Maîtrise de Mathématiques fondamentales, Université de Nice-Sophia Antipolis.

1.2.4. Publications

1.2.4.1. Publications dans des revues internationales

- N. Aragon, P. Gaborit, A. Hauteville, O. Ruatta, G. Zémor, **Low Rank Parity Check Codes: New Decoding Algorithms and Applications to Cryptography**., IEEE Trans Inf Theory, vol. 65, no 12, p. 7697-7717, 2019, doi: 10.1109/TIT.2019.2933535.
- P. Gaborit, O. Ruatta, et J. Schrek, **On the complexity of the rank syndrome decoding problem**, IEEE Trans. Inf. Theory, vol. 62, no 2, p. 1006-1019, 2016, doi: 10.1109/TIT.2015.2511786.

- Pierre Bonnelie, Loïc Bourdin, Fabien Caubet, Olivier Ruatta. **Flip procedure in geometric approximation of multiple-component shapes - Application to multiple-inclusion detection**. SMAI Journal of Computational Mathematics, Société de Mathématiques Appliquées et Industrielles (SMAI), 2016, 2, pp.255-276.
- Philippe Gaborit, Olivier Ruatta et Julien Schreck. **On the complexity of Syndrome Decoding Problem**. IEEE Transaction on Information Theory, 62(2), p. 1006-1019, 2016.
- Olivier Ruatta, Agnes Szanto et Mark Sciabica. **Overdetermined Weierstrass iteration and distance to consistent systems**. Theoretical Computer Science, 562, p.346–364, 2015.
- D.-N. Diatta, B. Mourrain, O. Ruatta. **Computing the topology of real algebraic surfaces with certainty**. Journal of Symbolic Computation, 47(8), p.903-925, 2012.
- B. Mourrain, V.Y. Pan et Olivier Ruatta. **Accelareted solution of multivariate polynomial systems of equations**. SIAM J. Comp., 32(2):435-454, 2003.
- B. Mourrain et Olivier Ruatta. **Relation between roots and coefficients, interpolation and application to system solving**. Journal of Symbolic Computation, 33:679-699, 2002.

1.2.4.2. Actes de congrès internationaux avec Comité de Lecture sur le texte complet (referees anonymes)

- Berger, Thierry P; Gueye, Anta Niane; Gueye, Cheikh Thiecoumba; Hasan, M Anwarul; Klamti, Jean Belo; Persichetti, Edoardo; Randrianarisoa, Tovohery H; Ruatta, Olivier, **Security Analysis of a Cryptosystem Based on Subspace Subcodes**, Code-Based Cryptography Workshop, Springer, 2021/6/21.
- T. P. Berger, C. T. Gueye, J. B. Klamti, et O. Ruatta, **Designing a Public Key Cryptosystem Based on Quasi-cyclic Subspace Subcodes of Reed-Solomon Codes**, vol. 1133 CCIS. 2019, p. 113. doi: 10.1007/978-3-030-36237-9_6.
- Magali Bardet, Pierre Briaud, Maxime Bros, Philippe Gaborit, Vincent Neiger, Olivier Ruatta, Jean-Pierre Tillich,**An algebraic attack on rank metric code-based cryptosystems**, in Advances in cryptology—EUROCRYPT 2020. Part III, copyright 2020, vol. 12107, p. 64-93. doi: 10.1007/978-3-030-45727-3_3.
- T. P. Berger, P. Gaborit, et O.Ruatta, **Gabidulin matrix codes and their application to small ciphertext size cryptosystems**, in Progress in cryptology—INDOCRYPT 2017,˙2017, vol. 10698, p. 247-266. doi: 10.1007/978-3-319-71667-1_13.
- Abraham Wendyida Kabore, Vahid Meghdadi, Jean-Pierre Cances, Philippe Gaborit, Olivier Ruatta. **Performance of Gabidulin Codes For Narrowband PLC Smart Grid Networks**. IEEE ISPLC 2015, 262-267.
- Philippe Gaborit, Olivier Ruatta, Julien Schrek, Gilles Zémor: **RankSign: An Efficient Signature Algorithm Based on the Rank Metric**. PQCrypto 2014[1.1]: 88-107
- Philippe Gaborit, Olivier Ruatta, Julien Schrek, Gilles Zémor: **New Results for Rank-Based Cryptography**. AFRICACRYPT[1.2] 2014: 1-12
- O. Ruatta. **On the Geometry and the Deformation of Shapes Represented by Piecewise Continuous Bézier Curves with Application to Shape Optimization**. LNCS 8085 GSI[1.3] 2013: 112-119.
- Philippe Gaborit, Olivier Ruatta, Julien Schrek et Gilles Zémor. **Low rank parity check codes and their application to cryptography**. Proc. WCC 2013, Bergen (Norway).

1.1. PQCrypto (Post-Quantum Cryptography) est une conférence spécialisée et de référence mondiale sur la cryptographie post-quantique.

1.2. EUROCRYPT et AFRICACRYPT sont des conférences internationales très sélectives (taux d'acceptation autour de 15%) et généralistes.

1.3. GSI (Geometric Science of Information) est une conférence annuelle de référence mondiale sur les statistiques sur les variétés différentielles.

- P. Boito et Olivier Ruatta. **Generalized companion matrix for approximate GCD**. Proc. of the 4th International Workshop on Symbolic-Numeric Computation (San Jose, CA), p. 74-80, ACM 2011.
- D. Boucher, P. Gaborit, Geiselmann, O. Ruatta et F. Ulmer. **Key exchange and encryption schemes based on non-commutative skew polynomials**. Proc. PQCrypto 2010, Lecture Notes in Computer Sciences, Springer, 2011.
- D.-N. Diatta, B. Mourrain et O. Ruatta. **On the Computation of the Topology of a Non-Reduced Implicit Space Curve**. ISSAC'08[1.4], Linz (Austria), 2008.
- F. Armknecht, C. Carlet, P. Gaborit, S. Künzli, W. Meier et O. Ruatta. **Efficient Computation of Algebraic Immunity for Algebraic Fast Algebraic Attack**. EUROCRYPT 06, Saint Petersburg, Russia.
- Olivier Ruatta. **A multivariate Weierstrass's rootfinder**. ISSAC'01, p. 276-283, London (Ontario), 2001.
- B. Mourrain et Y. V. Pan et O. Ruatta. **Asymptotic acceleration of solving multivariate polynomial systems of equations**. In Proceedings of Smale fest 2000, Foundations of Computational Mathematics, pp. 267-294. World Scientific, NewJersey, London, Singapore, Hong Kong, 2002.

1.2.4.3. Publication de résumés dans des conférences avec comité de lecture

- A. Dia, C. Durousseau, C. Menudier, L. Carpentier, O. Ruatta, et S. Bila, **Optimisation de formes de circuits hyperfréquences par un paramétrage utilisant des courbes de Bézier couplées à une méthode de gradients**, Caen, France, mai 2019. Consulté le: juill. 13, 2021. [En ligne]. Disponible sur: https://hal-unilim.archives-ouvertes.fr/hal-02094287
- O. Ruatta et V. D. Hoang, **Normal forms of parametrizations of curves and distance between curves**, Arcachon, juin 28, 2018. Consulté le: juill. 14, 2021. [En ligne]. Disponible sur: https://cs2018.sciencesconf.org/209108
- Stéphane Bila, Pierre Bonnelie, Olivier Ruatta et Satafa Sanogo. **Free-Form Method for Designing Optimal Microwave filter**. Symposium on Electric and Magnetic Fields (EMF 2016) to be held in Lyon on April 12-14 2016.
- F. Armknecht, P.-L. Cayrel, P. Gaborit, O. Ruatta. **Improved algorithm to find equations for algebraic attacks for combiners with memory**. BFCA'07.
- O. Ruatta. **Computing topology of intersection curve of two parametrized surfaces**. Computational Algebraic Geometry and Applications, 2006, Nice, France.
- P. Gaborit et O. Ruatta. **Improved Hermite multivariable polynomial interpolation**. ISIT 2006.
- P.Gaborit et O. Ruatta. **Efficient errasure list-decoding of Reed-Muller codes**. ISIT 2006.
- O. Ruatta. **A multivariate interpolation scheme**. EACA 04, Santander.
- O. Ruatta, M. Sciabica et A. Szanto. **Over-constrained Weierstrass iteration and the distance to consistent systems**. EACA 04, Santander.

1.2.4.4. Rapport de soumission à des concours internationaux (NIST avec comité de lecture)

- Nicolas Aragon, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Adrien Hauteville, Olivier Ruatta, Jean-Pierre Tillich, Gilles Zémor, **LAKE-Low rAnk parity check codes Key Exchange**, 2017. https://hal.archives-ouvertes.fr/hal-01946967

1.4. ISSAC (International Symposium On Symbolic and Algebraic Computation) est la conférence annuelle de référence mondiale en Calcul Formel.

- Nicolas Aragon, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Adrien Hauteville, Olivier Ruatta, Jean-Pierre Tillich, Gilles Zémor, **LOCKER - LOw rank parity ChecK codes EncRyption.** p. http://nicolas-aragon.fr/locker/, nov. 2017. Consulté le: juill. 14, 2021. [En ligne]. Disponible sur: https://hal.archives-ouvertes.fr/hal-01946962
- Nicolas Aragon, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Adrien Hauteville, Olivier Ruatta, Jean-Pierre Tillich, Gilles Zémor, Carlos Aguilar Melchor, Slim Bettaieb, Loïc Bidoux, Bardet Magali, Ayoub Otmani, **ROLLO (merger of Rank-Ouroboros, LAKE and LOCKER)**, https://pqc-rollo.org/doc/rollo-nist-conference_2019-08-24.pdf

1.2.4.5. Posters

- O. Ruatta, M. Sciabica, A. Szanto. **Over-constrained Weierstrass iteration and the distance to consistent systems**. Présenté durant ISSAC 2004 (prix ACM du meilleur poster).
- D.N. Diatta, B. Mourrain, O. Ruatta. **Computing the topology of non-reduced space curves**. Présenté durant ISSAC 2007.

1.2.5. Encadrement

5 thèses (2 soutenues et 3 en cours).

- Thèse de Daouda-Niang Diatta, à 50%, “Calcul de la topologie de courbes et surfaces algébriques réelles”, soutenue en septembre 2009. Cette thèse a donné lieu à deux publications. Daouda est professeur à l’Université Assane Seck de Ziguinchor (Sénégal).
- Thèse de Gaëtan Murat (professeur agrégé) à 50%, sur les « Résultants de polynômes de Ore et Cryptosystèmes de McEliece sur des Codes Rang Faiblement Structurés », début en 2010, soutenu le 9/12/2014. Cette thèse a aboutie à 2 publications. Gaëtan est professeur en lycée en Corrèze.
- Thèse de Pierre Bonnelie à 50% en co-encadrement avec Stéphane Bila et Paul Armand sur l’optimisation de formes par la méthode des contours libres avec application à la conception de filtres électromagnétiques. Soutenue le 13/02/2017.
- Thèse de Adrien Hauteville à 20% sur la cryptographie basée sur les codes en métrique rang. Cette thèse à déjà donné lieu à 4 publications. Soutenue le 04/12/2017.
- Thèse de Van Duc Hoang à 70%, bourse du programme franco-vietnamien 911 en co-encadrement avec Jacques-Arthur Weil. Soutenue le 12/02/2020.

3 post-doctorants :

- Post-doctorat de Marie-Ève Modolo (2011), “Modélisation de frontière de région plane par des courbes paramétrées et optimisation de formes”. Marie-Ève est PRAG dans une école d’ingénieur.
- Post-doctorat de Guillaume Quintin (2013) sur « interpolation multivarié en cryptanalyse », classé deuxième au CNRS qui a fait le choix de l’industrie.
- Post-doctorat de Satafa Sanogo 2015-2016 sur la modélisation de problèmes d’électromagnétisme et l’implantation de ces modèles dans un logiciel libre (FreeFem++).

33 mémoires de master (hors INSPÉ) : 13 M2 et 20 M1

- Chaque année, j’ai encadré un à deux groupes pour « l’initiation à la recherche » en Master 1 Mathématiques (20 mémoires soutenus) jusqu’à ma mutation à l’INSPÉ.
- 7 mémoires de Master 2 ACSYON : 1 étudiant en 2011, 2 étudiants en 2012 et 1 étudiant en 2013, 1 étudiant en 2014, 2 étudiants en 2015 (tous soutenus).
- 3 mémoires de Master 2 CRYPTIS : 1 étudiant en 2005, 1 étudiant en 2006, 1 étudiant en 2009 (tous soutenus).

- 3 mémoires de DEA (tous soutenus).

Mémoire de Master MEEF : Chaque année, entre 5 et 10 mémoires pour le second degré et 2 à 3 mémoires pour le premier degré.

1.2.6. Jurys de thèse : 9

J'ai participé à deux jurys de thèse en Physique à l'Université de Limoges (Hassan Khalil en 2009 et Atousa Assadi-Haghi en 2007), un jury de thèse en robotique à l'Université d'Amiens (Pauline Merveilleux en 2012) et sept thèses en mathématiques (cinq à Limoges, une à Toulouse pour Satafa Sanogo en 2015 et une à Poitiers pour Marie-Ève Modolo en 2011).

1.2.7. Responsabilités

1.2.7.1. Responsabilités nationales pédagogiques

- Vice-président de l'Université Numérique Thématique UNICIEL de 2012 à 2016.
- Responsable de la plateforme de formation pour l'Informatique et Sciences du Numérique (lien vers la plateforme ISN: http://isn.unilim.fr), projet du Ministère de l'Education National (DGESCO) et de l'Université de Limoges (formation des professeurs à l'enseignement ISN en terminale, 1100 stagiaires en deux ans, 2011-2013).

1.2.7.2. Responsabilités nationales de recherche

Elu à la section 25 du CNU. Vice-Président rang B du bureau de la section 25 du CNU et élu à la CP-CNU pour le groupe 5 (regroupant les sections 25-26 et 27 du CNU) 2016-2020. Elu à la section 25 du CNU 2020-2024.

1.2.7.3. Responsabilités pédagogiques locales

- Responsable de la formation des enseignants de Mathématiques à l'INSPÉ de Limoges depuis 2020.
- Responsable du département de Mathématiques de l'INSPÉ de 2018 à 2020.
- Chargé de mission TIC à la Faculté des Sciences et Techniques (formation et promotion de dispositifs innovants : conversion de deux Masters en Formation Ouverte et à Distance en mathématiques et en informatique) de 2008 à 2014.
- Co-responsable du département d'enseignement TIC de la Faculté des Sciences et Techniques de l'Université de Limoges de 2008 à 2012.
- Responsable du DU Economie de l'Immatériel (2008-2011).
- Correspondant pour Limoges à l'Université des Sciences et Technologies d'Hanoi depuis 2012, responsable du module « Mathematics for ICT » du master ICT.

1.2.7.4. Responsabilités de recherche au laboratoire XLIM

- Responsable de l'équipe CRYPTIS (depuis février 2022)
- Correspondant XLIM pour le RTP Éducation du CNRS.
- Membre du Conseil Animation Scientifique Interdisciplinaire de XLIM (depuis 2016)
- Elu pour le département DMI au conseil du laboratoire XLIM (2012-2018).
- Correspondant à Limoges du projet « Géométrie Algébrique, Arithmétique et Cryptographie » de l'axe « Mathématiques et applications » de la fédération MIRES (Poitiers-Limoges-La Rochelle) depuis la création de MIRES jusqu'en 2018.
- Responsable du colloquium du département Mathématiques-Informatique de XLIM 2007-2013.
- Responsable du séminaire de Calcul Formel de 2006 à 2008.

1.2.8. Enseignements

1.2.8.1. Enseignement d'informatique au département TIC

- Langage du web niveau 2 (DEUST)
- Web dynamique (DEUST et Licence pro)
- Organisation des données (DEUST)
- Ingénierie de projet et d'avant-projet (DEUST et Licence pro de 2006 à 2012)

1.2.8.2. Enseignement en Master 1 pour les départements de mathématiques et d'informatique

- Algèbre approfondie - Représentations des groupes finis - (Master 1, pendant 4 ans, 3 crédits, de 2005 à 2008)
- Systèmes polynomiaux (cours Master 1 mathématiques, 3 crédits, depuis 2008). Il s'agit d'un cours que j'ai créé.
- Calculabilité, complexité et évaluation de performance (cours Master 1 mathématiques et master 1 informatique, 3 crédits, 2006-2015). Il s'agit d'un cours pluri-disciplinaire que j'ai créé.
- Symbolic Analysis (cours Master 1 ACSYON, 3 crédits, en anglais, 2012-2021). Il s'agit d'un cours que j'ai créé.
- Théorie des codes (30h en 2016 et 2017)
- Algorithmique des corps finis (30h depuis 2019)
- Tous sujets disciplinaires à l'INSPÉ de Limoges.

1.2.8.3. Enseignements en Master 2 pour le département de mathématiques

- Cryptographie à clé privée (Master 2 Cryptis 2022)
- Développement de logiciels cryptographiques (Master 2 CRYPTIS, 3 crédits) de 2005-2011.
- Calcul formel (Master 2 avec Moulay Barkatou, pendant trois ans, 6 crédits, 2006-2007-2008).
- Programmation pour le Calcul Scientifique (M2 ACSYON, 2012).
- Calcul Symbolique-Numérique (Master 2 ACSYON, en anglais, 3 crédits, 2009-2021).
- Courbes et Surfaces pour la CAO (Master 2 ACSYON avec Benoit Crespin, 3 crédits, 2009-2016). Il s'agit d'un cours que j'ai créé.

1.2.8.4. Enseignement en Licence pour les départements de mathématiques et d'informatique

- TD de systèmes différentiels (cours assuré par M. Barkatou en 2006-2007).
- Cours-TD parcours Préparation aux concours (L2 et L3, 2009-2010). Il s'agit d'un cours que j'ai créé.
- Informatique 1 (1 groupe de cours/TD au premier semestre du portail SI depuis 2014).
- Licence SAE : TIC et multimédia (L3, depuis 2015). Il s'agit d'un cours en pédagogie inversée que j'ai créé.
- Licence PPPE : TICE (L1 depuis 2022).

1.2.8.5. Enseignement à l'INSPÉ

J'interviens dans l'ensemble des enseignements disciplinaires et sur la formation à la recherche depuis mon recrutement à l'INSPÉ.

1.2.8.6. Enseignement à l'étranger

- 2012 : Université Scientifique et Technique de Phnom Penh (Cambodge), Master 1 : Géométrie des courbes et des surfaces (20 heures, en anglais).
- 2012-2017 : Université Scientifique et Technique de Hanoi (Vietnam), Master 1 STIC : Mathématiques appliquées (50 heures, en anglais).
- Depuis 2013 : chargé de la formation ISN (Informatique et Sciences du Numérique) des professeurs de l'Agence de l'Enseignement Français à l'Étranger (Ministère des affaires étrangères, stages annuels de 90 heures entièrement en formation à distance en français).

1.2.9. Projet et contrats

1.2.9.1. Projets auprès d'agences de moyens

- Responsable scientifique local de l'ANR CBCRYPT (Code Based Cryptography 2018-2022)
- Membre de l'ANR Gecko (Géométrie de la complexité, 2005-2008).
- Membre de l'ANR XCODE (2006-2009).
- Co-responsable du projet Carnot « Modélisation paramétrique polynomiale par morceaux de frontières pour l'optimisation de formes », 2008-2009.
- Responsable du projet régional « Traitement d'images et problèmes inverses pour la caractérisation de matériaux de construction (IMAT) » répondant à l'Appel à Projet « Thématique » de la région 2016 inter-instituts (XLIM-MATHIS et IPAM-GEMH).

1.2.9.2. Contrats industriels

- Contrat industriel avec la société APTICOD sur la calibration d'un codeur optique (2006).
- Contrat industriel avec le CNES sur l'implantation d'un nouvel algorithme de décodage en liste "souple", 2006.
- Contrat avec le CNES (co-responsable avec Stéphane Bila) sur la conception de filtres électromagnétiques optimaux pour 3 ans (lancement décembre 2015).

1.2.10. Responsabilités éditoriales

Je réalise régulièrement des rapports d'expertise pour des revues internationales (Journal of Symbolic Computation, IEEE IT, Journal of Algebra, Theoretical Computer Sciences, Computer Aided Geometric design) et des conférences internationales (ISSAC, SNC, CRYPTO, EUROCRYPT, ...). J'accepte de rapporter entre 5 et 10 articles par an.

J'ai été membre du Comité FUSCIA de l'INRIA (comité éditorial de formation scientifique de l'INRIA) et chargé de mission pour l'UNT UNISCIEL chargé du C2i et de l'articulation mathématique/informatique et bac -3/bac +3.

1.2.11. Organisation et animation de manifestations scientifiques

- Co-organisateur d'une école d'été à Sophia-Antipolis du 5 au 9 septembre 2005 sur les logiciels libre pour les calculs algébriques et géométriques.
- Co-organisateur des premières Journées Matrices Structurées qui ont eu lieu à Limoges les 18 et 19 janvier 2006. Colloque pérennisé depuis (bi-annuel).
- Co-organisateur des Journées Nationales de Calcul Formel 2007 (et mise au format d'école thématique du CNRS), qui ont eu lieu à Marseille du 29 janvier au 2 février 2007.
- Co-organisateur de l'école CIMPA Méthodes effectives et logiciels de la logique et de l'algèbre pour la géométrie algébrique et la cryptographie du 24 août au 4 septembre 2009 à Yaoundé (Cameroun).

1.2.12. Réalisations de logiciels

- Bases de données (MySql) avec interfaces web (PHP+HTML) : création du logiciel permettant la gestion du séminaire de l'équipe GALAAD de l'INRIA qui permet aux responsables du séminaire de soumettre, d'annoncer, de modifier un exposé où qu'ils soient et sans avoir à se connecter à une machine de l'INRIA où la base de données est hébergée.
- Participation au développement de SYNAPS, qui est une bibliothèque C++ d'outils collaboratifs orientée vers le calcul numérique-symbolique (`http://www-sop.inria.fr/galaad/logiciels/synaps/`). Cette bibliothèque s'est progressivement intégrée au logiciel Mathemagix.
- Participation à la bibliothèque Multires qui est une bibliothèque Maple pour la manipulation de polynômes multivariés, le calcul de résultants et de résidus ainsi que des méthodes de résolution de systèmes.
- Réalisation en collaboration avec P. Trébuchet d'un ensemble de fonctions pour les changements d'ordres dans les bases de Gröbner et l'implicitation d'hypersurfaces rationnelles (distribué avec le livrable 13.13 du workpackage 3.1 du projet GAIA 2, IST 200135512).
- Participation (marginale) à TeXmacs (http://www.texmacs.org), un logiciel WYSIWYG d'édition scientifique avancé.
- Participation à Mathemagix : j'en suis l'instigateur avec Joris Van der Hoeven, nous en avons écrit conjointement la première version en 2005 (http://www.mathemagix.org).

1.2.13. Autres

J'ai aussi réalisé des travaux comme évaluateur pour des Pôles de Compétitivité et des Commissions de Spécialistes.

Chapitre 2

Codes en métrique rang et cryptographie

Depuis quelques années, principalement en collaboration avec Philippe Gaborit, je me suis intéressé aux codes en métrique rang. Je rappellerai très sommairement ce que sont les codes en métrique rang. J'introduirai ensuite les outils nécessaires concernant les polynômes de Öre et brièvement ensuite les codes LRPC, leur décodage et certains de leurs usages en cryptographie.

2.1. Codes en métrique rang

Dans tout ce qui suit $q = p^a$ est une puissance d'un nombre premier p, et $\mathbb{F}_q$ est « le » corps à q éléments. Je noterai généralement $\Bbbk = \mathbb{F}_q$ ce « corps de base » ou « petit corps ». Je définis maintenant la métrique rang :

Définition 2.1. *Soient V et W deux $\Bbbk$-espaces vectoriels de dimensions finies et soit $G = \mathcal{L}(V, W)$ l'espace vectoriel des applications linéaires de V dans W, on définit alors d_r: $\begin{cases} G \times G \longrightarrow \mathbb{N} \\ (\varphi, \psi) \longmapsto \mathrm{rk}(\varphi - \psi) \end{cases}$ l'application qui à deux telles applications linéaires associe le rang de la différence de ces applications.*

La métrique rang est une distance :

Proposition 2.2. *L'application d_r est une distance sur $\mathcal{L}(V, W)$.*

Je peux maintenant définir les codes pour la métrique rang.

Définition 2.3. *Un ensemble $\mathcal{C}$ est appelé code rang ou code pour la métrique rang s'il existe deux $\Bbbk$-espaces vectoriels de dimensions finies V et W tels que $\mathcal{C}$ est un $\Bbbk$-sous-espace vectoriel de $\mathcal{L}(V, W)$.*

Le cas le plus fréquent est celui où $\mathcal{C}$ est un sous-espace vectoriel de $\mathcal{M}_{n\times m}(\Bbbk)$ l'espace vectoriel des matrices $n \times m$ à coefficients dans $\mathbb{F}_q$. Dans ce cas on parle de codes matriciels en métrique rang. Une situation qui va revenir souvent est la suivante : je considère $\mathbb{F}_{q^m}$ une extension de degré m de $\mathbb{F}_q$ et on considère un $\mathbb{F}_q$-sous-espace vectoriel de $(\mathbb{F}_{q^m})^n$. Je considère $\zeta_1, \cdots, \zeta_m$ une $\mathbb{F}_q$-base de $\mathbb{F}_{q^m}$, tout élément $u \in \mathbb{F}_{q^m}$ s'écrit alors $u = \sum_{i=1}^{l} u_i \cdot \zeta_i$. Je vais associer à tout élément $\boldsymbol{z} = (z_1, \cdots, z_n) \in (\mathbb{F}_{q^m})^n$ une matrice $M_{\boldsymbol{z}}$ de la façon suivante : chaque coordonnée z_j de $\boldsymbol{z}$ s'écrit $z_i = \sum_{i=1}^{l} z_{i,j} \cdot \zeta_i$ et je lui associe le vecteur colonne $\begin{pmatrix} z_{i,1} \\ \vdots \\ z_{i,l} \end{pmatrix}$ qui sera pris comme colonne d'indice j de $M_{\boldsymbol{z}}$.

$$M_{\boldsymbol{z}} = \begin{array}{c} \\ \zeta_1 \\ \vdots \\ \zeta_l \end{array} \begin{array}{c} \begin{array}{ccc} z_1 & \cdots & z_n \end{array} \\ \begin{pmatrix} z_{1,1} & \cdots & z_{1,n} \\ \vdots & \ddots & \vdots \\ z_{l,1} & \cdots & z_{l,n} \end{pmatrix} \end{array}. \tag{2.1}$$

Ainsi, à tout sous-corps de $\mathbb{K}$ correspond un code matriciel qui peut être considéré comme code muni de la distance rang. J'appellerai ces codes des codes sous-espaces.

Remarquons que tous les codes matriciels linéaires peuvent être réalisés comme codes sous-espaces. Soient $M_1, \cdots, M_k \in \mathcal{M}_{n\times m}(\mathbb{F}_q)$ une base de $\mathcal{C}$ et soient $\zeta_1, \cdots, \zeta_m \in \mathbb{F}_{q^m}$, on a $M_l = \left(m_{i,j}^{(l)} \right)_{\substack{1\leqslant i\leqslant m \\ 1\leqslant j\leqslant n}}$ pour tout $l \in \{1, \cdots, k\}$, on note alors $m_i^{(l)} = \sum_{j=1}^{m} m_{j,i}^{(l)} \cdot \zeta_j \in \mathbb{F}_{q^m}$. Ainsi, on associe à chaque M_l pour $l \in \{1, \cdots, k\}$ un vecteur $m^{(l)} = \left(m_1^{(l)}, \cdots, m_n^{(l)}\right) \in (\mathbb{F}_{q^m})^n$ tel qu'on puisse prolonger cette identification en un isomorphisme $\mathbb{F}_q$-linéaire de $\mathcal{C}$ sur le $\mathbb{F}_q$-sous-espace vectoriel de $(\mathbb{F}_{q^m})^n$ engendré par $m^{(1)}, \cdots, m^{(k)}$ et dans ce cas c'est bien le même ensemble de matrices quand on regarde les matrices pour chaque code. Je me consacrerai donc aux codes sous-espaces.

2.1.1. Bornes classiques pour les codes sous-espaces

Pour la métrique rang sur les codes sous-espaces, on retrouve un certain nombre de bornes classiques jouant un rôle analogue à leurs homologues en métrique de Hamming. La première est la généralisation de la borne du singleton qui permet de caractériser la capacité de détection et de correction d'un code.

PROPOSITION 2.4. *Soit $\mathcal{C} \subset (\mathbb{F}_{q^m})^n$ un code en métrique rang de distance minimale d, alors :*

$$|\mathcal{C}| \leqslant q^{\min(m\cdot(n-d+1), m\cdot(n-d+1))}.$$

Dans le cas où $\mathcal{C}$ est un code linéaire avec $n \leqslant m$ cette inégalité donne :

$$d \leqslant n - k + 1$$

et dans le cas où $n > m$ on obtient :

$$d \leqslant 1 + \left\lfloor \frac{(n-k)\cdot m}{n} \right\rfloor. \tag{2.2}$$

DÉFINITION 2.5. *Soit $\mathcal{C} \subset (\mathbb{F}_{q^m})^n$ un code en métrique rang tel que $m \geqslant n$ qui atteint la borne $|\mathcal{C}| = q^{m\cdot(n-d+1)}$ et donc $d = n - k + 1$, alors $\mathcal{C}$ sera dit MRD pour Maximum Rank Distance.*

La notion de code MRD est la généralisation au cas des codes en métrique rang de celle de code MRD pour la distance de Hamming.

La seconde borne est la borne de Gilbert-Varshamov pour la métrique rang. Il y a plusieurs interprétations de cette borne. C'est conjecturalement la distance minimale atteinte par un code aléatoire. C'est aussi une borne permettant de déterminer des valeurs de paramètres pour lesquelles des codes linéaires existent mais c'est également l'entier à partir duquel la solution au problème de décodage n'est plus unique (l'espérance du nombre de solutions dépasse 1).

On note $\mathcal{S}(n, m, q, t)$ le nombre d'éléments dans la sphère « rang » de rayon t dans $(\mathbb{F}_{q^m})^n$, c'est-à-dire le nombre de matrices $m \times n$ à coefficients dans $\mathbb{F}_q$. Pour $t = 0$, on a $S(n, m, q, 0) = S_0 = 1$ et dans le cas général, on a [36] :

$$S(n, m, q, t) = \prod_{i=0}^{t-1} \frac{(q^n - q^i)\cdot(q^m - q^i)}{q^t - q^i}.$$

J'introduis maintenant la notion de binôme de Gauss qui généralise celle de binôme de Newton.

DÉFINITION 2.6. *On note $\begin{bmatrix} m \\ t \end{bmatrix}_q = \prod_{i=0}^{t-1} \frac{q^m - q^i}{q^t - q^i}$ le nombre de sous-espaces vectoriels sur $\mathbb{F}_q$ de $\mathbb{F}_{q^m}$ de dimension t.*

Ces coefficients sont très importants dans l'énumération des sous-espaces vectoriels et donc dans l'évaluation de probabilités dans le cadre de la métrique rang.

À partir du nombre d'éléments sur les « sphères », nous pouvons maintenant évaluer le nombre d'éléments dans les « boules » :

$$B(n,m,q,d)=\sum_{i=0}^{d} S(n,m,q,i).$$

Proposition 2.7. *Si $B(n,m,q,d)<q^{m\cdot(n-k)}$ alors il est possible de construire un code en métrique rang de longueur n, de dimension k et de distance minimale d sur $\mathbb{F}_{q^m}$.*

Définition 2.8. *La borne de Gilbert-Varshamov pour les codes en métrique rang $\mathbb{F}_q$-linéaire sur $\mathbb{F}_{q^m}$, notée $\mathrm{GVR}(n,k,m,q)$ est le plus petit entier d tel que $B(n,m,q,d)\geqslant q^{m\cdot(n-k)}$.*

Proposition 2.9. *[29] Lorsque n et m tendent vers l'infini, on a un équivalent asymptotique de $\mathrm{GVR}(n,k,m,q)$:*

$$\mathrm{GVR}(n,k,m,q)\backsim\frac{m+n-\sqrt{(m-n)^2+4\cdot k\cdot m}}{2}$$

qui se simplifie de la façon suivante lorsque $n=m$:

$$\mathrm{GVR}(n,k,n,q)\sim n\cdot\left(1-\sqrt{\frac{k}{n}}\right).$$

2.1.2. Problèmes de décodage

Dans toute cette section, $\mathcal{C}$ est un code sous-espace linéaire en métrique rang de dimension k sur $\mathbb{F}_q$ dans $\mathbb{F}_{q^m}$ et de distance minimale d.

Soit G une matrice $n\times k$ à coefficients dans $\mathbb{F}_{q^m}$ de rang k telle que $G(\mathbb{F}_{q^k})=\mathcal{C}\subset(\mathbb{F}_{q^m})^n$. On dit que G est une matrice génératrice de $\mathcal{C}$. Un mot de code s'obtient alors en multipliant un vecteur ligne $\boldsymbol{x}=(x_1,\cdots,x_k)\in\mathbb{F}_q^k$ par G, i.e. $\mathcal{C}=\{\boldsymbol{x}\cdot G\mid\boldsymbol{x}\in\mathbb{F}_q^k\}\subset(\mathbb{F}_{q^m})^n$.

Définition 2.10. *On appelle matrice de parité d'un code sous-espace linéaire $\mathcal{C}$ de matrice génératrice G toute matrice $H\in\mathcal{M}_{(n-k)\times n}(\mathbb{F}_{q^m})$ de rang plein telle que $H\cdot G=0$.*

Proposition 2.11. *Soit H une matrice de parité d'un code sous-espace linéaire $\mathcal{C}$ et soient $y\in\mathcal{C}$ et $e\in(\mathbb{F}_{q^m})^n$ alors $H\cdot(y+e)^T=H\cdot e^T$. Le vecteur $H\cdot e^T$ est appelé syndrome (associé à e).*

Problème 2.1. Soit $\mathcal{C}$ un code sous-espace linéaire de dimension k sur $\mathbb{F}_q$ dans $(\mathbb{F}_{q^m})^n$ et de matrice génératrice G, soit $\boldsymbol{y}=\boldsymbol{x}\cdot G\in\mathcal{C}$ et $\boldsymbol{e}\in(\mathbb{F}_{q^m})^n$, le problème de décodage consiste à retrouver $\boldsymbol{y}$ connaissant $\boldsymbol{y}+\boldsymbol{e}$.

Clairement, si $\mathcal{C}$ est de distance minimale d et si $\mathrm{rg}(M_{\boldsymbol{e}})<\frac{d}{2}$ alors il existe une unique solution au problème de décodage 2.1.

Problème 2.2. Soit $\mathcal{C}$ un code sous-espace linéaire de dimension k sur $\mathbb{F}_q$ dans $(\mathbb{F}_{q^m})^n$, de matrice de parité H, soit $\boldsymbol{e}\in(\mathbb{F}_{q^m})^n$, le problème de syndrome consiste à retrouver $\boldsymbol{e}$ connaissant $H\cdot\boldsymbol{e}^T$.

De la même façon, si $\mathcal{C}$ est de distance minimale d et si $\mathrm{rg}(M_{\boldsymbol{e}})<\frac{d}{2}$ alors il existe une unique solution au problème du syndrome 2.2.

Proposition 2.12. *Les problèmes 2.1 et 2.2 sont équivalents.*

Le problème de décodage au maximum de vraisemblance consiste à retrouver le mot de code le plus proche du mot reçu.

Problème 2.3. Soit $\mathcal{C}$ un code sous-espace linéaire de dimension k sur $\mathbb{F}_q$ dans $(\mathbb{F}_{q^m})^n$ et de matrice génératrice G, soit $\boldsymbol{y}=\boldsymbol{x}\cdot G\in\mathcal{C}$ et $\boldsymbol{e}\in(\mathbb{F}_{q^m})^n$, le problème de décodage au maximum de vraisemblance consiste à retrouver un mot $\boldsymbol{z}\in\mathcal{C}$ le plus proche (pour la distance rang) de $\boldsymbol{y}+\boldsymbol{e}$.

Ce problème revient, en toute généralité, à celui-ci :

Problème 2.4. Soit $\mathcal{C}$ un code sous-espace linéaire de dimension k sur $\mathbb{F}_q$ dans $(\mathbb{F}_{q^m})^n$, de matrice de parité H, soit $\boldsymbol{e} \in (\mathbb{F}_{q^m})^n$ et $s = H \cdot e^T$, le problème consiste à trouver un vecteur de $\boldsymbol{z} \in (\mathbb{F}_{q^m})^n$ de plus petit poids rang tel que $H \cdot \boldsymbol{z}^T = \boldsymbol{s}$.

Les analogues de ces problèmes en métrique de Hamming sont connus pour être difficiles (NP-complet) mais il n'existe pas de preuve dans le cas de la métrique rang, néanmoins il existe une réduction probabiliste du cas Hamming au cas de la métrique rang ce qui conforte l'hypothèse que ces problèmes sont difficiles.

Dans la suite de ce chapitre, nous proposerons des algorithmes pour traiter ces problèmes dans des cas où ils peuvent être résolus efficacement et nous proposerons des méthodes pour masquer l'existence de tels algorithmes pour concevoir des primitives cryptographiques.

Problème 2.5. Soit $\mathcal{C}$ un code de dimension k, de longueur n et de distance minimale d, le problème de décodage en liste consiste à retrouver tous les mot de code à distance l d'un mot $\boldsymbol{z}$ de $\mathbb{F}_q^n$.

Si $l \leqslant \frac{d-1}{2}$ alors la liste est de longueur 1, mais lorsque $l \geqslant \frac{d}{2}$ alors il peut y avoir plusieurs mots du code à distance l. Si l n'est pas trop devant $\frac{d}{2}$, alors il y aura « peu » de mots et plus l grandit plus il y a de mots de codes dans la boule de centre $\boldsymbol{z}$ et de rayon l. Nous allons maintenant nous intéresser à l'évaluation qualitative de $\sharp(B(\boldsymbol{z},l) \cap \mathcal{C})$ en fonction de l où $B(\boldsymbol{z},l)$ est la boule de centre $\boldsymbol{z}$ et de rayon l. Au début il y a 0 ou 1 élément quand $l \leqslant \frac{d-1}{2}$ puis que $\sharp(B(\boldsymbol{z},l) \cap \mathcal{C})$ deviendra « polynomial » pour enfin adopter un régime « exponentiel » dans le cadre d'un code linéaire pour la distance de Hamming. Ces résultats dus à Sudan [48] puis avec Guruswami [28] ont conduit à de nombreuses publications sur le décodage en liste dans le cas où la taille de liste est polynomiale. Pour faire du décodage en liste, un des outils principaux est l'interpolation polynomiale multivariée que j'avais abordée dans ma thèse et avec Philippe Gaborit nous avons utilisé des algorithmes rapides pour l'interpolation de Lagrange [23] et l'interpolation d'Hermite [25] pour traiter des problèmes de décodage en liste pour des codes algébriques (Reed-Solomon et Reed-Muller).

Comme nous le verrons plus tard, on peut voir les codes Gabidulin comme une généralisation des codes de Reed-Solomon et il y a beaucoup d'analogies entre les deux. Mais les codes de Gabidulin sont des codes intéressants pour la métrique rang. Nous avons donc eu comme objectif d'étudier le décodage en liste dans le cadre de la métrique rang et nous souhaitions proposer des algorithmes pour cela. Si nous avons échoué sur ce point, nous avons commencé à travailler sur les codes en métrique rang à partir de là avec beaucoup de plus succès.

Pour entrer dans le domaine des codes en métrique rang, comme nous voulions faire du décodage en liste, nous nous sommes beaucoup intéressés aux polynômes linéarisés pour lesquels les codes de Gabiulin sont des codes d'évaluation. C'est l'objet de la sous-section suivante.

2.1.3. Polynômes tordus et polynômes linéarisés

Polynômes tordus

J'expose ici les définitions qui me seront nécessaires par la suite concernant les polynômes tordus qui sont un cas particulier des polynômes de Öre. Je présente la théorie avec $\Bbbk$ un corps, $\mathbb{K}$ une extension de $\Bbbk$ et θ un $\Bbbk$-automorphisme de $\mathbb{K}$. Le cas qui nous intéressera est celui où $q = p^s$ est une puissance d'un nombre premier, $\Bbbk = \mathbb{F}_q$, $\mathbb{K} = \mathbb{F}_{q^a}$ et θ: $\begin{cases} \mathbb{F}_{q^a} \to \mathbb{F}_{q^a} \\ x \mapsto x^a \end{cases}$ est le Frobenius de $\mathbb{F}_{q^a}/\mathbb{F}_q$.

L'anneau des polynômes tordus est $\mathbb{K}[X]$ en tant qu'espace vectoriel mais on le munit du produit tordu donné par le produit de termes suivants : $(a \cdot X^i) \cdot (b \cdot X^j) = a \cdot \theta^j(b) \cdot X^{i+j}$ pour tout i et $j \in \mathbb{N}$ et tout a et $b \in \mathbb{K}$.

Dès que $\Bbbk$ est un corps commutatif, on a une division à droite qui est euclidienne. Pour le cas qui nous intéresse de $\mathbb{F}_q$ et $\mathbb{F}_{q^a}$, on a une division euclidienne à droite et à gauche. Nous verrons ci-dessous que ces polynômes tordus sont « isomorphes » aux polynômes linéarisés. Néanmoins, il y a deux raisons de les introduire ici : pour tester des conjectures, même sur les polynômes linéarisés, j'utilise la bibliothèque sur les polynômes tordus développée dans Sagemath et nous verrons un protocole basé sur la difficulté de la factorisation des polynômes tordus dans la sous-section 2.3.1.

Polynômes linéarisés

Je considère une extension $\mathbb{F}_{q^a}$ de degré a de $\mathbb{F}_q$ et je considère l'ensemble $\mathbb{F}_{q^a}\langle X^q\rangle = \left\{ p(X) \in \mathbb{F}_{q^m}[X] \mid \exists d \in \mathbb{N} \text{ et } a_0, \ldots, a_d \in \mathbb{F}_{q^a} \text{ t.q. } p(X) = \sum_{i=0}^{d} p_i X^{q^i} \right\}$ qui est le sous-$\mathbb{K}$-espace vectoriel de $\mathbb{K}[X]$ dont une base est $\{X^{q^i} \mid i \in \mathbb{N}\}$. Du fait que pour tout $i \in \mathbb{N}$, l'application $a \in \mathbb{K} \mapsto a^{q^i} \in \mathbb{K}$ est $\Bbbk$-linéaire, si je munis cet espace vectoriel de la composition comme loi « produit », j'obtiens un anneau non-commutatif, que je noterai $\mathbb{K}\langle X^q\rangle$, appelé anneau de q-polynômes ou anneau de polynômes linéaires (ou linéarisés). C'est une instance particulière d'anneaux de polynômes de Öre qui fut le premier à en faire une étude systématique dans [42].

L'anneau des q-polynômes $(\mathbb{K}\langle X^q\rangle, +, \circ)$ est isomorphe à l'anneau des polynômes de Öre $\mathbb{K}[X, \theta, 0]$ où 0 est la dérivation nulle et $\theta : u \in \mathbb{K} \to u^q \in \mathbb{K}$ est l'automorphisme de Frobenius de $\mathbb{K}$ sur $\Bbbk$. L'isomorphisme est simplement donné par $\sum_{i=0}^{d} u_i X^{q^i} \mapsto \sum_{i=0}^{d} u_i X^i$. Ceci permet d'utiliser l'expressivité du formalisme des q-polynômes pour faire des preuves, donner des interprétations mathématiques et de profiter des implantations existantes et efficaces sur les polynômes tordus.

Division euclidienne à droite

Je donne ici l'algorithme de calcul de la division euclidienne à droite dans un anneau de polynômes linéarisés. L'idée est une extension directe de la division des monômes. Soient a et $b \in \mathbb{K}$ et $i < j \in \mathbb{N}$, je cherche à diviser bX^{q^j} à droite par aX^{q^i} en commençant par remarquer que cela revient à chercher $c \in \mathbb{K}$ tel que $cX^{q^{j-i}} \circ aX^{q^i} = bX^{q^j}$. Or $cX^{q^{j-i}} \circ aX^{q^i} = c \cdot a^{q^{j-i}} X^{q^j}$. Je dois donc choisir c tel que $c \cdot a^{q^{j-i}} = b$ et donc $c = b \cdot a^{q^i - j}$. C'est à partir de cette idée que je décris l'algorithme d'Euclide pour la division à droite des q-polynômes.

Algorithme [Euclide] 2.1

Entrée : $A(X)$ et $B(X)$ de q-degrés respectifs d_A et d_B
Sortie : $Q(X)$ et $R(X)$ tels que $B(X) = Q(X) \circ A(X) + R(X)$ avec le q-degré de $R(X)$ strictement inférieur au q-degré de $A(X)$
$R(X) \leftarrow B(X)$
$Q(X) \leftarrow 0$
$d_R \leftarrow d_B$
Tant que $d_A \leqslant d_R$ **faire**
  $Q(X) \leftarrow Q(X) + \mathrm{lc}(R(X)) \cdot \mathrm{lc}(A(X))^{q^{d_A - d_R}} X^{q^{d_R - d_A}}$
  $R(X) \leftarrow R(X) - \left(\mathrm{lc}(R(X)) \cdot \mathrm{lc}(A(X))^{q^{d_A - d_R}} X^{q^{d_R - d_A}}\right) \circ A(X)$
  $d_R \leftarrow q - \text{degré de } R(X)$
Fin du tant que
Renvoyer $(Q(X), R(X))$

Cet algorithme est conçu pour qu'à chaque étape $B(X) = Q(X) \circ A(X) + R(X)$ (c'est un invariant de boucle) et l'algorithme s'arrête lorsque $d_R < d_A$, ce qui donne une preuve de la correction de l'algorithme.

La division euclidienne à droite a un rôle particulier par rapport aux noyaux. En effet, si $A(X)$ divise $B(X)$ à droite, alors toute racine de $A(X)$ est une racine de $B(X)$.

Le fait que $\mathbb{K}\langle X^q\rangle$ soit euclidien à droite implique l'existence d'un plus grand diviseur commun à droite et d'un plus petit multiple commun à gauche. L'objectif du prochain paragraphe est justement de donner un algorithme pour calculer ces deux objets.

Algorithme d'Euclide étendu, plus grand diviseur commun à droite et plus petit multiple commun à gauche

L'algorithme d'Euclide étendu à droite pour les q-polynômes est une généralisation directe de l'algorithme d'Euclide étendu classique pour les polynômes mais en utilisant la division euclidienne à droite pour les q-polynômes. Cet algorithme calcule des « relations de Bézout ». La dernière itération de cet algorithme (qui correspond au reste nul) permet de calculer le plus petit multiple commun à gauche et l'avant-dernière itération (et précisément le dernier reste non nul) permet de calculer le plus grand diviseur commun à droite. Je décris dans un premier temps l'algorithme d'Euclide étendu classique pour le calcul du plus grand diviseur commun à droite, je montre ensuite comment le modifier pour calculer le plus grand multiple commun à gauche.

Algorithme [Euclide étendu] 2.2

Entrées : $P_1, P_2 \in \mathbb{K}\langle X^q\rangle$
Sorties : U, V, R tels que $U \circ P_1 + V \circ P_2 = R = P_1 \wedge P_2$
$U_0 \leftarrow 1, V_0 \leftarrow 0, U_1 \leftarrow 0, V_1 \leftarrow 1, R_0 \leftarrow A, R \leftarrow B, Q \leftarrow 0, U_t \leftarrow 0, V_t \leftarrow 0$
Tant que $R \neq 0$ **faire**
    $R \leftarrow R_0$
    $(Q, R) \leftarrow \text{Euclide}(R_0, R)$
    $U_t \leftarrow U_1, V_t \leftarrow V_1, U_1 \leftarrow U_0 - Q \circ U_1, V_1 \leftarrow V_0 - Q \circ V_1, U_0 \leftarrow U_t, V_0 \leftarrow V_t, R \leftarrow R_0$
fin du tant que
Renvoyer (U_0, V_0, R_0)

La preuve consiste à constater qu'à chaque itération, on a $U_0 \circ P_1 + V_0 \circ P_2 = R_0$ et $U_1 \circ P_1 + V_1 \circ P_2 = R$. Cette dernière relation de Bézout donne $U_1 \circ P_1 + V_1 \circ P_2 = 0$ à la fin de l'algorithme, si bien que $U \circ P_1$ est un multiple scalaire de $P_1 \vee_g P_2$ et il suffit de modifier la sortie de l'algorithme d'Euclide étendu 2.2 en retournant $U_1 \circ P_1$ pour avoir un multiple scalaire du plus petit multiple commun à gauche de P_1 et P_2. Voici donc un algorithme permettant de calculer une relation de Bézout et qui à terme permettra de calculer le plus petit multiple commun à gauche de deux q-polynômes.

Algorithme Bézout LLCM 2.3

Entrées : $P_1, P_2 \in \mathbb{K}\langle X^q\rangle$
Sorties : U, V tels que $U \circ P_1 + V \circ P_2 = 0$ et $q\text{-deg}(U) = q\text{-deg}(P_2) - q\text{-deg}(P_1 \wedge_d P_2)$ et $q\text{-deg}(V) = q\text{-deg}(P_2) - q\text{-deg}(P_1 \wedge_d P_2)$.
$U_0 \leftarrow 1, V_0 \leftarrow 0, U_1 \leftarrow 0, V_1 \leftarrow 1, R_0 \leftarrow A, R \leftarrow B, Q \leftarrow 0, U_t \leftarrow 0, V_t \leftarrow 0$
Tant que $R \neq 0$ **faire**
    $R \leftarrow R_0$
    $(Q, R) \leftarrow \text{Euclide}(R_0, R)$
    $U_t \leftarrow U_1, V_t \leftarrow V_1, U_1 \leftarrow U_0 - Q \circ U_1, V_1 \leftarrow V_0 - Q \circ V_1, U_0 \leftarrow U_t, V_0 \leftarrow V_t, R \leftarrow R_0$
fin du tant que
Renvoyer (U_1, V_1)

En effet, à la fin de l'algorithme, comme $U \circ P_1 + V \circ P_2 = 0$ alors $U \circ P_1$ est divisible à droite par P_2 et par suite $U \circ P_1$ est un multiple à gauche commun à P_1 et P_2. Pour des raisons de q-degré, il s'agit du plus petit multiple à gauche.

L'algorithme d'Euclide étendu 2.2 permet de calculer le plus grand commun diviseur à droite :

Algorithme RGCD 2.4

Entrées : $P_1, P_2 \in \mathbb{K}\langle X^q\rangle$
Sortie : $R = P_1 \wedge_d P_1$
$(U, V, R) \leftarrow \text{Euclide étandu}(P_1, P_2)$
Renvoyer R

L'algorithme 2.3 qui est juste une variante de 2.2 permet de calculer le plus petit commun multiple à gauche.

Algorithme LLCM 2.5

Entrées : $P_1, P_2 \in \mathbb{K}\langle X^q\rangle$
Sortie : $R = P_1 \vee_g P_1$
$(U, P) \leftarrow$ Bézout LLCM$(P_1, P2)$
Renvoyer$(U \circ P_1)$

Lemme 2.13. *Si $P(X) \in \mathbb{F}_{q^m}\langle X^q\rangle$ alors l'application $\zeta \in \mathbb{F}_{q^m} \mapsto P(\zeta) \in \mathbb{F}_{q^m}$ est $\mathbb{F}_q$-linéaire.*

Démonstration. C'est une simple conséquence du fait que X^q est identifié au Frobenius par évaluation. □

Corollaire 2.14. *Soit $P(X) \in \mathbb{F}_{q^m}\langle X^q\rangle$, l'ensemble $\mathcal{Z}(P) = \{\zeta \in \mathbb{F}_{q^m} | P(\zeta) = 0\}$ est un sous-espace vectoriel de $\mathbb{F}_{q^m}$ qu'on appellera noyau de $P(X)$.*

Lemme 2.15. *Soit $P(X) \in \mathbb{F}_{q^m}\langle X^q\rangle$, alors $P(\zeta) = 0$ si et seulement si $X^q - \zeta^{q-1} \cdot X$ divise $P(X)$ à droite.*

Démonstration. Si $X^q - \zeta^{q-1} \cdot X$ divise $P(X)$ à droite alors on a $P(X) = Q(X) \circ (X^q - \zeta^{q-1} \cdot X)$ et donc $P(\zeta) = Q(0) = 0$.

Réciproquement, en divisant $P(X)$ par $X^q - \zeta^{q-1} \cdot X$ à droite on a $P(X) = Q(X) \circ (X^q - \zeta^{q-1} \cdot X) + R(X)$ et donc si $P(\zeta) = 0$ alors $R(\zeta) = 0$ et comme $R(X)$ est de q-degré 0, $R(X) = 0$ et $X^q - \zeta^{q-1} \cdot X$ divise $P(X)$ à droite. □

Lemme 2.16. *Soient $P_1(X)$ et $P_2(X) \in \mathbb{F}_{q^m}\langle X^q\rangle$ des polynômes linéarisés de q-degré inférieur ou égal à d, alors si $\dim_{\mathbb{F}_q}(\mathcal{Z}(P_1(X) - P_2(X)) \geqslant d$ on a $P_1(X) = P_2(X)$.*

Démonstration. On note $P(X) = P_1(X) - P_2(X)$ et on considère $\zeta_1, \cdots, \zeta_l$ $(l > d)$ une base de $\mathcal{Z}(P)$, on sait que comme ζ_i est une racine de $P(X)$, alors $X^q - \zeta_i^{q-1} \cdot X$ divise $P(X)$ à droite pour tout $i \in \{1, \cdots, l\}$ et on note $Q_i(X)$ le quotient de $P(X)$ par $X^q - \zeta_i^{q-1} \cdot X$. Puisque $P(X)$ est divisible par chacun des $X^q - \zeta_i^{q-1} \cdot X$, il est divisible par le $H(X) = \text{LLCM}(X^q - \zeta_1^{q-1} \cdot X, \cdots, X^q - \zeta_l^{q-1} \cdot X)$. Or le q-degré de $H(X)$ est plus grand que celui de $P(X)$. On en conclut que $P(X) = 0$. □

Corollaire 2.17. *Soient $P(X) \in \mathbb{F}_{q^m}\langle X^q\rangle$ et $\zeta \in \mathbb{F}_{q^m}$ alors le reste de la division euclidienne de $P(X)$ à droite par $X^q - \zeta \cdot X$ est $P(\zeta) \cdot X$.*

Le polynôme plus petit multiple à gauche des $X^q - \zeta_i \cdot X$ joue un rôle particulier, c'est un polynôme d'interpolation :

Lemme 2.18. *Soit E un sous-espace vectoriel de $\mathbb{F}_{q^m}$ et soit $\zeta_1, \cdots, \zeta_d$ une base de E alors il existe un unique polynôme linéarisé unitaire $P_E(X) = \bigvee_{i=1}^{d}{}_g (X^q - \zeta_i^{q-1} \cdot X)$ de q-degré d exactement tel que $E = \mathcal{Z}(P_E)$.*

De façon analogue à la géométrie algébrique classique on a deux correspondances :

- La première allant de l'ensemble des sous-espaces vectoriels (Grassmannienne) de $\mathbb{F}_{q^m}$ dans les polynômes linéarisés : $E \longmapsto P_E$.
- La seconde allant des polynômes linéarisés dans l'ensemble des sous-espaces vectoriels de $\mathbb{F}_{q^m}$: $P \longmapsto \mathcal{Z}(P)$.

En fait, la correspondance se prolonge en replaçant les polynômes linéarisés par les idéaux à gauche des polynômes linéarisés.

2.1.4. Résultants des polynômes linéarisés

Cette sous-section est consacrée aux résultants des polynômes linéarisés. Bien sûr, beaucoup des constructions données ici se généralisent à des classes beaucoup plus larges de polynômes de Öre. Néanmoins, comme je m'intéresserai à des aspects géométriques sur les corps finis, je donnerai aussi des résultats spécifiques aux polynômes linéarisés et j'ai donc spécialisé tous les résultats dans ce contexte pour simplifier la présentation. Une partie des résultats présentés ici l'a été également dans la thèse de Gaëtan Murat [40].

Deux constructions du résultant

La construction de la matrice de Sylvester de deux polynômes linéarisés $A(X)=\sum_{i=0}^{d} a_i \cdot X^{q^i}$ et $B(X)=\sum_{i=0}^{e} b_i \cdot X^{q^i} \in \mathbb{F}_{q^m}\langle X^q\rangle$ de q-degré respectivement d et e se généralise directement en considérant la matrice de l'application suivante :

$$\mathcal{S}_{e,d}: \begin{cases} \mathbb{F}_{q^m}\langle X^q\rangle_e \times \mathbb{F}_{q^m}\langle X^q\rangle_d & \longrightarrow \mathbb{F}_{q^m}\langle X^q\rangle_{d+e-1} \\ (U(X),V(X)) & \longmapsto U(X)\circ A(X)+V(X)\circ B(X) \end{cases}.$$

On écrit alors la matrice de cette application linéaire dans les bases $\{(X,0),(X^q,0),\cdots,(X^{q^{e-1}},0),(0,X),(0,X^q),\cdots,(0,X^{q^{d-1}})\}$ comme base de la source et $\{X,X^q,\cdots,X^{d+d-1}\}$ comme base de la cible. Ainsi, si on note $\mathrm{Syl}_{e,d}=(s_{i,j})$ on a alors pour $0\leqslant i<e$, $s_{i,j}$ est le coefficient du terme de q-degré j dans $X^{q^i}\cdot A(X)=\sum_{k=0}^{d} a_k^{q^i}\cdot X^{q^{i+k}}$ ce qui vaut 0 si $j<i$, $a_{j-i}^{q^i}$ pour $i\leqslant j\leqslant i+e-1$ et 0 pour $j>d$ et si $d\leqslant i\leqslant d+e-1$ alors $s_{i,j}$ est le coefficient du terme de q-degré j dans $X^{q^i}\cdot B(X)$ ce qui vaut 0 si $j<i-e$, $b_{j-i+e}^{q^{i-e}}$ pour $i-e\leqslant j\leqslant i-e+d-1$ et 0 sinon, ce qui donne une matrice du type suivant si $d=e$:

$$\mathrm{Syl}_{d,d}=\begin{pmatrix} a_0 & & & b_0 & & \\ \vdots & \ddots & & \vdots & \ddots & \\ a_d & & a_0^{q^{d-1}} & b_d & & b_0^{q^{d-1}} \\ & \ddots & \vdots & & \ddots & \vdots \\ & & a_d^{q^{d-1}} & & & b_d^{q^{d-1}} \end{pmatrix}.$$

Cette application est injective si et seulement si $A(X)\wedge_d B(X)=X$. Ainsi, $\mathrm{Res}(A(X),B(X))=\det(\mathrm{Syl}_{e,d})$ est tel que $\mathrm{Res}(A(X),B(X))=0$ si et seulement si $A(X)\wedge_d B(X)=X$. Cette formulation est la généralisation de la matrice de Sylvester classique. Mais les propriétés du résultant ne sont pas faciles à voir avec cette formulation. J'introduis donc la formulation en termes de matrices de multiplication (matrices compagnons généralisées) qui permettra de généraliser beaucoup de résultats bien connus sur les résultants commutatifs de façon très naturelle.

La construction en termes de matrices de multiplication ou de matrices compagnons généralisées du résultant est à rapprocher du travail entrepris avec Paola Boito dans le cadre du calcul symbolique-numérique. L'idée n'est pas nouvelle, elle permet de démontrer des propriétés du résultant qui sont bien connues dans le cadre commutatif dans le cadre non-commutatif mais de façon plus simple qu'avec la formulation en termes de matrice de Sylvester. Par exemple, les formules de sous-résultants s'expriment bien pour les polynômes de Öre (voir [40]) et permettent de retrouver les formules données initialement par Zimming Li [35].

Si on considère $A(X) \in \mathbb{F}_{q^m}\langle X^q\rangle$, on peut alors définir la relation d'équivalence $U(X) \cong V(X)$ si et seulement ils ont le même reste par la division à droite par $A(X)$. L'ensemble des classes d'équivalence est appelé quotient à droite par l'idéal engendré par $P(X)$ et il est noté $\mathbb{F}_{q^m}\langle X^q\rangle/(A(X))$. On définit Π_A: $\left\{ \begin{smallmatrix} \mathbb{F}_{q^m}\langle X^q\rangle \longrightarrow \mathbb{F}_{q^m}\langle X^q\rangle/(A(X)) \\ U(X) \longmapsto \Pi_A(U)(X) \end{smallmatrix} \right.$ où $\Pi_A(U)(X)$ est le reste de la division euclidienne à droite de $U(X)$ par $A(X)$. On vérifie facilement que cette application est un morphisme d'anneaux.

Si $B(X) \in \mathbb{F}_{q^m}\langle X^q\rangle$, on définit la multiplication à droite par $B(X)$ dans $\mathbb{F}_{q^m}\langle X^q\rangle/(A(X))$ de la façon suivante :

$$\mathcal{M}_B: \left\{ \begin{array}{ccc} \mathbb{F}_{q^m}\langle X^q\rangle/(A(X)) & \longrightarrow & \mathbb{F}_{q^m}\langle X^q\rangle/(A(X)) \\ U(X) & \longmapsto & \Pi_A(U(X)\circ B(X)) \end{array} \right. .$$

Remarquons que, si le q-degré de $B(X)$ est plus grand que celui de $A(X)$, alors multiplier par $B(X)$ revient à multiplier par $\Pi_A(B)(X)$ car Π_A est une projection. Un des intérêts de l'introduction de cette construction est d'avoir une nouvelle construction du résultant.

Proposition 2.19. *Soient $A(X)$ et $B(X) \in \mathbb{F}_{q^m}\langle X^q\rangle$, on note $\mathcal{M}_B$ la matrice de multiplication par $B(X)$ à droite dans $\mathbb{F}_{q^m}\langle X^q\rangle/(A(X))$, alors $\det(\mathcal{M}_B) = 0$ si et seulement si $A(X) \wedge_d B(X) \neq X$, en d'autres termes, il existe $\alpha \in \mathbb{F}_{q^m}^{\times}$ tel que $\det(\mathcal{M}_B) = \alpha \cdot \mathrm{Res}(A(X), B(X))$.*

Je renvoie à la thèse de Gaëtan Murat [40] pour les formules de sous-résultants (qui sont données dans un cas plus général des polynômes de Öre). Il est facile de dériver un algorithme de calcul de $A(X) \wedge_d B(X)$ et $A(X) \vee_g B(X)$ à partir de la construction de la matrice $\mathcal{M}_B$ et de réinvestir les idées de [8] pour obtenir un algorithme de bonne complexité. Je donne maintenant l'algorithme pour le calcul du PGCD à droite et je donnerai l'évaluation de complexité de cet algorithme.

L'algorithme est composé par deux étapes principales : le calcul de $\mathcal{M}_B$ et la mise sous forme échelonnée de cette matrice. Si pour la mise sous forme échelonnée nous ne proposons pas d'algorithmes particuliers, pour la construction de la matrice de multiplication, il est facile d'améliorer les algorithmes « génériques ». En effet, il est naturel de chercher comme dans [8] à tirer parti de la structure de la matrice. L'idée est d'écrire la multiplication par $B(X)$ dans la $\mathbb{F}_{q^m}$-base monomiale $X, X^q, \cdots, X^{q^{d-1}}$ de $\mathbb{F}_{q^m}\langle X^q\rangle/(A(X))$. La colonne d'indice $i \in \{0, \cdots, d-1\}$ de la matrice M_B de $\mathcal{M}_B$ dans cette base est le vecteur formé des coefficients du reste de $X^{q^i} \circ B(X)$ par la division euclidienne par $A(X)$: $\Pi_A(X^{q^i} \circ B(X))$. Il est intéressant de calculer ces restes récursivement. En effet, si $C(X)$ est un polynôme linéarisé de q-degré d, alors $\Pi_A(C)(X) = C(X) - \frac{c_d}{a_d} \cdot A(X)$, ce qui demande $\mathcal{O}(d)$ opérations dans $\mathbb{F}_{q^m}$. On va se servir de cela en tirant profit de la relation $\Pi_A(X^{q^{i+1}} \circ B(X))(X) = \Pi_A(X^q \circ \Pi_A(X^{q^i} \circ B(X)))(X)$. Il faut donc multiplier un élément de q-degré $d-1$ par X^q, ce qui revient essentiellement à appliquer le Frobenius à tous les coefficients, et à « shifter » en q-degré, ce qui nécessite $\mathcal{O}(d)$ opérations arithmétiques dans $\mathbb{F}_{q^m}$, ce qui permet de construire la matrice de multiplication en $\mathcal{O}(d^2)$ opérations arithmétiques dans $\mathbb{F}_{q^m}$. Le coût dominant dans l'algorithme de calcul du PGCD à droite par la méthode de la matrice de multiplication est donc le calcul de la forme échelonnée de la matrice M_B.

On a des résultats analogues pour le calcul des sous-résultants avec cette même méthode [40]. Certains résultats deviennent beaucoup plus faciles à démontrer avec cette formulation du résultant. Par exemple, comme $\Pi_A(U \circ V)(X) = \Pi_1(\Pi_A(U)(X) \circ \Pi_A(V)(X))(X)$, on retrouve facilement la multiplicativité du résultant grâce à celle du déterminant.

Proposition 2.20. *Soit $A(X), B(X)$ et $C(X) \in \mathbb{F}_{q^m}\langle X^q\rangle$ on a alors $\mathrm{Res}(A(X), B(X) \circ C(X)) = \mathrm{Res}(A(X), B(X)) \cdot \mathrm{Res}(A(X), C(X))$.*

Matrices compagnons généralisées et interpolation

Depuis l'introduction des polynômes linéarisés par Öre [41], l'interpolation et donc l'interprétation géométrique de ces polynômes jouent un rôle important dans la théorie car en dehors de ce cas et de celui des opérateurs différentiels, il est difficile d'avoir une interprétation géométrique des polynômes de Öre.

Déjà dans [41], l'auteur introduit la matrice q-Vandermonde. Les $\zeta_1, \cdots, \zeta_d \in \mathbb{F}_{q^m}$ alors le déterminant de q-Vandermonde associé est :

$$V_{q,\boldsymbol{\zeta}} = \begin{vmatrix} \zeta_1 & \cdots & \zeta_d \\ \zeta_1^q & \cdots & \zeta_d^q \\ \vdots & & \vdots \\ \zeta_1^{q^{d-1}} & \cdots & \zeta_d^{q^{d-1}} \end{vmatrix}.$$

PROPOSITION 2.21. *Les valeurs $\zeta_1, \cdots, \zeta_d$ sont $\mathbb{F}_q$-libres si et seulement si $V_{q,\boldsymbol{\zeta}} \neq 0$.*

Je fais maintenant l'hypothèse que les $\zeta_1, \cdots, \zeta_d$ sont $\mathbb{F}_q$-libres. Nous allons généraliser quelques formules de l'interpolation de Lagrange dans le cadre des polynômes linéarisés. Pour $i \in \{1, \cdots, d\}$, si je remplace la colonne d'indice i par la colonne $\begin{pmatrix} X \\ X^q \\ \vdots \\ X^{q^{d-1}} \end{pmatrix}$, j'obtiens un polynôme linéarisé $V_i(X)$ tel que $V_i(\zeta_j) = 0$ pour tout $j \in \{1, \cdots, i-1, i+1, \cdots, d\}$ et $V_i(z_i) = (-1)^i \cdot V_{q,\boldsymbol{\zeta}}$. Je définis alors le polynôme linéarisé $L_{i,\boldsymbol{\zeta}}(X) = (-1)^i \cdot V_i(X)/V_{q,\boldsymbol{\zeta}}$ qui est tel que :

$$L_{i,\boldsymbol{\zeta}}(\zeta_j) = \begin{cases} 0 \text{ si } j \in \{1, \cdots, d\} \setminus \{i\} \\ 1 \text{ si } i = j \end{cases}.$$

On peut construire ces polynômes de façon différente : on note $U_{i,\boldsymbol{\zeta}}(X) = \underset{j \neq i}{\vee_g} (X^q - \zeta_j \cdot X)$, on a alors $U_{i,\boldsymbol{\zeta}}(\zeta_j) = 0$ si $j \neq i$ car $(X^q - \zeta_j \cdot X)$ divise $U_{i,\boldsymbol{\zeta}}(X)$ à droite et $U_{i,\boldsymbol{\zeta}}(\zeta_i) \neq 0$ car $(X^q - \zeta_i \cdot X)$ ne divise pas $U_{i,\zeta}(X)$. Ainsi, on a $L_{i,\boldsymbol{\zeta}}(X) = U_{i,\boldsymbol{\zeta}}(X)/U_{i,\zeta}(\zeta_i)$.

Je considère maintenant $Z(X) = \underset{i=1}{\overset{d}{\vee_g}} (X^q - \zeta_i \cdot X)$ et je note $\mathcal{A}_{\boldsymbol{\zeta}} = \mathbb{F}_{q^m}\langle X^q \rangle / (Z(X))$ et comme auparavant Π_Z l'application consistant à calculer le reste d'un polynôme linéarisé par la division à droite par $Z(X)$. C'est en quelque sorte une généralisation de l'anneau des coordonnées.

Remarquons que les polynômes linéarisés $L_{1,\boldsymbol{\zeta}}(X), \cdots, L_{d,\boldsymbol{\zeta}}(X)$ sont $\mathbb{F}_{q^m}$-libres. En effet, s'il étaient liés, il existerait $(\lambda_1, \cdots, \lambda_d) \in (\mathbb{F}_{q^m})^d$ tels que $\lambda_1 \cdot L_{1,\boldsymbol{\zeta}}(X) + \cdots + \lambda_d \cdot L_{d,\boldsymbol{\zeta}}(X) = 0$ et dans ce cas en évaluant en ζ_i on aurait $\lambda_i \cdot L_{i,\boldsymbol{\zeta}}(\zeta_i) = \lambda_i = 0$ pour tout $i \in \{1, \cdots, d\}$ et par conséquent la seule combinaison linéaire entre ces éléments est la triviale. Ceci permet de montrer la proposition suivante :

PROPOSITION 2.22. *La famille $\{L_{1,\boldsymbol{\zeta}}(X), \cdots, L_{d,\boldsymbol{\zeta}}(X)\}$ est une base de $\mathcal{A}_{\boldsymbol{\zeta}}$ comme $\mathbb{F}_{q^m}$-espace vectoriel.*

Cette proposition est une incarnation de la formule d'interpolation de Lagrange pour les polynômes linéarisés :

THÉORÈME 2.23. *Soit $\boldsymbol{\zeta} = \{\zeta_1, \cdots, \zeta_d\} \subset \mathbb{F}_{q^m}$ et $(v_1, \cdots, v_d) \in (\mathbb{F}_{q^m})^d$ alors il existe un unique polynôme linéarisé d'interpolation $P(X) \in \mathbb{F}_{q^m}\langle X^q \rangle$ de q-degré inférieur ou égal à $d-1$ tel que $P(\zeta_i) = v_i$ et il est donné par :*

$$P(X) = \sum_{i=1}^{d} v_i \cdot L_{i,\boldsymbol{\zeta}}(X).$$

Comme $\mathcal{A}_{\boldsymbol{\zeta}}$ est un $\mathbb{F}_{q^m}$-espace vectoriel de dimension d, on peut donner une base de son espace dual en considérant les $\mathbb{1}_{\zeta_i} : \begin{cases} \mathcal{A}_{\boldsymbol{\zeta}} \to \mathbb{F}_{q^m} \\ P(X) \mapsto P(\zeta_i) \end{cases}$ qui sont des formes linéaires linéairement indépendantes.

On s'intéresse à l'application linéaire $\mathcal{M}_{X^q}$ de multiplication par X^q (remarquez que la multiplication par X est simplement l'identité). On se demande alors quels sont les valeurs et vecteurs propres. La première remarque est la suivante :

Lemme 2.24. *La valeur ζ_i^q est une valeur propre associée au vecteur propre* $\begin{pmatrix} \zeta_i \\ \zeta_i^q \\ \vdots \\ \zeta_i^{q^{d-1}} \end{pmatrix}$ *de $\mathcal{M}_{X^q}$.*

C'est une conséquence de la formule d'interpolation de Lagrange. En effet, par dualité, la matrice de $\mathcal{M}_{X^q}$ dans la base de Lagrange est :

$$\begin{pmatrix} \zeta_1^q & & \\ & \ddots & \\ & & \zeta_d^q \end{pmatrix}.$$

Un corollaire direct du lemme 2.24 est :

Corollaire 2.25. *Soit $\boldsymbol{\zeta} = \{\zeta_1, \cdots, \zeta_d\} \subset \mathbb{F}_{q^m}$ et soit $Z(X) = \bigvee_{g,i=1}^{d} (X^q - \zeta_i \cdot X)$ alors pour tout $B(X) \in \mathcal{A}_Z$, la matrice de $\mathcal{M}_B$ dans la base de Lagrange est :*

$$\begin{pmatrix} B(\zeta_1) & & \\ & \ddots & \\ & & B(\zeta_d) \end{pmatrix}.$$

Lemme 2.26. *Soit $\boldsymbol{\zeta} = \{\zeta_1, \cdots, \zeta_d\} \subset \mathbb{F}_{q^m}$, $Z(X) = \bigvee_{g,i=1}^{d} (X^q - \zeta_i \cdot X)$ et soit $B(X) \in \mathbb{F}_{q^m}\langle X^q \rangle$ alors le reste de la division de $B(X)$ par $Z(X)$ à droite est donné par $\sum_{i=1}^{d} B(\zeta_i) \cdot L_{i,\boldsymbol{\zeta}}(X)$.*

Dans le cadre commutatif, une formule classique de résultant, reliée au résidu de Grothendieck, est donnée par :

Proposition 2.27. *Soit $A(X)$ et $B(X) \in \mathbb{A}[X]$ où $\mathbb{A}$ est un anneau intègre et soit $\mathbb{K}$ la clôture algébrique du corps des fractions de $\mathbb{A}$, on note $\zeta_1, \cdots, \zeta_d \in \mathbb{K}$ les racines de A et $\xi_1, \cdots, \xi_e \in \mathbb{K}$ les racines de B, alors $\mathrm{Res}_X(A, B) = \prod_{i=1}^{d} B(\zeta_i) = \prod_{j=1}^{e} A(\xi_j)$.*

Grâce au lemme 2.24 on a une proposition analogue pour les polynômes linéarisés :

Proposition 2.28. *Soit $\boldsymbol{\zeta} = \{\zeta_1, \cdots, \zeta_d\} \subset \mathbb{F}_{q^m}$, $Z(X) = \bigvee_{g,i=1}^{d} (X^q - \zeta_i \cdot X)$ et soit $B(X) \in \mathbb{F}_{q^m}\langle X^q \rangle$ alors $\mathrm{Res}(Z(X), B(X)) = \prod_{i=0}^{d} B(\zeta_i)$.*

Ces résultats peuvent s'étendre de la façon suivante :

Proposition 2.29. *Soit $A(X)$ et $B(X) \in \mathbb{F}_{q^m}\langle X \rangle$ sans facteur carré de q-degrés respectivement d et e et soit $\mathbb{K} = \mathbb{F}_{q^l}$ le corps de décomposition de $A(X) \vee_g B(X)$ et soit $\{\zeta_1, \cdots, \zeta_d\} = \mathcal{Z}(A(X))$ et $\{\xi_1, \cdots, \xi_e\} = \mathcal{Z}(B(X))$ alors $\mathrm{Res}(A(X), B(X)) = \prod_{i=1}^{d} B(\zeta_i) = \prod_{j=1}^{e} A(\xi_j) \in \mathbb{F}_{q^m}$.*

Ces formules sont des incarnations des formules dites de Poisson dans le cas classique (voir [1] pour un exposé systématique du cas commutatif) mais pour les polynômes linéarisés. Beaucoup des formules classiques de discriminant sont aussi généralisables grâce aux formalismes développés ici et devraient faire l'objet d'un travail systématique dans les mois à venir.

2.1.5. Liens entre polynômes linéarisés et endomorphismes linéaires des corps finis

On note $\mathrm{End}_{\mathbb{F}_q}(\mathbb{F}_{q^m})$ l'ensemble des applications linéaires de $\mathbb{F}_{q^m}$ dans $\mathbb{F}_{q^m}$ qui sont $\mathbb{F}_q$-linéaires. Le $\mathbb{F}_q$ espace vectoriel $\mathrm{End}_{\mathbb{F}_q}(\mathbb{F}_{q^m})$ peut être muni d'une structure de $\mathbb{F}_q$-algèbre en la munissant de la composition comme produit. Un résultat classique est le lemme suivant :

LEMME 2.30. *Les $\mathbb{F}_q$-algèbres $\mathrm{End}_{\mathbb{F}_q}(\mathbb{F}_{q^m})$ et $\mathbb{F}_{q^m}\langle X^q\rangle/(X^{q^m}-X)$ sont isomorphes.*

Autrement dit, toute application linéaire de $\mathbb{F}_{q^m}$ dans lui-même peut être représentée par une polynôme linéarisé. Le lien entre ces polynômes et la géométrie dans $\mathbb{F}_{q^m}$ est l'objet de la première partie de la thèse de Gaëtan Murat. Le point de vue est proche de celui de la géométrie algébrique et nous avons commencé à généraliser des formules classiques d'élimination au cas des polynômes linéarisés, c'est-à-dire dans un contexte non-commutatif, mais « faiblement » non-commutatif. La nature euclidienne de l'anneau $\mathbb{F}_{q^m}\langle X^q\rangle$ et la relation avec la géométrie permettent d'obtenir des résultats très analogues à ce qu'on connaît pour l'élimination classique.

Je rappellerai rapidement ici les principaux résultats de la thèse de Gaëtan Murat sur le sujet puis je montrerai quelques résultats nouveaux et en quoi ils sont reliés à la géométrie sur $\mathbb{F}_{q^m}$.

PROPOSITION 2.31. *Soient E et F deux sous-espaces vectoriels de $\mathbb{F}_{q^m}$ alors $P_{E\cap F}(X)=P_E\wedge_r P_F(X)$ et $P_{E+F}(X)=P_E\vee_g P_F(X)$.*

Cette correspondance entre opérations sur les sous-espaces vectoriels et polynômes linéarisés passe aussi aux idéaux comme dans le cas commutatif.

2.1.6. Décodage des codes en métrique rang

Cette sous-section est consacrée au décodage des codes en métrique rang. Avec Philippe Gaborit et Julien Schrek, nous avons proposé dans [26] un algorithme générique pour le problème de décodage par syndrome en métrique rang (problème RSD). Ce problème est important car il conditionne l'usage des codes en métrique rang pour la cryptographie.

Les algorithmes traitant l'instance générique de ce problème sont exponentiels en une quantité qui est quadratique des paramètres du code qu'on cherche à décoder. Si je veux décoder un code de longueur n et de dimension k sur $\mathbb{F}_{q^m}$ en corrigeant une erreur de rang r, l'algorithme de Chabaud-Stern proposé en 1996 coûte essentiellement $\mathcal{O}((n\cdot r+m)^3\cdot q^{(m-r)\cdot(r-1)})$ [13] et l'algorithme de Ourivski et Johansson proposé en 2003 coûte $\mathcal{O}((k+r)^3\cdot r^3\cdot q^{(k+1)\cdot(r-1)})$ [43]. Des algorithmes plus spécifiques permettent de traiter des codes particuliers comme les codes de Gabidulin (codes qui sont l'objet de la sous-section suivante).

Pendant une dizaine d'années, l'état de l'art sur le problème RSD générique n'a pas évolué. Néanmoins, en 2006 Lévy-dit-Véhel et Perret (ref) proposaient une approche algébrique mais qui se limitait à $r=2$ ou 3 et en 2008, Faugère et Lévy-dit-Véhel [22] considéraient ce problème dans le cas $n=m$ comme un cas particulier du problème MaxRank et proposaient un algorithme basé comme le précédent sur un calcul de bases de Gröbner.

C'est dans ce cadre que dans [26], avec Gaborit et Schrek, nous avons étudié des algorithmes de décodage pour le problème RSD générique. Nous avons proposé deux approches. La première est essentiellement une généralisation des approches de Chabaud-Stern [13] et Ourivski-Johansson [43] en étendant une attaque sur le support du cas de la distance de Hamming à la distance rang. La seconde approche repose sur les polynômes linéarisés. Dans les deux cas, on se ramène à construire un système algébrique dont les solutions permettent de décoder. La complexité de nos approches repose donc sur la complexité du calcul d'une base de Gröbner en grande partie.

Idée centrale pour attaquer RSD

L'idée centrale est de décomposer le décodage par syndrome en 2 étapes : calculer une base du support de l'erreur puis calculer l'erreur (ce ne sera plus que de l'algèbre linéaire si on connaît le support).

Problème 2.6. On considère $\mathcal{C} \subset (\mathbb{F}_{q^m})^n$ un code linéaire en métrique rang de dimension k et soit $e \in \mathbb{F}_{q^m}$ un mot de poids rang w, on note $H \in (\mathbb{F}_{q^m})^{(n-k)\times n}$ et on se donne $s^T = H \cdot e^T$ le syndrome associé à e. Le problème consiste à retrouver e connaissant s.

Traitons d'abord le cas $n \geqslant m$. On note $e = (e_1, \cdots, e_n)$ l'erreur qui vit dans $(\mathbb{F}_{q^m})^n$, on note $E = \langle e_1, \cdots, e_n \rangle \in \mathbb{F}_{q^m}$ le $\mathbb{F}_q$-sous-espace vectoriel de $\mathbb{F}_{q^m}$ engendré par les coordonnées de l'erreur et on note $\{F_1, \cdots, F_r\}$ une $\mathbb{F}_q$-base d'un sous-espace vectoriel F de $\mathbb{F}_{q^m}$ avec $r > w$. On suppose que $E \subset F$. Chaque $e_i, i \in \{1, \cdots, n\}$, peut s'écrire $e_i = \sum_{j=1}^{r} e_{i,j} \cdot F_j$. Si on note $H = (\ h_{i,j}\ )_{\substack{i \in \{1,\cdots,n-k\} \\ j \in \{1,\cdots,n\}}}$ alors en écrivant l'équation de syndrome on obtient :

$$\begin{cases} h_{1,1} \cdot e_1 + \cdots + h_{1,n} \cdot e_n = s_1 \\ \vdots \\ h_{n-k,1} \cdot e_1 + \cdots + h_{n-k,n} \cdot e_n = s_{n-k} \end{cases}$$

et en développant les e_i dans la base $\{F_1, \cdots, F_r\}$, on obtient :

$$\begin{cases} \sum_{l=1}^{r} (h_{1,1} \cdot e_{1,l} \cdot F_l + \cdots + h_{1,n} \cdot e_{n,l} \cdot F_l) = s_1 \\ \vdots \\ \sum_{l=1}^{r} (h_{n-k,1} \cdot e_{1,l} \cdot F_l + \cdots + h_{n-k,n} \cdot e_{n,l} \cdot F_l) = s_{n-k} \end{cases} . \tag{2.3}$$

On peut tout écrire sur $\mathbb{F}_q$ en prenant $\{\beta_1, \cdots, \beta_m\}$ une $\mathbb{F}_q$-base de $\mathbb{F}_{q^m}$ et en notant $\varphi_i \colon \mathbb{F}_{q^m} \to \mathbb{F}_q$ l'application qui a $x = x_1 \cdot \beta_1 + \cdots + x_m \cdot \beta_m \mapsto x_i$ on obtient :

$$\begin{cases} \varphi_u\Big(\sum_{l=1}^{r} (h_{v,1} \cdot e_{1,l} \cdot F_l + \cdots + h_{v,n} \cdot e_{n,l} \cdot F_l) \Big) = \varphi_u(s_i) \\ u \in \{1, \cdots, m\} \text{ et } v \in \{1, \cdots, n-k\} \end{cases} \tag{2.4}$$

ce qui par linéarité des φ_i donne :

$$\begin{cases} \sum_{l=1}^{r} e_{1,l} \cdot \varphi_u(h_{v,1} \cdot F_l) + \cdots + e_{n,l} \cdot \varphi_u(h_{v,n} \cdot F_l) = \varphi_u(s_i) \\ u \in \{1, \cdots, m\} \text{ et } v \in \{1, \cdots, n-k\} \end{cases} . \tag{2.5}$$

Si on suppose connue la base $\{F_1, \cdots, F_r\}$, on a $n \cdot r$ inconnues (les $e_{i,l}$) et $(n-k) \cdot m$ équations. Autrement dit, le système a plus d'équations que d'inconnues dès que $n \cdot r \leqslant m \cdot (n-k)$ donc dès que $r \leqslant m - \left\lfloor \frac{m \cdot k}{n} \right\rfloor$.

Si on tire F au hasard, la probabilité que $E \subset F$ est $\frac{\left[\begin{smallmatrix} r \\ w \end{smallmatrix} \right]_q}{\left[\begin{smallmatrix} m \\ w \end{smallmatrix} \right]_q} = \Theta(q^{-w(m-r)})$ (le nombre de sous-espaces de dimension w dans un espace de dimension r sur le nombre d'espaces de dimension w dans un espace de dimension m). Donc par énumération des espaces de dimension r, on obtient une complexité en $\mathcal{O}\Big((m \cdot (n-k))^3 \cdot q^{w \cdot \left\lceil \frac{k \cdot m}{n} \right\rceil} \Big)$ en prenant $r = m - \left\lceil \frac{k \cdot m}{n} \right\rceil$ et avec de l'algèbre linéaire naïve.

Dans le cas $m > n$ (qui est une configuration rare en pratique), la complexité devient un $\mathcal{O}((m \cdot (n-k))^3 \cdot q^{w \cdot k})$ avec un raisonnement analogue.

Nous avons ensuite proposé différentes améliorations de cette approche.

Approche à base de polynômes annulateurs

On considère un code $\mathcal{C}$ de longueur n sur $\mathbb{F}_{q^m}$ et de dimension k ayant une matrice génératrice $G \in (\mathbb{F}_{q^m})^{k \times n}$. Supposons que l'on ait $y = c + e$ avec $c^T = x^T \cdot G \in \mathcal{C}$ et $\text{rang}_{\mathbb{F}_q}(e) = r$. Dans la modélisation algébrique classique (ref LdV-P), la mise en équation comporte r inconnues dans $\mathbb{F}_{q^m}$ pour une base de l'espace vectoriel engendré par les coefficients de e, k inconnues dans $\mathbb{F}_{q^m}$ pour les coefficients de x et $n \times r$ inconnues pour les coordonnées de e dans la base de l'erreur.

Nous présentons maintenant un modèle algébrique avec $n+k$ inconnues dans $\mathbb{F}_{q^m}$. Soit $E = \langle e_1, \cdots, e_n \rangle$ le $\mathbb{F}_q$-sous-espace vectoriel de $\mathbb{F}_{q^m}$ engendré par les coefficients de l'erreur. Soit $P_E(X)$ le polynôme linéarisé annulateur de E qui est un polynôme linéarisé unitaire de q-degré r d'après le lemme 2.18. On a donc $P_E(e_i) = P_E\left(y_i - \sum_{j=1}^{k} x_j \cdot g_{j,i} \right) = 0$ pour tout $i \in \{1, \cdots, n\}$ ce qui nous donne $k+r$ inconnues (les x_i et les coefficients de $P_E(X)$) et n équations. Les monômes intervenant dans ce système d'équations sont généralement de la forme $p_i \cdot x_j^{q^j}$ où les p_i sont les coefficients de $P_E(X)$. Si le système est bien linéaire en les p_i, il ne l'est malheureusement pas en les x_j (le degré est même exponentiel en r). Mais il est très creux.

Dans [26] nous avons alors étudié comment essayer de linéariser ce système quand c'est possible. On aboutit alors à la proposition suivante :

PROPOSITION 2.32. *Soit $\mathcal{C}$ un code aléatoire en métrique rang de longueur n et de dimension k sur $\mathbb{F}_{q^m}$, de matrice génératrice $G \in (\mathbb{F}_{q^m})^{k \times n}$ et supposons qu'on reçoive $y = c + e$ avec $c \in \mathcal{C}$ et e de rang r. S'il existe un entier $t \leqslant k$ tel que $n - t \geqslant (r+1) \cdot (k+1-t)$ alors la complexité du décodage est dans $\mathcal{O}(((n \cdot k \cdot t) + r^3 \cdot k^3) \cdot q^{r \cdot t})$ en termes d'opérations dans $\mathbb{F}_{q^m}$.*

En corollaire on a :

COROLLAIRE 2.33. *Soit $\mathcal{C}$ un code aléatoire en métrique rang de longueur n et de dimension k sur $\mathbb{F}_{q^m}$, et on suppose qu'on a $y = c + e$ avec $c \in \mathcal{C}$ et e de rang r. Alors si $\left\lceil \frac{(r+1)\cdot(k+1)-(n+1)}{r} \right\rceil \leqslant k$, on peut décoder en utilisant $\mathcal{O}\left(r^3 \cdot k^3 \cdot q^{r \cdot \left\lceil \frac{(r+1)\cdot(k+1)-(n+1)}{r} \right\rceil} \right)$ opérations dans $\mathbb{F}_{q^m}$.*

Pour aller un peu plus loin, on peut utiliser les bases de Gröbner ou faire de l'hybride entre des bases de Gröbner et de l'exhaustif sur certaines variables.

Digressions cryptographiques

Le problème de décodage « générique » est très important pour la cryptographie basée sur les codes en métrique rang. En effet, il s'agit d'une attaque sur les cryptosystèmes qui est souvent parmi les plus efficaces. Les liens avec le problème MinRank et les systèmes proposés pour la compétition post-quantique du NIST ont conduit à de nouveaux travaux comme [3] qui ont encore été améliorés récemment dans [4] qui est à ma connaissance et au moment où j'écris ces lignes la meilleure attaque connue (ce qui ne sera peut-être plus le cas quand le texte sera envoyé aux rapporteurs).

2.1.7. Les codes de Gabidulin

Les codes de Gabidulin sont des codes d'évaluation qui sont l'analogue pour les q-polynômes de la notion de codes de Reed-Solomon pour les polynômes commutatifs usuels. Je donne ici la description classique de ces codes. Nous verrons des constructions qui généralisent cette construction. On considère $\mathbb{F}_{q^m}\langle X^q \rangle_k$, l'ensemble des polynômes linéarisés de q-degré inférieur strict à k et $\zeta_1, \cdots, \zeta_n \in \mathbb{F}_{q^m}$, n éléments libres sur $\mathbb{F}_q$. On note $Z = \{\zeta_1, \cdots, \zeta_n\} \subset \mathbb{F}_{q^m}$ et on définit l'application $\mathbb{F}_q$-linéaire suivante :

$$\mathrm{ev}_Z : \begin{cases} \mathbb{F}_{q^m}\langle X^q \rangle_k \longrightarrow (\mathbb{F}_{q^m})^n \\ P(X) \longmapsto (P(\zeta_1), \cdots, P(\zeta_n)) \end{cases}.$$

Le code de Gabidulin associé $\mathcal{G}_{k,n}$ est défini comme $\mathrm{Im}(\mathrm{ev}_Z)$. Les mots du code sont identifiés à des matrices $m \times n$ en écrivant en colonnes les coordonnées du vecteur ligne représentant le mot du code dans une base de $\mathbb{F}_{q^m}$. Les codes de Gabidulin sont les codes les plus étudiés en métrique rang. Leur construction en tant que codes d'évaluation semble leur donner un rôle analogue aux codes de Reed-Solomon pour la métrique de Hamming. La variété des constructions algébriques (sur des corps) pour les codes correcteurs d'erreurs en métrique de Hamming ne trouve pour l'instant pas d'analogue en métrique rang.

Du point de vue transmission ces codes de Gabidulin sont assez remarquables : ils sont « Maximal Rank Separable » qui est l'équivalent de « Maximal Distance Separable » des codes de Hamming mais dans le cadre de la métrique rang.

2.1.8. Les codes LRPC

Les codes LRPC (low rank parity check) sont des codes rang pour lesquels on connaît une matrice de parité dont le support des coefficients est de faible rang. C'est une analogie avec codes LDPC (low density parity check) du cas de la métrique de Hamming. Ces codes ont été introduits dans [24].

Avant d'introduire les codes LRPC, nous introduisons un problème sur lequel repose le décodage de ces codes :

Produit de deux sous-espaces de $\mathbb{F}_{q^m}$

Pour décoder les codes LRPC, nous aurons besoin de « diviser » par un sous-espace vectoriel. Il convient donc d'abord de définir le produit de deux sous-espaces vectoriels.

Définition 2.34. *Soient A et B deux $\mathbb{F}_q$-sous-espaces vectoriels de $\mathbb{F}_q^m$ de dimension respectivement α et β et de base respectivement $\{A_1, \cdots, A_\alpha\}$ et $\{B_1, \cdots, B_\beta\}$, on note $A \cdot B = \langle \{a \cdot b \,|\, a \in A \text{ et } b \in B\} \rangle$ l'espace vectoriel engendré par le produit de tous les éléments de A avec les éléments de B.*

Nous verrons comment utiliser une instance du problème suivant pour permettre le décodage des codes LRPC :

Problème 2.7. Soient A et B deux $\mathbb{F}_q$-sous-espaces vectoriels de $\mathbb{F}_q^m$ et supposons B et $A \cdot B$ connus (on connaît une base ou une famille génératrice pour chacun des deux). Peut-on calculer une base de A ? Si oui, avec quelle complexité ?

Nous allons maintenant donner des conditions pour lesquelles on peut résoudre ce problème avec une probabilité contrôlée de succès. Pour évaluer des probabilités d'appartenir à certains sous-espaces, nous aurons besoin de savoir évaluer la dimension du produit de deux sous-espaces.

Lemme 2.35. *Soient A et B deux $\mathbb{F}_q$-sous-espaces vectoriels de $\mathbb{F}_q^m$ de dimension respectivement α et β et de base respectivement $\{A_1, \cdots, A_\alpha\}$ et $\{B_1, \cdots, B_\beta\}$ alors $\{A_i \cdot B_j \,|\, i \in \{1, \cdots, \alpha\}$ et $b \in \{1, \cdots, \beta\}\}$ est une famille génératrice de $A \cdot B$.*

En corollaire de ce lemme, on sait que la dimension de $A \cdot B$ est inférieure à $\alpha \cdot \beta$. Dans le cas où $\alpha \cdot \beta < m$ on se demande quelle est la dimension qu'on peut attendre pour $A \cdot B$.

Lemme 2.36. *Soient A' et B deux $\mathbb{F}_q$-sous-espaces vectoriels de $\mathbb{F}_q^m$ de dimension respectivement α' et β tels que $\dim_{\mathbb{F}_q}(A' \cdot B) = \alpha' \cdot \beta$. Soit $A = A' + \langle a \rangle$ avec a pris uniformément parmi les éléments de $\mathbb{F}_{q^m}$. Alors $\mathbb{P}(\dim_{\mathbb{F}_q}(A \cdot B) < \alpha' \cdot \beta + \beta) \leqslant \frac{q^{\alpha' \cdot \beta + \beta}}{q^m}$.*

Proposition 2.37. *Soit B un $\mathbb{F}_q$-sous-espace vectoriel fixé de $\mathbb{F}_q^m$ et soient $A_1, \cdots, A_\alpha \in \mathbb{F}_q^m$ des vecteurs choisis de façon aléatoirement indépendante engendrant un sous-espace noté A. Alors on a $\dim_{\mathbb{F}_q}(A \cdot B) = \alpha \cdot \beta$ avec probabilité $1 - \alpha \cdot \frac{q^{\alpha \cdot \beta}}{q^m}$.*

On suppose maintenant que B contient 1 et on note alors B^2 le sous-espace vectoriel engendré par les produits d'éléments de B. On note $\beta_2 = \dim_{\mathbb{F}_q}(B^2)$ et on considère A un sous-espace vectoriel de dimension α.

Lemme 2.38. *Supposons que $\dim_{\mathbb{F}_q}(A \cdot B^2) = \alpha \cdot \beta_2$ et soit $e \in A \cdot B \setminus A$ ($1 \in B$ donc $A \subset A \cdot B$), si $eB \subset A \cdot B$ alors il existe $x \in B$ tel que $xB \subset B$.*

PROPOSITION 2.39. *Supposons que m est premier. Soient A et B deux espaces choisis aléatoirement de dimension respectivement α et β, soit $\{b_1, \cdots, b_\beta\}$ une base de B et $S = A \cdot B$, alors la probabilité que $A = \bigcap_{i=1}^{\beta} b_i^{-1} \cdot S$ est minorée par $1 - \alpha \cdot \frac{q^{\alpha \cdot \beta \cdot (b+1)/2}}{q^m}$.*

PROPOSITION 2.40. *Soit B un $\mathbb{F}_q$-sous-espace vectoriel de dimension β contenant 1 et tel que $\dim_{\mathbb{F}_q}(B + B \cdot b^{-1}) = 2 \cdot \beta - 1$ pour un certain $b \in B$ et soit A un $\mathbb{F}_q$-sous-espace vectoriel aléatoire de dimension α, alors la probabilité que $(A \cdot B) \cap ((A \cdot B) \cdot b^{-1}) = A$ est minorée par $1 - \alpha \cdot \frac{q^{\alpha \cdot (2 \cdot \beta - 1)}}{q^m}$.*

Définition des codes LRPC

DÉFINITION 2.41. *Un code à matrice de parité de petit rang (LRPC pour « low rang party check »), de rang d, longueur n et dimension k sur $\mathbb{F}_{q^m}$ est un code dont une matrice $H = (h_{i,j})$ est une matrice $(n-k) \times n$ dont le $\mathbb{F}_q$-sous espace vectoriel $F = \langle h_{i,j} \rangle \subset \mathbb{F}_{q^m}$ engendré par ses coefficients est de dimension au plus d (d est supposé petit devant m). On notera $\{F_1, \ldots, F_d\}$ une base de F.*

En pratique, on utilise souvent des codes QC-LRPC pour quasi-cyclique LRPC :

DÉFINITION 2.42. *Un code quasi-cyclique à matrice de parité de petit rang (QC-LRPC) est un code quasi-cyclique dont une matrice de parité $H = (h_{i,j})$ est quasi-cyclique et dont la dimension du $\mathbb{F}_q$-sous-espace vectoriel $F = \langle h_{i,j} \rangle \subset \mathbb{F}_{q^m}$ engendré par ses coefficients est de dimension au plus d (d est supposé petit devant m).*

Grâce à cette propriété de petite dimension de F, nous allons pouvoir proposer un algorithme de décodage efficace.

Décodage des LRPC

On considère $\mathcal{C}$ un code rang LRPC de dimension k et de longueur n de matrice de parité $H = (h_{i,j})_{\substack{1 \leqslant i \leqslant k \\ 1 \leqslant j \leqslant n}}$ telle que $F = \langle h_{i,j} \,|\, 1 \leqslant i \leqslant k \text{ et } 1 \leqslant j \leqslant n \rangle_{\mathbb{F}_q}$ soit de dimension d et de base $\{F_1, \cdots, F_d\}$. Soient $c \in \mathcal{C}$ et $e \in (\mathbb{F}_{q^m})^n$ et soit $e = (e_1, \cdots, e_n) \in (\mathbb{F}_{q^m})^n$ tel que $E = \langle e_1, \cdots, e_n \rangle$ est un sous-espace de $\mathbb{F}_{q^m}$ de dimension r et de base $\{E_1, \cdots, E_r\}$. Pour tout $i \in \{1, \cdots, n\}$, on note $e_i = \sum_{j=0}^{r} e_{j,i} \cdot E_j$ et pour tout $u \in \{1, \cdots, k\}$ et $v \in \{1, \cdots, n\}$ on note $h_{u,v} = \sum_{j=1}^{d} h_{u,v,j} \cdot F_j$.

On suppose qu'on reçoit $y = c + e$ (donc avec $c = x \cdot G$) et on veut déterminer c ou e. Voici un algorithme pour les LRPC qui permet de décoder par syndrome :

Algorithme 2.6

1. **Calcul de l'espace des coefficients du syndrome**
   Calculer $H \cdot y^T = s = (s_1, \cdots, s_{n-k})$ et calculer une base de $S = \langle s_1, \cdots, s_{n-k} \rangle$
2. **Calcul du support de l'erreur**
   On définit $S_i = F_i^{-1} \cdot S$ le sous-espace dont les générateurs sont ceux de S multipliés par F_i^{-1}. Calculer $E = S_1 \cap S_2 \cap \cdots \cap S_d$ et calculer une base $\{E_1, \cdots, E_r\}$ de E.
3. **Calcul de l'erreur**
   On écrit $H \cdot e^T$ dans la base $\{E_1 \cdot F_1, \cdots, E_1 \cdot F_d, \cdots, E_r \cdot F_1, \cdots, E_r \cdot F_d\}$ ainsi que les coefficients de s, on résout alors $H \cdot e^T = s$ obtenant ainsi un système linéaire sur $\mathbb{F}_q$ dont les $n \cdot r$ inconnues sont les $e_{i,j}$ et pour lequel on a $(n-k) \cdot r \cdot d$ équations.
4. **Calcul du message**
   On résout $x \cdot G = y - e$ (dans $\mathbb{F}_{q^m}$).

Le fait que $\dim_{\mathbb{F}_q}(E \cdot F) = r \cdot d$ avec une bonne probabilité est donné par la proposition 2.37 et le fait que $E = S_1 \cap S_2 \cap \cdots \cap S_d$ est donné par la proposition 2.39.

Le cas à traiter est celui de $\dim_{\mathbb{F}_q}(S) = r \cdot d$ et on a la proposition suivante :

Proposition 2.43. *La probabilité que $s_1, \cdots, s_{n-k}$ n'engendre pas $E \cdot F$ est majorée par $q^{1+(n-k)-r\cdot d}$.*

Il est donc possible de rendre la probabilité d'erreur de cet algorithme de décodage aussi petit que toute valeur fixée au préalable en choisissant bien les paramètres du code. Et l'étape qui est la plus critique est bien donnée par la proposition 2.43.

Finalement, le comportement de l'algorithme de décodage est caractérisé par le théorème suivant :

Théorème 2.44. *Soit H la matrice de parité $(n-k) \times n$ d'un code LRPC dont les coefficients engendrent un $\mathbb{F}_q$-espace vectoriel de petite dimension $d \geqslant 2$ dans $\mathbb{F}_{q^m}$, alors l'algorithme 2.6 décode une erreur aléatoire de rang k telle que $r \cdot d \leqslant n-k$ avec une probabilité d'échec en $q^{-(n-k+1-r\cdot d)}$ et une complexité dans $\mathcal{O}(r^2 \cdot (4 \cdot d^2 \cdot m + n^2))$.*

C'est l'existence de cet algorithme de décodage et le peu de structure de ces codes qui nous ont permis de les utiliser pour concevoir des primitives cryptographiques.

2.2. Cryptographie basée sur les codes

La cryptographie basée sur les codes a démarré avec l'introduction du protocole de McEliece [37]. Le principe du protocole de McEliece est assez simple. C'est un protocole de chiffrement asymétrique. La clé privée est un code $\mathcal{C} \in \mathbb{F}_q^n$ pour lequel vous connaissez une matrice génératrice G et un algorithme de décodage jusqu'à la distance t, i.e. vous disposez d'un algorithme `decod` tel que si $\boldsymbol{e}$ est un vecteur de poids $\leqslant t$ et $\boldsymbol{y} \in \mathcal{C}$ alors $\texttt{decod}(\boldsymbol{y} + \boldsymbol{e}, t)$ retourne bien $\boldsymbol{y}$ ainsi que 2 matrices U et V (les matrices de masquages) telles que U est inversible et V est une isométrie.

La clé publique est alors la matrice $G' = U \cdot G \cdot V$ et t, pour envoyer $\boldsymbol{c} \in \mathbb{F}_q^k$, on calcule $\boldsymbol{c}^T \cdot G' = \boldsymbol{y}$ et on ajoute $\boldsymbol{e} \in \mathbb{F}_q^n$ de poids $\leqslant t$ pour obtenir $\boldsymbol{z} = \boldsymbol{y} + \boldsymbol{e}$ qui est le chiffré envoyé. Ainsi, pour recouvrer $\boldsymbol{c}$, on calcule $\boldsymbol{z}^T \cdot U^{-1} = \boldsymbol{c}^T \cdot G' \cdot U^{-1} + \boldsymbol{e}^T \cdot U^{-1}$, or comme U est une isométrie, on a $\boldsymbol{e}^T \cdot U^{-1}$ qui est bien de poids $\leqslant t$, ainsi, en décodant $\boldsymbol{z}^T \cdot U^{-1}$, on obtient $\boldsymbol{c}^T \cdot V$ et comme V est inversible, en multipliant à droite par V^{-1}, on retrouve $\boldsymbol{c}$. La sécurité du protocole tient à la difficulté de distinguer la matrice G' de celle d'un code aléatoire. En effet, s'il est possible de retrouver G (ou toute matrice ayant la même image) à partir de G' alors on dispose d'une attaque sur la clé privée. La sécurité du chiffrement repose quant à elle sur la difficulté du décodage d'un code aléatoire.

Un des inconvénients de cette approche est que les clés publiques et privées sont des matrices et donc des objets de très grande taille. Pour pallier ce problème, la démarche est d'utiliser des codes structurés (généralement des codes algébriques). Par exemple, si la matrice génératrice de $\mathcal{C}$ est une matrice structurée de rang de déplacement r, on aura besoin de $r \cdot n$ coefficients pour la décrire et donc $r \cdot n \cdot \log(q)$ au lieu de $k \cdot n \cdot \log(q)$ bits en espace. Par contre, il est difficile de gagner sur la clé publique (qui doit être indistinguable d'un code aléatoire et donc la clé publique est de taille $k \cdot n$). Par exemple la connaissance des points d'évaluation permet de reconstruire la matrice de Vandermonde qui est la matrice génératrice d'un code de Reed-Solomon (qui est de rang de déplacement 1) mais les codes Reed-Solomon sont si structurés qu'il semble difficile de les masquer. On leur préfère les codes quasi-cycliques pour lesquels il n'y a pas d'attaques très efficaces connues. C'est une façon de faire un compromis entre efficacité (taille des clés et complexité du décodage) et la sécurité (indistinguabilité).

2.2.1. McEliece protocole basé sur les LRPC

Les LRPC présentent plusieurs atouts importants : il y a un algorithme de décodage, ils ne sont pas très structurés du point de vue algébrique et le décodage d'un code aléatoire en métrique rang semble un problème difficile. C'est pourquoi nous avons proposé un protocole de type McEliece basé sur les LRPC [24]. La sécurité du système dépend alors de la difficulté à décoder un code aléatoire en métrique rang et à masquer un code ayant une matrice de parité dont le support est de petite dimension.

2.2.2. Protocole d'échange de clés basé sur le problème 2.7

Il s'agit là d'un KEM (Key Exchange Manager) proposé pour l'appel à propositions « post-quantique » du NIST : LAKE [2]. Il s'agit de donner un exemple de cryptosystème inspiré des travaux sur les LRPC. Un schéma d'encapsulation de clé est un triplet (KeyGen,Encap,Decap) d'algorithmes probabilistes qui est associé à un espace de clé $\mathcal{K}$. L'algorithme de génération de clé KeyGen engendre une paire de clés (pk,sk) où pk est une clé publique et sk est une clé privée. L'algorithme d'encapsulation Encap utilise la clé publique pk pour produire une encapsulation c et une clé $K \in \mathcal{K}$. Enfin, Decap utilise la clé privée sk et l'encapsulation c pour retrouver la clé K ou retouner une erreur $\perp$.

On suppose qu'on dispose d'une fonction de hachage G. Je décris maintenant les trois algorithmes :

Algorithme KeyGen

Entrées : 1^λ
Sorties : $(\mathsf{pk}=(H,P),\mathsf{sk}=(U,V))$

- Choisir un polynôme $P \in \mathbb{F}_q[X]$ irréductible de degré n.
- Choisir un $\mathbb{F}_q$-sous-espace vectoriel $F \subset \mathbb{F}_{q^m}$ choisi selon une distribution uniforme et un couple (U,V) de polynômes de $\mathbb{F}_{q^m}[X]$ de degré $n-1$ à coefficients dans F
- Calculer $H = U^{-1} \cdot V \bmod P$

Renvoyer ; $((H,P),(U,V))$

Algorithme Encap

Entrées : pk
Sorties :

- Choisir uniformément un $\mathbb{F}_q$-sous-espace vectoriel E de $\mathbb{F}_{q^m}$ de dimension r et choisir un couple (R_1,R_2) de polynômes de $\mathbb{F}_{q^m}[X]$ de degré $n-1$ mais à coefficients dans E.
- Calculer $\mathsf{c}=R_1+R_2 \cdot H \bmod P$
- Calculer $K = G(E)$

Renvoyer : (K,c)

Algorithme Decap

Entrées : sk
Sorties : K

- Calculer $U \cdot \mathsf{c}=U \cdot R_1 + V \cdot R_2 \bmod P$ et retrouver E avec le **calcul du support de l'erreur** de l'algorithme 2.6
- Calculer $K = G(E)$

Renvoyer : K.

Ainsi, Alice utilise `GenKey` pour avoir une paire (`pk,sk`) et elle envoie `pk` à Bob. Bob utilise alors `Encap` avec `pk` en entrée pour calculer K et `c`. Bob envoie `c` à Alice et celle-ci utilise `Decap` avec `c` en entrée pour calculer K. Ainsi avec une bonne probabilité (voir [2]), Alice et Bob partagent la connaissance de K.

Même si les codes LRPC ne sont pas explicitement utilisés dans ce protocole, le principe est largement inspiré du décodage de ces codes.

2.3. Autres travaux sur le domaine

2.3.1. Un protocole basé sur la factorisation des polynômes tordus

Dans [12], nous avons proposé un protocole d'échange de clés du type Diffie-Hellman basé sur un problème de factorisation dans les anneaux de polynômes tordus.

Protocole de Diffie-Hellman

Soit X et Y deux ensembles et $f: X \times Y \to X$ une application telle que pour tout $x \in X$ et pour tout $(a,b) \in Y^2$ on ait $f(f(x,a),b) = f(f(x,b),a)$ et on suppose que connaissant $f(x,a)$ et $f(x,b)$ il est difficile de calculer $f(f(x,a),b)$ (c'est le problème DH). Cette hypothèse implique que connaissant x et $f(x,a)$ il est difficile de retrouver a (ce problème-ci sera le problème du calcul d'une section ou CS).

Dans le protocole de Diffie-Hellman, Alice et Bob ont en commun $x \in X$ qui est public. Alice choisit $a \in Y$ et envoie $y_a = f(x,a)$ à Bob qui de son coté choisit $b \in Y$ et envoie $y_b = f(x,b)$ à Alice. Dès lors, Alice calcule $f(y_b,a) = f(f(x,b),a)$ et Bob calcule $f(y_a,b) = f(f(x,a),b)$ or par hypothèse ces deux valeurs sont les mêmes. Ainsi Alice et Bob partagent la connaissance de $f(f(x,a),b)$. Quelqu'un (un espion) qui aurait intercepté toutes les communications connaîtrait x, $f(x,a)$ et $f(x,b)$. Alors pour connaître le secret partagé par Alice et Bob il devrait calculer $f(f(x,a),b)$ à partir de $x, f(x,a)$ et $f(x,b)$, c'est-à-dire résoudre DH qu'on a supposé difficile.

L'instance initialement proposée par Diffie et Hellman est pour $X = G$ un groupe, $Y = \mathbb{N}$ et $f(x,a) = x^a$. Ainsi, $g \in G$ est publique et Alice envoie à Bob g^a et Bob envoie à Alice g^b, ainsi, Alice calcule alors $(g^b)^a = g^{a \cdot b}$ et Bob calcule $(g^a)^b = g^{a \cdot b}$. Le problème de calculer $g^{a \cdot b}$ connaissant g^a et g^b est le problème généralement connu comme problème de Diffie-Hellman. Le problème de calculer a connaissant g et g^a est le problème du logarithme discret (LD) qui est le problème (CS) dans le cas particulier de l'exponentiation dans un groupe G.

Clairement, les instances de (DH) sont toujours plus faibles que celles de (CS). Néanmoins, on ne connaît pas d'attaque plus efficace que celle sur le logarithme discret pour le problème de Diffie-Hellman pour l'exponentiation dans les groupes.

Factorisation dans R

Soit a et $b \in R$, alors on a $a \cdot b = b' \cdot a'$ avec $a' \neq a$ et $b' \neq b$ à moins que a et b ne commutent. Autrement dit, la factorisation dans R n'est pas unique. Remarquons que $(a \cdot X^i) \cdot (b \cdot X^j) = (\theta^i(b) \cdot X^j) \cdot (\theta^{-j}(a) \cdot X^i)$. Ainsi, généralement,la factorisation n'est pas unique.

Définition 2.45. *Soit P et $Q \in R$, on dit que P est similaire à Q si les R-modules à gauche (ou à droite) $R/(P)$ et $R/(Q)$ sont isomorphes.*

Si nous n'avons pas unicité de la factorisation, on a la proposition suivante [32] :

Théorème 2.46. *Si $P \in R$ admet deux décompositions $P = P_1 \cdots P_n = P'_1 \cdots P'_m$ en produit de facteurs irréductibles alors $n = m$ et il existe une permutation $\sigma \in S_n$ telle que $P_{\sigma(i)}$ et P'_i soient similaires pour tout $i \in \{1, \cdots, n\}$.*

Si les facteurs ne commutent pas entre eux, on comprend qu'on aura autant de factorisations que d'éléments dans S_n.

Protocole de partage de secret basé sur les polynômes tordus

On considère $g = \prod_{i=1}^{m} c_i \in R$ un polynôme tordu admettant m facteurs irréductibles et soit d un degré qui servira de paramètre de sécurité. Soient A et B des sous-ensembles de R tels que pour tout a_1 et $a_2 \in A$ on a $a_1 \cdot a_2 = a_2 \cdot a_1$ et pour tout b_1 et $b_2 \in B$ on a $b_1 \cdot b_2 = b_2 \cdot b_1$ qui seront appelés ensembles commutants qui ne doivent pas être inclus dans $\mathbb{F}_q[X, \theta]$ qui est le centre de R. Le polynôme g et les ensembles commutants A et B sont publics. Alice choisit $a_1, \cdots, a_l \in A$ et $b_1, \cdots, b_k \in B$ tels que $m_1 = a_1 \cdots a_l \cdot g \cdot b_1 \cdots b_k$ soit de degré d et envoie m_1 à Bob tandis que Bob choisit $a'_1, \cdots, a'_u \in A$ et $b'_1, \cdots, b'_v \in B$ tels que $m_2 = a'_1 \cdots a'_u \cdot g \cdot b_1 \cdots b'_v$ soit de degré d et envoie m_2 à Alice . Dès lors Alice calcule $a_1 \cdots a_l \cdot m_2 \cdot b_1 \cdots b_k = a_1 \cdots a_k \cdot a'_1 \cdots a'_u \cdot g \cdot b'_1 \cdots b'_v \cdot b_1 \cdots b_k$ et Bob calcule $a'_1 \cdots a'_u \cdot m_1 \cdot b'_1 \cdots b'_v = a'_1 \cdots a'_u \cdot a_1 \cdots a_l \cdot g \cdot b_1 \cdots b_k \cdot b'_1 \cdots b'_v$ or ces deux valeurs sont les mêmes puisque a_i et a'_j commutent entre eux et b_i et b'_j commutent entre eux. Ainsi Alice et Bob partagent la valeur $m = a_1 \cdots a_l \cdot a'_1 \cdots a'_u \cdot g \cdot b_1 \cdots b_k \cdot b'_1 \cdots b'_v$ et quelqu'un qui aurait espionné les échanges connaîtrait $g, A, B, m_1 = a_1 \cdots a_l \cdot g \cdot b_1 \cdots b_k$ et $m_2 = a'_1 \cdots a'_u \cdot g \cdot b'_1 \cdots b'_v$. La sécurité de ce protocole repose donc sur la difficulté de calculer m connaissant m_1 et m_2. C'est le problème Diffie-Hellman associé qui sera noté (DH-PT). Le problème de « type » logarithme discret est une sous-instance du problème de factorisation des polynômes tordus (FPT).

On récapitule ci-dessous le protocole :

◇ Diffie-Hellman avec des polynômes tordus

1. Alice et Bob sont d'accord $d \in \mathbb{N}$, $g \in R$ de degré d et A et B
2. Alice : $L_A \longleftarrow a_1 \cdots a_l$ et $R_A \longleftarrow b_1 \cdots b_k$ puis $m_1 \longleftarrow L_A \cdot g \cdot R_A$ enfin $m_1 \rightsquigarrow$ Bob
   Bob : $L_B \longleftarrow a'_1 \cdots a'_u$ et $R_B \longleftarrow b'_1 \cdots b'_v$ puis $m_2 \longleftarrow L_B \cdot g \cdot R_B$ enfin $m_2 \rightsquigarrow$ Alice
3. Alice : $m \longleftarrow L_A \cdot m_2 \cdot R_A$
   Bob : $m \longleftarrow L_B \cdot m_1 \cdot R_B$

À partir de ce protocole, on peut dériver un protocole de chiffrement en utilisant le schéma de El-Gammal.

Protocole de chiffrement basé sur les polynômes tordus

Le schéma de El-Gammal permet de dériver un schéma de chiffrement d'un protocole de type Diffie-Hellman. On suppose qu'Alice et Bob sont d'accord sur $d \in \mathbb{N}$, A et B deux commutants et sur $h : R \longrightarrow \mathbb{F}_{q^m}^n$. Alice construit alors ses clés en choisissant $g_A \in R$ de degré d en multipliant des facteurs irréductibles ensemble, obtenant L_A à partir d'éléments de A et R_A à partir d'éléments dans B constituant sa clé privée $\{L_A, R_A\}$ et sa clé publique $\{g_A, L_A \cdot g_A \cdot R_A\}$. Bob récupère la clé publique de Alice et pour chiffrer un message m de longueur n sur $\mathbb{F}_{q^m}$, Bob choisit L_B à partir d'éléments de A et R_B à partir d'éléments de B puis il calcule $G = L_B \cdot L_A \cdot g_A \cdot R_A \cdot R_B$ et $g_B = L_B \cdot g_A \cdot R_B$ et envoie à Alice (c, g_B) où $c = m + h(G)$. En recevant (c, g_B) Alice peut calculer $G = L_A \cdot g_B \cdot R_A$ et donc elle recouvre m en calculant $c - h(G)$.

◇ El-Gammal avec des polynômes tordus

1. Alice et Bob partagent $d \in \mathbb{N}$ et A et B
2. Alice : $g_A \longleftarrow \text{rand}(d)$, $L_A \longleftarrow a_1 \cdots a_l$ et $R_A \longleftarrow b_1 \cdots b_k$ puis $m_1 \longleftarrow L_A \cdot g_a \cdot R_A$ enfin $(g_A, m_1) \rightsquigarrow$ Bob
   Bob : $L_B \longleftarrow a'_1 \cdots a'_u$ et $R_B \longleftarrow b'_1 \cdots b'_v$ puis $m_2 \longleftarrow m + h(L_B \cdot m_1 \cdot R_B)$, $G \longleftarrow L_B \cdot g_A \cdot R_B$ enfin $(G, m_2) \rightsquigarrow$ Alice
3. Alice : $m \longleftarrow m_2 - h(L_A \cdot G \cdot R_A)$

Quelques digressions sur les commutants

Si a et $b \in \mathbb{F}_{q^m}[X,\theta]$ sont des polynômes tordus commutant entre eux, alors $a+b$ commute avec a et b ainsi que $a \cdot b$. L'idée est donc de choisir des éléments commutant ensemble, de petits degrés et n'étant pas dans le centre (ici on les prend donc en dehors de $\mathbb{F}_q[X,\theta]$) et d'en faire des combinaisons par sommation pour générer des facteurs de petits degrés. Ainsi, on ne stocke pas A et B mais des éléments permettant d'engendrer ces ensembles.

Construire des générateurs d'un commutant n'est pas facile, mais comme c'est un pré-calcul, cela peut être fait par force brute. On considère un ensemble $G_0 \subset \mathbb{F}_{q^m}[X,\theta]$ avec beaucoup d'éléments et on tire $g_0 \in G_0 \setminus \mathbb{F}_q[X,\theta]$. On note $\mathrm{Com}(g_0) = \{h \in \mathbb{F}_{q^m}[X,\theta] |\ g_0 \cdot h = h \cdot g_0\}$. On pose alors $G_1 = G_0 \setminus \mathrm{Com}(g_0)$ puis on tire $g_1 \in G_1$ et on itére jusqu'à ce que G_i soit vide.

Algorithme 2.7

Entrée : $G_0 \subset \mathbb{F}_{q^m}[X,\theta]$

1. $i \leftarrow 0$; tirer $g \in G_0$; $\mathcal{S} \leftarrow \{g\}$
2. tant que $G_i \neq \varnothing$ faire
   i. $i \leftarrow i+1$
   ii. $G_i \leftarrow G_{i-1} \cap \mathrm{Com}(g)$
   iii. tirer $g \in G_i$
   iv. $\mathcal{S} \leftarrow \mathcal{S} \cup \{g\}$

Sortie : $\mathcal{S}$

À chaque étape, les éléments de $\mathcal{S}$ commutent entre eux, ce qui donne le résultat suivant :

Lemme 2.47. *La sortie de l'algorithme 2.7 est un ensemble d'éléments commutant entre eux.*

Attaques sur le problème FPT et travaux relatifs

Pour commencer, nous n'avons pas pu déterminer d'attaque spécifique au problème DH-PT et nous avons donc étudié des attaques sur le problème de factorisation (FPT). Le problème peut être présenté comme suit :

Problème 2.8. Soient $G \in \mathbb{F}_{q^m}[X,\theta]$, A et B deux commutants connus de $\mathbb{F}_{q^m}[X,\theta]$, soit alors L construit en combinant des éléments de A et R en combinant des éléments de B, il s'agit alors de calculer L et R connaissant $P = L \cdot G \cdot R$.

L'équation $P = L \cdot G \cdot R$ sera appelée l'équation clé.

La première attaque testée est basée sur la factorisation et plus précisément l'algorithme donné par Giesbrecht dans [27]. Cet algorithme est utilisé pour calculer des candidats pour R avec l'approche suivante :

Algorithme 2.8

Données : $P = L \cdot G \cdot R$

1. Utiliser l'algorithme de [27] pour calculer un facteur à droite R' de P.
2. Calculer le quotient P' à droite de P divisé par R' puis la division à droite de P' par G, si le reste est non nul, alors on retourne à l'étape 1.
3. Si le reste était nul, on note L' le quotient de P' divisé à droite par G.

Sortie : (L', R').

Si L et R ne contiennent pas de facteurs commutant avec G, alors la sortie de l'algorithme donnera la solution du problème 2.8. L'étape 1 de l'algorithme se fait en temps polynomial, mais le nombre de facteurs à explorer pour R' est potentiellement exponentiel en le nombre de facteurs de L et R. Les paramètres peuvent être choisis pour que le coût de cette approche soit rédhibitoire.

Il y a une autre attaque tirant profit du fait que L et R sont des combinaisons d'éléments de A et B respectivement. Supposons que $A=\{a_1,\cdots,a_c\}$ et que $B=\{b_1,\cdots,b_d\}$ alors $L=\sum_{i=1}^{c}\lambda_i\cdot a_i$ et $R=\sum_{j=1}^{d}\mu_j\cdot b_j$ et donc l'équation clé devient :

$$P=\left(\sum_{i=1}^{c}\lambda_i\cdot a_i\right)\cdot G\cdot\left(\sum_{j=1}^{d}\mu_j\cdot b_j\right). \tag{2.6}$$

Les inconnues sont les λ_i et les μ_j. On a donc $c+e$ inconnues et $2\cdot\deg(P)$ inconnues. Cette dépendance linéaire entre le nombre de variables, le nombre d'équations et les degrés des facteurs rend l'attaque inopérante dès que les tailles grossissent.

Comme on connaît le degré de $G\cdot R$, on peut choisir un polynôme tordu f de degré plus grand que le degré de $G\cdot R$. Si f est choisi aléatoirement, alors f sera premier avec L et L sera inversible modulo f. Une dernière approche consiste à réduire le problème à un problème linéaire en multipliant à gauche les deux membres de l'équation clé par L^{-1} modulo f, on obtient :

$$L^{-1}\cdot P=G\cdot R \bmod f. \tag{2.7}$$

Dans ce cas, on est face à un système linéaire dont les inconnues sont les coefficients de L^{-1} et R. Faire le calcul modulo f va augmenter le nombre de solutions du problème. Donner une base d'un espace vectoriel contenant L^{-1} connaissant une base de celui contenant L est possible, permettant de donner le nombre d'inconnues de ce système mais il est facile de se protéger de cette attaque en pratique.

2.3.2. Les sous-codes sous-espaces pour la cryptographie basée sur les codes

Comme je l'ai déjà évoqué, les codes de Reed-Solomon et les codes de Gabidulin sont des formes analogues pour deux métriques différentes et ils présentent une très forte structure algébrique qu'il est difficile de dissimuler. Ils sont, par conséquent, sensibles aux attaques par distingueurs. Avec Thierry Berger et Philippe Gaborit [6], puis avec Thierry Berger, Cheikh Thiécomba Gueye et Jean Bélo Klamti [7] nous avons proposé des protocoles permettant de dissimuler cette structure en modifiant le code pour les codes de Gabidulin et les codes Reed-Solomon respectivement.

2.4. TRAVAUX EN COURS : qCRT

Depuis presque cinq ans, je travaille avec différents collègues (Alain Couvreur, Philippe Gaborit, Adrien Hauteville, Victor Dysern, Jean-Pierre Cancès) sur un sujet qui devrait donner une première communication bientôt : les q-polynonômes chineses remainder codes (qCRT). J'évoque ici rapidement ces codes, leurs constructions et leur décodage uniquement. Il s'agit d'une nouvelle famille de codes en métrique rang.

2.4.1. Théorème des restes chinois pour les polynômes linéarisés

Dans un premier temps, je rappellerai le théorème des restes chinois pour les q-polynômes en utilisant librement les notions de RGCD et LLCM abordées dans la sous-section 2.1.3. Soient f_1 et $f_2\in\mathbb{F}_{q^m}\langle X^q\rangle$ de q-degré respectivement d_1 et d_2. Pour tout polynôme linéarisé g, on note $\pi_i(g)$ son reste par la division à droite par f_i. On suppose que f_1 et f_2 sont premiers entre eux, c'est-à-dire qu'ils vérifient une relation de Bézout $S_1\circ f_1+S_2\circ f_2=X$ avec q-deg$(S_1)<d_2$ et q-deg$(S_2)<d_1$ en utilisant l'algorithme 2.4. Remarquons que S_1 est l'inverse de f_1 modulo f_2, i.e. $\pi_2(S_1\circ f_1)=X$ et que S_2 est l'inverse de f_2 modulo f_1, i.e. $\pi_1(S_2\circ f_2)=X$.

LEMME 2.48. *Soit $g\in\mathbb{F}_{q^m}\langle X^q\rangle$ alors $g-\pi_1(g)\circ S_1\circ f_1-\pi_2(g)\circ S_2\circ f_2$ est dans l'idéal à gauche $(f_1\vee_l f_2)_l$.*

Démonstration. Si on calcule le reste de $g-\pi_2(g)\circ S_1\circ f_1-\pi_1(g)\circ S_2\circ f_2$ par la division à droite par f_1, on a $\pi_1(g)-\pi_1(g)\circ S_2\circ f_2$ mais comme S_2 est l'inverse de f_2 modulo f_1, on a $g-\pi_2(g)\circ S_1\circ f_1-\pi_1(g)\circ S_2\circ f_2=\pi_1(g)-\pi_2(g)=0$. Cela signifie que f_1 divise $g-\pi_2(g)\circ S_1\circ f_1-\pi_1(g)\circ S_2\circ f_2$ à droite. On peut faire exactement la même chose en échangeant les rôles de f_1 et f_2. On en déduit que $g-\pi_2(g)\circ S_1\circ f_1-\pi_1(g)\circ S_2\circ f_2$ est divisible à droite par f_1 et f_2 et qu'il est donc un multiple à gauche de $f_1\vee_l f_2$. □

Pour tout $g\in\mathbb{F}_{q^m}\langle X^q\rangle$, on note $\pi_3(g)$ le reste de la division à droite de g par $f_1\vee_l f_2$. On est maintenant capable de donner le théorème des restes chinois modulo deux polynômes linéarisés.

Théorème 2.49. *Soit $g\in\mathbb{F}_{q^m}\langle X^q\rangle$ de q-degré strictement inférieur à d_1+d_2 alors $\pi_3(\pi_2(g)\circ S_1\circ f_1+\pi_1(g)\circ S_2\circ f_2)=g$.*

Démonstration. Par le lemme 2.48, on sait que $g-(\pi_2(g)\circ S_1\circ f_1+\pi_1(g)\circ S_2\circ f_2)\in(f_1\vee_l f_2)_l$ et comme q-deg$(g)<d_1+d_2$ on a $\pi_3(g-(\pi_2(g)\circ S_1\circ f_1+\pi_1(g)\circ S_2\circ f_2))=\pi_3(g)-\pi_3(\pi_2(g)\circ S_1\circ f_1+\pi_1(g)\circ S_2\circ f_2)=g-\pi_3(\pi_2(g)\circ S_1\circ f_1+\pi_1(g)\circ S_2\circ f_2)=0$. On en déduit alors que $g=\pi_3(\pi_2(g)\circ S_1\circ f_1+\pi_1(g)\circ S_2\circ f_2)$. □

Pour aller plus avant, il est important de comprendre que même si trois polynômes linéarisés f_1, f_2 et f_3 sont premiers entre eux deux à deux, $f_1\vee_l f_2$ ne l'est pas forcément avec f_3. Considérons $f_1=X^q-\zeta^{q-1}\cdot X$, $f_2=X^q-\xi^{q-1}\cdot X$ et $f_3=X^q-(\zeta+\xi)^{q-1}\cdot X$. Ces polynômes linéarisés sont premiers entre eux dès que $\zeta\neq\xi$. Puisque $\zeta\in\ker(f_1)$ et $\xi\in\ker(f_2)$, on a $\zeta+\xi\in\ker(f_1\vee_l f_2)$ et donc $f_3\in(f_1\vee_l f_2)_g$. Cela explique pourquoi l'extension du théorème des restes chinois n'est pas aussi directe qu'on pourrait le croire dans le cas de trois polynômes ou plus. C'est pourquoi je le présente ici complétement.

Soit $f_1,f_2,\cdots,f_k\in\mathbb{F}_{q^m}\langle X^q\rangle$ de q-degré respectivement $d_1,d_2,\cdots,d_k$ et tels que $f_1\vee_l f_2\vee_l\cdots\vee_l f_{i-1}$ soit premier avec f_i pour tout $i\in\{3,\cdots,k\}$. Pour tout $g\in\mathbb{F}_{q^m}\langle X^q\rangle$, on note $\pi_i(g)$ le reste de g par la division à droite par f_i . On définit $h_1=f_1$, $h_{i+1}=h_i\vee_l f_i$ pour $i\in\{2,\cdots,k-1\}$ et $\pi_{1,i}(g)$ le reste de g par la division à droite par h_i pour tout $i\in\{2,\cdots,k\}$. On considère aussi $S_{i,1}$ et $S_{i,2}$ les coefficients de la relation de Bézout $S_{i,1}\circ h_i+S_{i,2}\circ f_i=X$ entre h_i et f_i (puisqu'ils sont supposés premiers entre eux).

Théorème 2.50. *Avec les notations ci-dessus, si $g\in\mathbb{F}_{q^m}\langle X^q\rangle$ alors en définissant $g_1=\pi_1(g)$ et $g_i=\pi_{1,i}(g_{i-1}\circ S_{1,i}\circ h_i+\pi_i(g)\circ S_{2,i}\circ f_i)$ on a $g-g_k\in(f_1\vee_l\cdots\vee_l f_k)_l$ et alors si q-deg$(g)<q$-deg$(f_1\vee_l\cdots\vee_l f_k)=d$ alors $g_k=g$.*

Démonstration. On montre ce résultat par récurrence. Tout d'abord, on a $g-g_1\in(h_1)_l$ puisque $h_1=l_1$ et $g=\pi_1(g)$ et si le q-degré de g est inférieur à celui de f_1 alors $g_1=g$. Supposons maintenant que $g-g_i\in(h_i)$ alors en appliquant le lemme 2.48, on a $g_{i+1}=\pi_{1,i}(g_i\circ S_{1,i}\circ h_i+\pi_{i+1}(g)\circ S_{2,i}\circ f_i)$ qui est tel que $g-g_{i+1}\in(h_{i+1})_l$. On montre alors que $g-g_{i+1}\in(h_{i+1})_l$ pour tout $i\leqslant k-1$. Pour $i=k-1$, on a $g-g_k\in(f_1\vee_l\cdots\vee_l f_k)$ et alors si q-deg$(g)<d$ on a $g-g_k=0$. □

Ce théorème se traduit directement algorithmiquement. Il permet d'avoir une approche de terminaison précoce (« early termination »). Il permet aussi de contrôler le q-degré des polynômes linéarisés utilisés dans l'algorithme. Il est équivalent au résultat suivant. On note $b_i=\vee_{\substack{l\\ j\neq i}} f_j$ et on suppose que $f_i\wedge_r b_i=X$ (ce qui est équivalent aux hypothèses du théorème 2.50) et on note $S_{1,i}\circ b_i+S_{2,i}\circ f_i=X$ (i.e. $S_{1,i}$ est l'inverse de b_i mod f_i).

Théorème 2.51. *On note $G=\pi_{1,k}\left(\sum_{i=1}^{k}\pi_i(g)\circ S_i\circ b_i\right)$ qui vérifie $g-G\in(f_1\vee_l\cdots\vee_l f_k)_l$ et si q-deg$(g)<d$ alors $G=g$.*

2.4.2. Codes restes chinois sur les polynômes linéarisés

On présente maintenant la construction de notre nouvelle famille de codes.

Soit $f_1, \cdots, f_s \in \mathbb{F}_{q^m}\langle X^q\rangle$ de q-degré respectivement $d_1, \cdots, d_s$, on note $\pi_i : \begin{cases} \mathbb{F}_{q^m}\langle X^q\rangle \longrightarrow \mathbb{F}_{q^m}\langle X^q\rangle \\ g \longmapsto \pi_i(g) \end{cases}$ où $\pi_i(g)$ est le reste de la division à droite de g par f_i, pour tout $i \in \{1, \cdots, s\}$. On définit alors l'application :

$$\Pi = \pi_1 \times \cdots \times \pi_l : \begin{cases} \mathbb{F}_{q^m}\langle X^q\rangle \longrightarrow \mathbb{F}_{q^m}\langle X^q\rangle/(f_1)_l \times \cdots \times \mathbb{F}_{q^m}\langle X^q\rangle/(f_s)_l \\ g \longmapsto (\pi_1(g), \cdots, \pi_s(g)) \end{cases}.$$

L'application Π est une application $\mathbb{F}_q$-linéaire. Soit $A \in \mathbb{F}_{q^m}\langle X^q\rangle$, on définit l'application :

$$\mathcal{M}_A : \begin{cases} \mathbb{F}_{q^m}\langle X^q\rangle \longrightarrow \mathbb{F}_{q^m}\langle X^q\rangle \\ g \longmapsto g \circ A \end{cases}.$$

L'application $\mathcal{M}_A$ est également $\mathbb{F}_q$-linéaire de façon que l'application suivante soit aussi $\mathbb{F}_q$-linéaire :

$$\Psi_A = \Pi \circ \mathcal{M}_A : \begin{cases} \mathbb{F}_{q^m}\langle X^q\rangle \longrightarrow \mathbb{F}_{q^m}\langle X^q\rangle/(f_1)_l \times \cdots \times \mathbb{F}_{q^m}\langle X^q\rangle/(f_s)_l \\ g \longmapsto (\pi_1(g \circ A), \cdots, \pi_s(g \circ A)) \end{cases}. \tag{2.8}$$

Soit $d \in \mathbb{N}$, on note $\mathbb{F}_{q^m}\langle X^q\rangle_d$ l'ensemble des polynômes linéarisés de q-degré strictement inférieur à d. On peut maintenant définir les codes qCRT.

DÉFINITION 2.52. *Le code* qCRT *associé à d, A et $F = (f_1, \cdots, f_s)$ est $\mathcal{C}_{F,d} = \Psi_A(\mathbb{F}_{q^m}\langle X^q\rangle_d)$.*

On suppose que $F = (f_1, \cdots, f_s) \in \mathbb{F}_{q^m}\langle X^q\rangle^s$ est tel que f_i est premier avec $h_i = f_1 \vee_l \cdots \vee_l f_{i-1}$ pour tout $i \in \{2, \cdots, s\}$ (si bien qu'on peut appliquer le théorème 2.50). On suppose que f_i est de q-degré d_i pour tout $i \in \{1, \cdots, s\}$ et on note $n = d_1 + \cdots + d_s$. Supposons que A est de q-degré α, i.e. $A = \sum_{i=0}^{\alpha} a_i \cdot X^{q^i}$.

PROPOSITION 2.53. *Soit $k < n - \alpha$, on considère le code qCRT $\mathcal{C}_{F,k}$, c'est un code $\mathbb{F}_q$-linéaire en métrique rang de dimension k et de longueur n.*

Il est nécessaire d'introduire le polynôme A pour augmenter la distance minimale des codes construits.

2.4.3. Idée de l'algorithme de décodage

L'agorithme de décodage présenté ici est un algorithme probabiliste assez analogue à celui utilisé pour les codes LRPC ou les codes simples. On procède en deux temps. Dans un premier temps, on détermine le support de l'erreur, puis on est ramené à résoudre un système linéaire.

Soit $F = (f_1, \cdots, f_s) \in \mathbb{F}_q\langle X^q\rangle$ avec f_i de q-degré d_i pour tout $i \in \{1, \cdots, s\}$ et satisfaisant les conditions d'application du théorème 2.50. On remarque bien qu'on a pris les f_i à coefficients dans $\mathbb{F}_q$ et non $\mathbb{F}_{q^m}$.

On note $n = \sum_{i=1}^{s} d_i$ et soient $k < d$ un entier, $\mathcal{C} = \mathcal{C}_{F,k}$ un code qCRT associé à F et k. Soit $\boldsymbol{m} \in \mathbb{F}_{q^m}\langle X^q\rangle_k$, on note $\boldsymbol{c} = \Pi(\boldsymbol{m}) = (\pi_1(\boldsymbol{m}), \cdots, \pi_s(\boldsymbol{m}))$ et $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e}$ où $\boldsymbol{e}$ est l'erreur. On note Ψ : $\begin{cases} \mathbb{F}_{q^m}\langle X^q\rangle_{d_1} \times \cdots \times \mathbb{F}_{q^m}\langle X^q\rangle_{d_s} \longrightarrow \mathbb{F}_{q^m}\langle X^q\rangle_n \\ (p_1, \cdots, p_s) \longmapsto p \end{cases}$ la remontée des restes chinois, i.e. si $p \in \mathbb{F}_{q^m}\langle X^q\rangle_n$ alors $\Psi \circ \Pi(p) = p$. Les deux applications sont $\mathbb{F}_{q^m}$-linéaires si bien que $\Psi(\boldsymbol{y}) = \Psi(\boldsymbol{c} + \boldsymbol{e}) = \Psi(\boldsymbol{c}) + \Psi(\boldsymbol{e})$ et puisque $\boldsymbol{c} \in \Pi(\mathbb{F}_{q^m}\langle X^q\rangle_k)$ on a $\Psi(\boldsymbol{c}) = \boldsymbol{m}$ et donc $\Psi(\boldsymbol{y}) = \boldsymbol{m} + \Psi(\boldsymbol{e})$. On note $\Psi(\boldsymbol{e}) = \underline{E} + \overline{E}$ où $\underline{E} \in \mathbb{F}_{q^m}\langle X^q\rangle_k$ et $\overline{E} = \sum_{i=k}^{n-1} e_i \cdot X^{q^i}$. On note $\mathcal{E} = \langle E\rangle$ le $\mathbb{F}_q$-sous-espace vectoriel de $\mathbb{F}_{q^m}$ engendré par les coefficients de E et de la même façon, on note $\underline{\mathcal{E}} = \langle \underline{E}\rangle$ et $\overline{\mathcal{E}} = \langle \overline{E}\rangle$ ceux associés à $\underline{E}$ et $\overline{E}$ respectivement. Supposons que E soit de rang r, i.e. $\dim(\mathcal{E}) = r$. Alors la probabilité que $\overline{\mathcal{E}} = \mathcal{E}$ est la probabilité d'avoir une base d'un espace de dimension r en tirant $n - \alpha - k$ éléments aléatoirement dans cet espace.

2.5. Conclusion

Dans cette section, j'ai exposé quelques uns des travaux auxquels j'ai participé, montrant que des problèmes parfois très mathématiques peuvent aboutir à des applications très concrètes. J'ai montré comment les polynômes de Öre peuvent être utilisés en théorie des codes et en cryptographie de diverses façons et comment des questions algorithmiques d'algèbre non-commutative peuvent impacter des domaines beaucoup plus appliqués. J'ai également expliqué comment le travail avec des doctorants (thèses de Gaëtan Murat [40], Adrien Hauteville [29] en co-encadrement mais aussi avec des étudiants que je ne dirigeais pas comme Nicolas Aragon ou Julien Schrek) intervient dans ce processus de recherche et comment un projet à long terme s'est dessiné sur cette thématique et qu'il a encore des perspectives intéressantes qui sont en cours d'exploration. Enfin, j'ai montré ma volonté de diffuser mes travaux et de les valoriser le plus possible à travers des publications et des projets scientifiques bien intégrés dans la communauté scientifique.

Chapitre 3

Méthode de Weierstrass, trajectoires optimales et applications

3.1. Méthode de Weierstrass surcontrainte

Durant ma thèse, j'ai introduit une généralisation de la méthode dite de Weierstrass (ou de Dochev ou de Durand-Kerner selon les auteurs) pour l'approximation itérative des racines d'un système algébrique en intersection complète ayant un nombre fini et connu de solutions. J'en rappelle ici rapidement le principe. L'objectif est de présenter sommairement les résultats obtenus avec Mark Sciabica et Agnes Szanto sur la méthode dans le cas surcontraint (le cas où on a plus d'équations que d'inconnues) après plusieurs années d'efforts pour terminer ce travail [46].

3.1.1. Méthode Weierstrass univariée

Je donne ici une introduction très rapide à la méthode de Weierstrass dans le cas d'un polynôme d'une variable. Dans le cas de la dimension 0 intersection complète, le contexte se généralise pour que les calculs soient essentiellement les mêmes.

Je note $\mathcal{S}_d$ le groupe symétrique de d éléments. Je note alors $\sigma_i(x_1, \cdots, x_d)$ le $i^{\text{ème}}$ polynôme symétrique élémentaire (pour $i \in \{1, \cdots, d\}$), i.e. $\sigma_i(x_1, \cdots, x_n) = \sum_{j_1 < \cdots < j_i} \prod_{k=1}^{i} x_{j_k}$. Soit $f(z) \in \mathbb{C}[z]$ tel que $f(z) = \prod_{i=1}^{d} (z - z_i)$, l'égalité classique : $f(z) = z^d + \sum_{i=1}^{d} (-1)^i \sigma_i(z_1, \cdots, z_d) \cdot z^{d-i}$ permet de définir une application de $\mathbb{C}^d$ dans l'espace affine des polynômes unitaires de degré exactement d comme suit :

$$\Sigma : \begin{cases} \mathbb{C}^d \longrightarrow \mathbb{C}[z] \\ \begin{pmatrix} z_1 \\ \vdots \\ z_d \end{pmatrix} \longmapsto \begin{pmatrix} 1 \\ -\sigma_1(z_1, \cdots, z_d) \\ \vdots \\ (-1)^d \cdot \sigma_d(z_1, \cdots, z_d) \end{pmatrix} \end{cases} \tag{3.1}$$

en écrivant les polynômes dans la base monomiale $\{z^d, z^{d-1}, \cdots, 1\}$. C'est une application différentiable presque partout (partout en dehors de la « diagonale », i.e. les vecteurs de $\mathbb{C}^d$ ayant deux coordonnées égales) et un homéomorphisme local qui fait que Σ est un revêtement presque partout (dont les fibres sont de cardinal $d!$). Si $f(z) = z^d + a_1 \cdot z^{d-1} + \cdots + a_d$, on note :

$$F_f : \begin{cases} \mathbb{C}^d \longrightarrow \mathbb{C}[z] \\ \begin{pmatrix} z_1 \\ \vdots \\ z_d \end{pmatrix} \longmapsto \Sigma(z_1, \cdots, z_d) - \begin{pmatrix} 1 \\ a_1 \\ \vdots \\ a_d \end{pmatrix} = \begin{pmatrix} 0 \\ -\sigma_1(z_1, \cdots, z_d) - a_1 \\ \vdots \\ (-1)^d \cdot \sigma_d(z_1, \cdots, z_d) - a_d \end{pmatrix}. \end{cases} \tag{3.2}$$

Si f est sans facteur carré (toutes ses racines sont distinctes) alors $F_f(z_1, \cdots, z_d) = 0$ si et seulement si $\mathcal{Z}_f := \{\zeta \in \mathbb{C} \mid f(\zeta) = 0\} = \{z_1, \cdots, z_d\}$. L'idée de la méthode de Weierstrass est d'appliquer la méthode de Newton à F_f pour approcher simultanément toutes les racines de f.

Je note $\boldsymbol{z} = (z_1, \cdots, z_d)$, on remarque alors que $DF_f(\boldsymbol{z}) = D\Sigma(\boldsymbol{z})$ de sorte que l'itération de Weierstrass est définie par :

$$\boldsymbol{z} \mapsto \boldsymbol{z} - D\Sigma(\boldsymbol{z})^{-1} \cdot F_f(\boldsymbol{z}). \tag{3.3}$$

En utilisant le fait que $F_f(\boldsymbol{z})$ est nécessairement un polynôme de degré $d-1$ (c'est la différence de deux polynômes unitaires de degré d) et la base d'interpolation de Lagrange, la formule de l'itération de Weierstrass 3.3 devient:

$$z_i \mapsto z_i - \frac{f(z_i)}{\prod_{j \neq i} (z_i - z_j)}, \forall i \in \{1, \cdots, d\}. \tag{3.4}$$

Cette formule est devenue assez populaire pour deux raisons : elle permet d'être distribuée facilement et le comportement global de la dynamique semble indiquer une convergence globale ou au moins extrêmement étendue à partir du moment où la première itération passe numériquement (la première itération consiste à projeter sur l'hyperplan $(-1)^d \cdot (z_1 + \cdots + z_d) = a_1$ ce qui peut être numériquement très compliqué). En dépit de son intérêt pratique qui a fait qu'il y a beaucoup de publications sur l'implémentation ou l'amélioration de cette itération et ce même dans le cas d'une variable, je ne connais pas d'avancée sur l'étude de la dynamique de l'itération de Weierstrass.

Durant ma thèse, j'ai généralisé cette itération au cas des intersections complètes de dimension zéro grâce à des formules explicites d'interpolation qui nous ont permis avec Bernard Mourrain de donner les analogues des fonctions symétriques élémentaires dans le cas de systèmes multivariés. Ces formules d'interpolation et de relations sont générales dans le cas de la dimension zéro et ne demandent pas d'être en intersection complète. Pourtant, une difficulté apparaît pour généraliser la méthode de Weierstrass. Il faut une base pour décomposer $F_f(z)$ afin de construire un analogue du système reliant racines et coefficients. Anne-Mercedes Bellido avait commencé à donner des formules dans certains cas particuliers [5]. Grâce aux formules d'interpolation de [39], j'ai pu donner des formules générales dans le cas de l'intersection complète multivariée et j'ai conçu dans ma thèse des méthodes locales et des méthodes par homotopie sur l'itération ainsi décrite.

Le cas surcontraint (quand on a plus d'équations que d'inconnues) est un peu plus compliqué. En effet, le cas surcontraint se réduit « facilement » au cas d'une seule équation (le cas intersection complète) à l'aide du PGCD puisque les zéros communs f et $g \in \mathbb{C}[z]$ sont les zéros de $f \wedge g$. Il faut nuancer cette apparente facilité car si dans le cadre du calcul formel, un tel résultat est satisfaisant, d'un point de vue numérique, la notion de PGCD est une notion difficile car très instable. En effet, si d et e sont les degrés respectifs de f et g, alors on sait que le résultant est un polynôme des coefficients de f et g qui s'annulent si et seulement si les deux polynômes ont une racine commune. Autrement dit, parmi les systèmes de polynômes univariés de degré d et e, ceux admettant une racine commune forment une sous-variété algébrique de codimension 1 et donc de mesure nulle. Ainsi, si f et g ont des racines communes, pour presque toute déformation des coefficients de f et g, le système déformé n'a plus de solution. Du point de vue numérique, si on ne connaît qu'une approximation des polynômes qu'on veut résoudre, le système dont on dispose n'a pas de solution, c'est-à-dire que dans le cas où les polynômes du système sont obtenus par un processus numérique, la notion de PGCD ne suffit plus. C'est ce qui a ouvert la recherche sur les PGCD approchés. Il s'agit de déterminer le système le plus « singulier » le plus proche du système dont on dispose. Par le plus singulier, j'entends celui qui admet le plus de solutions.

3.1.2. PGCD approchés

Déjà au cours de ma thèse, pour les besoin du calcul Symbolique-Numérique, je me suis intéressé aux PGCD approchés. L'approche classique pour ce problème consiste à étudier un problème de valeurs singulières sur une matrice de Sylvester ou Macaulay, une vue sur les phénomènes apparaissant avec cette approche peut être trouvée dans la thèse de David Rupprecht [47].

J'ai commencé à travailler avec Agnes Szanto et Mark Sciabica sur les liens entre la méthode de Weierstrass et les PGCD approchés lors d'un séjour à l'Université de Caroline du Sud en 2004. Après maintes difficultés techniques, le travail aboutira en 2015 à la publication de [46]. Entre-temps, nous avions proposé avec Paola Boito une autre formulation basée sur les matrices compagnons généralisées en 2011 dans [8]. Il y a un lien entre les deux approches et l'exploitation de la structure des algèbres quotients de dimension zéro. Néanmoins, je me focaliserai sur [46], l'idée principale de [8] ayant été réinvestie dans la thèse de Gaëtan Murat. Le problème est présenté dans le cadre analytique (analyse complexe), mais on peut facilement se restreindre au cas où les fonctions analytiques considérées sont des polynômes et réciproquement en utilisant le théorème de division de Weierstrass. Considérons le problème suivant :

Problème 3.1. Soit $\vec{f}=(f_1,\cdots,f_N):\mathbb{C}^n\longrightarrow\mathbb{C}^N$ une application analytique (les f_i sont des fonctions analytiques de $\mathbb{C}^n\to\mathbb{C}$) avec $N>n$ et $k>0$, trouver $p_1,\cdots,p_N$ dans un sous-espace vectoriel de dimension finie de l'espace vectoriel $\mathcal{P}$ des applications analytiques de $\mathbb{C}^n\longrightarrow\mathbb{C}$ et des points $\boldsymbol{z}_1,\cdots,\boldsymbol{z}_k\in\mathbb{C}^n$ tels que $\boldsymbol{z}_i\in\mathcal{Z}_{\mathbb{C}^n}(f_1-p_1,\cdots,f_N-p_N)$, pour tout $i\in\{1,\cdots,k\}$ et $\|f_1-p_1\|_2+\cdots+\|f_N-p_N\|$ soit minimal.

Ce problème avait été considéré par Karmakar et Lakshman dans [33] pour le cas $n=1$. Nous généralisons la formule de Karmakar et Lakshman dans le cas de plusieurs variables, puis nous généralisons l'itération de Weierstrass de la section précédente en utilisant l'itération de Gauss-Newton en lieu et place de l'itération de Newton sur un système généralisant celui des relations racines-coefficients. Pour avoir une bonne vue sur les travaux reliés aux PGCD approchés, je renvoie à ([46] section 1.1.).

Si $I\in\mathbb{N}$, je noterai $\mathbb{C}[z]_I$ l'ensemble des polynômes à support dans $\{z^i|i\in I\}$.

Interpolation de Lagrange généralisée

On commence par généraliser un peu l'interpolation de Lagrange. En toute rigueur, il s'agit plus d'une méthode d'approximation qu'une méthode d'interpolation. On cherche à approcher une déformation qui permet d'imposer des zéros.

Problème 3.2. Soit $z_1,\cdots,z_k$ des points complexes distincts et $f_1,\cdots,f_k$ des valeurs complexes quelconques, on considère $I\subset\mathbb{N}$ tel que $|I|\geqslant k$. Le problème d'interpolation de Lagrange généralisée consiste à trouver $F\in\mathbb{C}[z]_I$ le polynôme à support dans I tel que $F(z_i)=f_i$ pour $i\in\{1,\cdots,k\}$ et minimisant la la norme 2.

Pour étudier ce problème, nous aurons besoin de quelques définitions :

Définition 3.1. *Soient $k\in\mathbb{N}$ et $I=\{i_1,\cdots,i_p\}\subset\mathbb{N}$ tels que $p\geqslant k$.*

- *Soit $\boldsymbol{z}=(z_1,\cdots,z_k)\in\mathbb{C}^k$, on définit la matrice de Vandermonde généralisée associée à $\boldsymbol{z}$ et I comme la matrice $k\times p$ suivante :*

$$V_I(\boldsymbol{z}):=\begin{pmatrix} z_1^{i_1} & \cdots & z_1^{i_p}\\ \vdots & & \vdots\\ z_k^{i_1} & \cdots & z_k^{i_p}\end{pmatrix}.$$

- *Pour $\boldsymbol{z}=(z_1,\cdots,z_k)\in\mathbb{C}^k$ on définit la matrice $k\times k$ $M_I(\boldsymbol{z})$ suivante :*

$$M_I(\boldsymbol{z}):=\left(\sum_{i\in I}(z_s\cdot\bar{z}_t)^i\right)_{s\ et\ t\in\{1,\cdots,k\}}$$

  qui satisfait $V_I(\boldsymbol{z})^*\cdot V_I(\boldsymbol{z})$.

- *Pour $I\subset\mathbb{N}$, on définit $\mathcal{R}_I:=\{\boldsymbol{z}=(z_1,\cdots,z_k)\in\mathbb{C}^k|\mathrm{rank}(M_I(\boldsymbol{z}))=k\}$. Pour I et $J\subset\mathbb{N}$, on définit $\mathcal{R}_{I,J}:=\mathcal{R}_I\cap\mathcal{R}_J$.*

- *Pour I et $J\subset\mathbb{N}$ et f et $g\in\mathbb{C}[z]$, on définit l'ensemble :*
  $\Omega_{I,J,k}(f,g):=\{(u,v)\in\mathbb{C}[z]^2|\exists\boldsymbol{z}\in\mathcal{R}_{I,J}$ pour lequel $\forall i\in\{1,\cdots,k\},u(z_i)=v(z_i)=0$ *et* $f-u\in\mathbb{C}[z]_I$ *et* $g-v\in\mathbb{C}[z]_J\}$.

De façon informelle, $\Omega_{I,J,k}(f,g)$ est l'ensemble des paires de polynômes ayant au moins k racines commune et obtenues de (f,g) en perturbant f sur les coefficients des termes à support dans I et g à support dans J. On omettra le couple (f,g) dans $\Omega_{I,J,k}(f,g)$ quand le contexte le permet.

On peut maintenant définir la généralisation des polynômes de Lagrange qui nous intéressent.

DÉFINITION 3.2. *Soit $I=\{i_1,\cdots,i_p\}$ et $\boldsymbol{z}\in\mathcal{R}_I$, on définit $z^I=(z^{i_1},\cdots,z^{i_p})$ et on note $\{\boldsymbol{e}_1,\cdots,\boldsymbol{e}_k\}\subset\mathbb{C}^k$ la base standard de $\mathbb{C}^k$. On définit la base de Lagrange généralisée associée au support I comme suit :*

$$L_{I,i}(\boldsymbol{z},z):=z_I\cdot V_I(\boldsymbol{z})^\dagger\cdot\boldsymbol{e}_i,\text{pour } i\in\{1,\cdots,k\}.$$

Remarquons que si $I=\{0,\cdots,k-1\}$ alors $\{L_I(\boldsymbol{z},z)|i\in\{1,\cdots,k\}\}$ est la base usuelle des polynômes de Lagrange.

La proposition suivante généralise l'interpolation de Lagrange usuelle en permettant de trouver le couple de polynômes de support I interpolant f et g sur les coordonnées de $\boldsymbol{z}$ (on suppose les coordonnées distinctes) et minimisant la norme 2 (une des plus petites déformations).

PROPOSITION 3.3. *Soient $k\in\mathbb{N}$, $I\subset\mathbb{N}$ de cardinal $p\geqslant k$, $\boldsymbol{z}=(z_1,\cdots,z_k)\in\mathcal{R}_I$, alors pour tout i et $j\in\{1,\cdots,k\}$ on a :*

$$L_{I,i}(\boldsymbol{z},z_j)=\delta_{i,j}.$$

PROPOSITION 3.4. *Soient $k\in\mathbb{N}$, $I\subset\mathbb{N}$ de cardinal $p\geqslant k$, $\boldsymbol{z}=(z_1,\cdots,z_k)\in\mathcal{R}_I$ et $\boldsymbol{f}=(f_1,\cdots,f_k)\in\mathbb{C}^k$ alors $F(z)=\sum_{i=1}^{k}f_i\cdot L_{I,i}(\boldsymbol{z},z)\in\mathbb{C}[z]_I$ et :*

$$F(z_j)=f_j,\forall i\in\{1,\cdots,k\}.$$

De plus :

$$\|F\|^2=\boldsymbol{f}^*\cdot M_I(\boldsymbol{z})^{-1}\cdot\boldsymbol{f}$$

est minimale parmi les polynômes à support dans I.

Finalement, on a :

THÉORÈME 3.5. *Soient f et $g\in\mathbb{C}[z]$, $I,J\subset\mathbb{N}$ et $\boldsymbol{z}\in\mathcal{R}_{I,J}$, on définit les deux polynômes suivants dans $\mathbb{C}[z]_I$ et $\mathbb{C}[z]_J$ respectivement :*

$$F_I(\boldsymbol{z},z)=\sum_{i=1}^{k}f(z_i)\cdot L_{I,i}(\boldsymbol{z},z)\ \text{ et }\ G_J(\boldsymbol{z},z)=\sum_{i=1}^{k}g(z_i)\cdot L_{J,i}(\boldsymbol{z},z).\tag{3.5}$$

Alors $(f(z)-F_I(\boldsymbol{z},z),g(z)-G_J(\boldsymbol{z},z))\in\Omega_{I,J,k}(f,g)$ et de plus si $\min_{\boldsymbol{z}\in\mathcal{R}_{I,J}}\{\boldsymbol{f}^*M_I(\boldsymbol{z})^{-1}\cdot\boldsymbol{f}+\boldsymbol{g}^*\cdot M_J(\boldsymbol{z})^{-1}\cdot\boldsymbol{g}\}$ existe et est atteint en $\boldsymbol{\zeta}\in\mathcal{R}_{I,J}$ alors on a :

$$\|F_I(\boldsymbol{\zeta},z)\|^2+\|G_J(\boldsymbol{\zeta},z)\|^2=\min_{(u,v)\in\Omega_{I,J,k}}\{\|f-u\|^2+\|g-v\|^2\}$$

avec $\boldsymbol{f}=(f(z_1),\cdots,f(z_k))$ et $\boldsymbol{g}=(g(z_1),\cdots,g(z_k))$.

Ce résultat montre le lien avec la formule de Karmarkar et Lakshman et montre que pour des déformations fixées on peut caractériser un PGCD approché comme un minimum s'il existe.

Grâce à cette « interpolation de Lagrange généralisée », nous allons construire un système analogue à celui reliant les coefficients d'un polynôme à ses racines et en déduire une itération de type itération de Weierstrass pour les PGCD approchés.

Généralisation de l'itération de Weierstrass pour les PGCD approchés multivariés

Dans ce paragraphe, je construis un système analogue à celui liant les racines d'un polynôme à ses coefficients pour les systèmes surcontraints. Contrairement au cas d'une seule équation, il n'y pas unicité des relations car les relations dépendent du support des déformations.

Définition 3.6. *Soient f et $g \in \mathbb{C}[z]$, $k \in \mathbb{N} \setminus \{0\}$ et I et $J \subset \mathbb{N}$ de cardinal plus grand que k. Pour $\boldsymbol{z} \in \mathcal{R}_{I,J}$, on note $F_I(\boldsymbol{z}, z) \in \mathbb{C}[z]_I$ et $G_J(\boldsymbol{z}, z) \in \mathbb{C}[z]_J$ les polynômes définis en 3.5, on définit alors l'application de Weierstrass généralisée comme suit :*

$$\mathcal{W}_{I,J} : \begin{cases} \mathbb{C}^k \longrightarrow \mathbb{C}[z]_I \times \mathbb{C}[z]_J \\ \mathbf{z} \longmapsto (F_I(\boldsymbol{z}, z), G_J(\boldsymbol{z}, z)) \end{cases}. \tag{3.6}$$

Le résultat suivant montre qu'un extremum de l'application que nous avons introduite coïncide avec une solution du problème d'optimisation proposé par Karmarkar et Lakshman :

Théorème 3.7. *Soient $\boldsymbol{z} = (z_1, \cdots, z_k) \in \mathbb{C}^k$, $(f, g) \in \mathbb{C}[z]^2$ et $\mathcal{W}_{I,J}$ telle que définie en 3.6. Alors :*

1. *$\mathcal{W}_{I,J}(\boldsymbol{z}) = 0$ si et seulement si $z_1, \cdots, z_k$ sont des racines communes à f et g.*
2. *Avec les notations du théorème 3.5, pour tout $\boldsymbol{z} \in \mathbb{C}^k$ nous avons*

$$\|\mathcal{W}_{I,J}(\boldsymbol{z})\|^2 = \boldsymbol{f}^* \cdot M_I \cdot \boldsymbol{f} + \boldsymbol{g}^* \cdot M_J \cdot \boldsymbol{g}.$$

3. $\min\limits_{\boldsymbol{z} \in \mathcal{R}_{I,J}} \|\mathcal{W}_{I,J}(\boldsymbol{z})\| = \min\limits_{(u,v) \in \Omega_{I,J}} \{\|f - u\|^2 + \|g - v\|^2\}$.

Nous pouvons maintenant appliquer la méthode de Newton-Gauss à $\mathcal{W}_{I,J}$ ce qui donnera une méthode itérative pour l'approximation d'un PGCD approché (par l'approximation des racines de ce PGCD à support de déformation fixée).

Itération de Weierstrass généralisée

En dépit du fait que si $|I| \neq k$ ou $|J| \neq k$ les coordonnées de $\mathcal{W}_{I,J}$ ne sont pas analytiques, il est possible de déduire de $\mathcal{W}_{I,J}(\boldsymbol{z}) = 0$ un système d'équations différentiables sur un ouvert dense de $\mathbb{C}^k$ (voir section 4 de [46]). Il est donc possible d'appliquer la méthode de Gauss-Newton pour avoir une itération de forme :

$$\boldsymbol{z} \longleftarrow \boldsymbol{z} - \mathcal{J}_{I,J}(\boldsymbol{z})^\dagger \cdot \mathcal{W}_{I,J}(\boldsymbol{z}). \tag{3.7}$$

Dans cette formule $\mathcal{J}_{I,J}(\boldsymbol{z})$ est le jacobien de $\mathcal{W}_{I,J}(z)$, ce qui nous permet finalement de donner explicitement une formule d'itération explicite :

Proposition 3.8. *Soient f et $g \in \mathbb{C}[z], k > 0$, I et $J \subset \mathbb{N}$ avec $|I|$ et $|J| \geqslant k$, pour $\boldsymbol{z} = (z_1, \cdots, z_k) \in \mathbb{C}^k$ tel que $z_i \neq z_j$ si $i \neq j$, en notant :*

$$f_{\boldsymbol{z}}(z) = f(z) - F_I(\boldsymbol{z}, z) \;\text{ et }\; g_{\boldsymbol{z}}(z) = g(z) - G_J(\boldsymbol{z}, z)$$

alors la formule 3.7 devient :

$$\boldsymbol{z} \longleftarrow \boldsymbol{z} - (D^*_{f_{\boldsymbol{z}}} \cdot M_I(\boldsymbol{z})^{-1} \cdot D_{f_{\boldsymbol{z}}} + D^*_{g_{\boldsymbol{z}}} \cdot M_J(\boldsymbol{z}) \cdot D_{g_{\boldsymbol{z}}})^{-1} \cdot (D^*_{f_{\boldsymbol{z}}} \cdot M_I(\boldsymbol{z}) \cdot \boldsymbol{f} + D^*_{g_{\boldsymbol{z}}} \cdot M_J(\boldsymbol{z}) \cdot \boldsymbol{g})$$

où :

$$D_{f_{\boldsymbol{z}}} = \operatorname{diag}(f'_{\boldsymbol{z}}(z_i))_{i=1\cdots k} \;\text{ et }\; D_{g_{\boldsymbol{z}}} = \operatorname{diag}(g'_{\boldsymbol{z}}(z_i))_{i=1\cdots k} \in \mathbb{C}^{k \times k}. \tag{3.8}$$

Dans le cas où $|I| = |J| = k$, c'est-à-dire quand on déforme exactement le même nombre de coefficients que le degré du PGCD cherché, nous avons pu donner des formules encore plus simples et montrer la convergence locale.

Généralisation au cas multivarié

Dans [46], nous généralisons toutes les situations précédentes dans le cas de polynômes de plusieurs variables. La description des résultats est plus technique et implique un ensemble important de notations mais l'idée fondamentale est la même. On fixe un support de déformation qui doit donner suffisamment de paramètres, c'est-à-dire plus que le nombre d'éléments dans le support du diviseur cherché. À noter que la méthode est naturellement creuse au sens algébrique du terme puisqu'on travaille sur les supports et pas uniquement sur des polynômes denses à degré fixé. Cela entraîne d'autres difficultés car on doit considérer les support des déformations et le support du diviseur commun, mais les outils de preuve sont essentiellement les mêmes. Nous avons alors pu étudier des variations autour de cette itération dans les cas où les dimensions des supports de déformation et du support du diviseur ont le même nombre d'éléments. Dans ce cas, on peut interpréter (comme je l'avais fait durant ma thèse) l'itération de Weierstrass comme un champ de vecteurs et chercher à calculer un point fixe qui correspondra au polynôme cherché. Nous proposons des méthodes basées sur les champs de gradients de l'application de Weierstrass et une itération quadratique et nous détaillons la complexité de l'itération de chacune des approches.

Liens avec les autres sujets de recherche

L'itération de Weierstrass est une méthode locale puisque c'est une forme de l'itération de Newton. Cette itération est utilisée comme opérateur de correction dans des algorithmes de prédiction-correction ou de façon équivalente comme correcteur local dans des méthodes par homotopie (ou par continuation). Ces méthodes ont l'avantage d'être numériques mais ont l'inconvénient d'avoir des complexités difficiles à appréhender. Pour étudier le comportement numérique et la complexité de ce type d'approche, il existe de nombreux travaux sur les liens entre complexité globale (nombre de pas dans un algorithme d'homotopie) et distance aux lieux des problèmes mal conditionnés ou singuliers. C'est le sens de la métrique du conditionnement : plus le chemin est près du lieu singulier, plus la taille des pas devra être petite et donc plus il faudra de faire de pas. Il y a donc un compromis à trouver entre taille des pas et longueur du chemin pour donner la complexité d'une méthode d'homotopie. C'est pour cela que la métrique du conditionnement a été introduite. C'est ce qui explique mon intérêt pour le calcul de géodésique pour la métrique du conditionnement.

3.2. Courbes de Bézier et contours libres

Les courbes admettant des paramétrisations polynomiales ont diverses applications très concrètes, comme la Conception Géométrique Assistée par Ordinateur, jusqu'à des applications plus théoriques, comme les éléments finis. La base de Bernstein permet de représenter les courbes de Bézier à partir de points particuliers : les sommets des polygones de contrôle. Les polynômes de Bernstein de degré d formant une base de l'espace vectoriel des polynômes de degré inférieur ou égal à d, toute courbe admettant une représentation polynomiale admet une représentation comme courbe de Bézier. Les résultats présentés aux début ne sont pas originaux et il existe une littérature très large sur le sujet. Je ne cherche pas du tout l'exhaustivité et je ne présenterai que les résultats permettant de comprendre la démarche proposée ici. Pour une vue d'ensemble sur le sujet, je renvoie à [21].

Courbes de Bézier : présentation synthétique

Soit E un espace affine et $P_0, \cdots, P_D \in E$, on considère alors la liste ordonnée $[P_0, \cdots, P_D]$ et on définit la paramétrisation de Bézier associée $B([P_0, \cdots, P_D], t) = (1-t) \cdot B([P_0, \cdots, P_{D-1}]) + t \cdot B([P_1, \cdots, P_D], t)$ sachant que si $P \in E$, on a $B([P], t) = P$. Cette décomposition récursive est le schéma d'évaluation de De Casteljau et il n'est pas difficile de montrer que $B([P_0, \cdots, P_D], t) = \sum_{i=0}^{D} P_i \cdot \boldsymbol{b}_{i,D}(t)$ où $\boldsymbol{b}_{i,D}(t) = \binom{i}{D} \cdot (1-t)^{D-i} \cdot t^i$ est le $i^{\text{ème}}$ polynôme de Bernstein de degré D. Comme les polynômes de Bernstein de degré D forment une base de l'espace vectoriel des polynômes de degré D, toute courbe ayant une paramétrisation polynomiale admet une représentation comme courbe de Bézier.

On considère $B([P_0, \cdots, P_D], t)$ pour des valeurs du paramètre t prenant des valeurs réelles entre 0 et 1 incluses car cela permet de relier certaines propriétés de la courbe à celle de la liste de points de $[P_0, \cdots, P_D]$. La représentation en paramétrisation de Bézier présente de nombreux avantages du point de vue algorithmique. Par exemple, si je note $\text{Conv}([P_0, \cdots, P_D])$ l'enveloppe convexe de l'ensemble des points $P_0, \cdots, P_D$, alors, pour tout $t \in [0,1]$, on a $B([P_0, \cdots, P_D], t) \in \text{Conv}([P_0, \cdots, P_D])$. De plus, on sait que $B([P_0, \cdots, P_D], 0) = P_0$ et $B([P_0, \cdots, P_D], 1) = P_D$ si bien que $B([P_0, \cdots, P_D], t)$ est un chemin joignant P_0 à P_D mais on a également $\dot{B}([P_0, \cdots, P_D], 0) = D \cdot \overrightarrow{P_1 P_0}$ et $\dot{B}([P_0, \cdots, P_D], 1) = D \cdot \overrightarrow{P_{D-1} P_D}$. L'algorithme qui nous a été le plus utile est l'interpolation. Soit $M_0, \cdots, M_D \in E$ et $t_0, \cdots, t_D$ une subdivision de $[0,1]$, si on cherche une paramétrisation Γ telle que $\Gamma(t_i) = M_i$, alors il existe une unique paramétrisation de degré D donnée par $B([P_0, \cdots, P_D], t)$ où les points P_i sont obtenus en résolvant le système linéaire suivant :

$$\begin{pmatrix} \boldsymbol{b}_{0,D}(t_0) & \cdots & \boldsymbol{b}_{D,D}(t_0) \\ \vdots & \ddots & \vdots \\ \boldsymbol{b}_{0,D}(t_D) & \cdots & \boldsymbol{b}_{D,D}(t_D) \end{pmatrix} \cdot \begin{pmatrix} P_0^T \\ \vdots \\ P_D^T \end{pmatrix} = \begin{pmatrix} M_0^T \\ \vdots \\ M_D^T \end{pmatrix}. \tag{3.9}$$

Beaucoup de problèmes (découpage ou élévation de degré) peuvent être reformulés en termes d'évaluation interpolation sans dégrader les propriétés numériques mais avec une présentation uniforme et des algorithmes faciles à implanter.

Si $\boldsymbol{t} = t_0 < \cdots < t_D$ est une subdivision de $[0,1]$, je noterai $\boldsymbol{B}_D(\boldsymbol{t})$ la matrice $\begin{pmatrix} \boldsymbol{b}_{0,D}(t_0) & \cdots & \boldsymbol{b}_{D,D}(t_0) \\ \vdots & \ddots & \vdots \\ \boldsymbol{b}_{0,D}(t_D) & \cdots & \boldsymbol{b}_{D,D}(t_D) \end{pmatrix}$ et si $t_i = \frac{i}{D}$ pour tout $i \in \{0, \cdots, D\}$.

Déformation de courbes

Un des avantage de l'interpolation pour calculer les coefficients d'une représentation de Bézier est la linéarité. Soient $M_0, \cdots, M_D \in E$ et $\vec{v}_0, \cdots, \vec{v}_D \in \vec{E}$ des vecteurs de l'espace vectoriel sous-jacent, $\boldsymbol{t} = t_0 < \cdots < t_D$ une subdivision de $[0,1]$, si on connaît une paramétrisation $\Gamma\colon [0,1] \longrightarrow E$ telle que $\Gamma(t_i) = M_i$ pour tout $i \in \{0, \cdots, D\}$ et qu'on cherche $\Theta\colon [0,1] \longrightarrow E$ telle que $\Theta(t_i) = M_i + \vec{v}_i$ pour tout $i \in \{0, \cdots, D\}$, alors en considérant :

$$\begin{pmatrix} K_0^T \\ \vdots \\ K_D^T \end{pmatrix} = \boldsymbol{B}_D^{-1} \cdot \begin{pmatrix} \vec{v}_1^T \\ \vdots \\ \vec{v}_D^T \end{pmatrix} \tag{3.10}$$

on a $\Theta(t) = \Gamma(t) + B([K_0, \cdots, K_D], t)$. De plus, si on travaille sur les paramétrisations polynomiales de degré au plus D, cette solution est unique. Ainsi si, $\Gamma(t) = B([P_0, \cdots, P_D], t)$ alors $\Theta(t) = B([P_0 + K_0, \cdots, P_D + K_D], t)$.

C'est ce résultat qui a motivé l'utilisation des courbes de Bézier pour l'optimisation de formes.

Courbes de Bézier par morceaux

Pour gagner des degrés de liberté pour décrire des courbes, il existe deux approches principales : augmenter le degré des paramétrisations considérées ou découper la courbe en morceaux. La seconde solution est celle que nous avons retenue car les degrés élevés introduisent beaucoup de problèmes numériques mais aussi de « globalisation » des erreurs.

Considérons $\boldsymbol{P}_1 = [P_{1,0}, \cdots, P_{1,D_1}]$ et $\boldsymbol{P}_2 = [P_{2,0}, \cdots, P_{2,D_2}]$ deux polygones de contrôle tels que $P_{1,D} = P_{2,0}$. On définit alors $B([\boldsymbol{P}_1, \boldsymbol{P}_2], t) = \begin{cases} B([P_{1,0}, \cdots, P_{1,D_1}], 2 \cdot t) \text{ si } 0 \leqslant t \leqslant \frac{1}{2} \\ B([P_{2,0}, \cdots, P_{2,D_2}], 2 \cdot t - 1) \text{ si } \frac{1}{2} < t \leqslant 1 \end{cases}$ qui est une courbe de Bézier à deux « patchs » de degré respectivement D_1 et D_2.

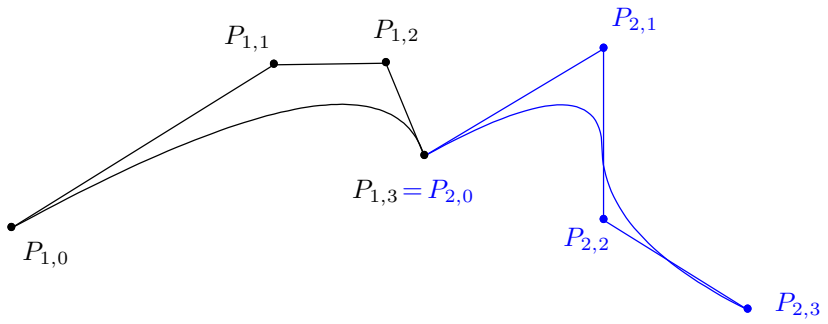

Figure 3.1. Courbes de Bézier à deux patchs homogènes de degré 3

Je ne considérerai ici que des courbes de Bézier par morceaux homogènes, ce qui signifie que tous les patchs ont le même degré et ce degré sera généralement 3. Les courbes de Bézier de degré 3 offrent toute la richesse géométrique locale possible avec très peu de paramètres. Elles réalisent les points lisses, d'inflexion et de rebroussement et même des points doubles bien que ces derniers (et tous les points multiples) ne soient pas intéressants pour les applications traitées ici.

Si pour tout $i \in \{1,...,N\}, \boldsymbol{P}_i = [P_{i,0}, \cdots, P_{i,D}]$ est un polygone de contrôle et si $P_{i,D} = P_{i+1,0}$ pour tout $i \in \{1, \cdots, N-1\}$, on définit $B([\boldsymbol{P}_1, \cdots, \boldsymbol{P}_N], t) = B([P_{i,0}, \cdots, P_{i,D}], N \cdot t - i)$ si $t \in \left[\frac{i}{N}, \frac{j+1}{N}\right]$ qui sera une courbe de Bézier à N patchs homogène de degré D. Ainsi, je noterai $B_{N,D}$ l'ensemble des courbes de Bézier par morceaux à N patchs de degré D et je m'intéresserai donc particulièrement à $B_{N,3}$.

Algorithmes pour les courbes de Bézier par morceaux

L'utilisation de courbes de Bézier par morceaux permet d'avoir des algorithmes adaptatifs. Je présente maintenant deux algorithmes pour manipuler des courbes de Bézier par morceaux : l'algorithme de raffinement « split » et l'algorithme de réarrangement « flip ». Ces algorithmes sont fondamentaux pour les applications d'évolution de courbes, comme l'évolution de contours pour l'optimisation de formes. Pour les trajectoires optimales, les applications que nous considérons sont très consommatrices de « split » mais pas de « flip » car il ne peut y avoir de boucle.

L'algorithme de raffinement est un algorithme très classique qui a plusieurs présentations. Je propose ici de ne réutiliser que les notions déjà présentées. Il s'agit de décomposer un patch en deux patchs réalisant exactement la même courbe.

Algorithme [Split]

Entrée : $[P_0, \cdots, P_D]$

1. Pour i allant de 0 à D :
   $Q_{1,i} \leftarrow B\Big([P_0, \cdots, P_D], \frac{i}{2 \cdot D}\Big)$
   $Q_{2,i} \leftarrow B\Big([P_0, \cdots, P_D], \frac{1}{2} + \frac{i}{2 \cdot D}\Big)$
2. $\boldsymbol{P}_1 = [P_{1,0}, \cdots, P_{1,D}] \leftarrow [Q_{1,0}, \cdots, Q_{1,D}] \cdot (\boldsymbol{B}_D^{-1})^T$
   $\boldsymbol{P}_2 = [P_{2,0}, \cdots, P_{2,D}] \leftarrow [Q_{2,0}, \cdots, Q_{2,D}] \cdot (\boldsymbol{B}_D^{-1})^T$
3. Renvoyer$([\boldsymbol{P}_1, \boldsymbol{P}_2])$

Les courbes $B([\boldsymbol{P}_1, \boldsymbol{P}_2], [0,1])$ et $B([P_0, \cdots, P_D], [0,1])$ sont exactement les mêmes pour des raisons de degré des paramétrisations, sauf que maintenant on dispose de $2 \cdot D + 1$ paramètres au lieu de $D + 1$ initialement. Cet algorithme permet d'avoir une approche adaptative dans les algorithmes de déformation, comme nous le verrons plus tard.

Le deuxième algorithme, le « flip », nous a permis de faire de l'optimisation topologique de formes dans [10] comme avec le gradient topologique ou les lignes de niveau. C'est un algorithme directement inspiré de méthodes de géométrie algorithmique sur les triangulations. Avant de donner l'algorithme en lui-même, je présente les motivations qui ont conduit à utiliser ce type d'algorithme dans ce contexte.

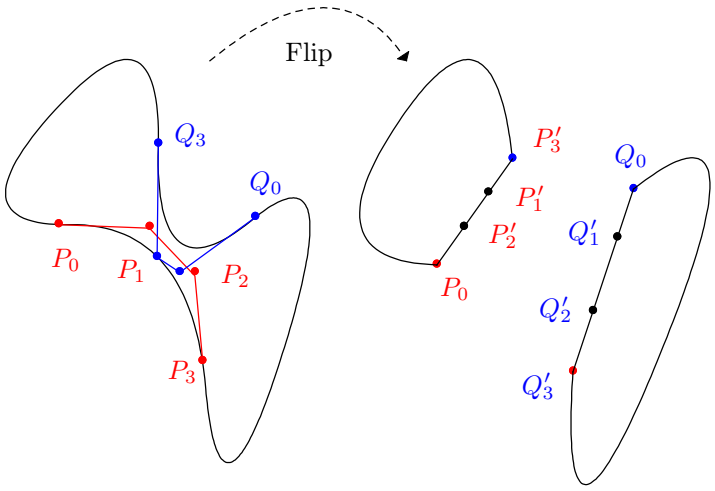

Figure 3.2. Application de l'algorithme de flip

L'idée est de détecter et de supprimer des auto-intersections apparaissant par exemple lors d'une évolution de la courbe. Comme l'idée provient d'un algorithme de géométrie algorithmique, comme d'habitude avec ce type d'algorithme, on peut passer l'essentiel du temps à gérer des événements qui ne peuvent pas ou que peu se produire. Premièrement, au lieu de tester l'intersection des patchs, on teste l'intersection des polygones de contrôle. C'est un filtre (au sens de la géométrie algorithmique) efficace. Tester l'intersection entre courbes est une opération complexe, il est souvent préférable de faire le « flip » si les polygones de contrôle se coupent et de voir si ça améliore la situation. Sur la figure 3.2, seuls deux patchs sont impliqués. Dans les algorithmes d'évolution de contours, les patchs sont généralement gardés de tailles comparables (au moins localement). Si bien que d'une point de vue heuristique, il ne peut y avoir que des intersections d'un patch avec un autre patch ou d'un patch avec deux autres patchs. Je ne donnerai ici l'algorithme que pour l'intersection de deux patchs et je renvoie à la thèse de Pierre Bonnelie [9] pour l'autre cas.

Algorithme Flip 1x1

Input : $\boldsymbol{P} = [P_0, \cdots, P_D]$ et $\boldsymbol{Q} = [Q_0, \cdots, Q_D]$

Output : $\boldsymbol{P'} = \left[P_0, P_0 + \frac{1}{3} \cdot \overrightarrow{P_0 Q_3}, P_0 + \frac{2}{3} \cdot \overrightarrow{P_0 Q_3}, Q_3\right]$ et $\boldsymbol{Q'} = \left[Q_0, Q_0 + \frac{1}{3}\overrightarrow{Q_0 P_3}, Q_0 + \frac{2}{3}\overrightarrow{Q_0 P_3}, P_3\right]$

Il y a plusieurs choses à remarquer. Cet algorithme n'est pas général. Il s'agit de se placer dans le contexte des déformations de contours et donc on suppose que les déformations sont petites devant la taille des polygones de contrôle et l'orientation est fixée par le parcours des polygones de contrôle si bien que les droites construites ne peuvent pas se croiser.

Interpolation et déformations

La linéarité de l'interpolation permet de déduire par interpolation la déformation induite du déplacement de points sur la courbe sur les sommets du polygone de contrôle. Encore une fois, je me limite au cas des cubiques et dans le plan mais les résultats sont bien plus généraux.

Je note ici $t_0=0, t_1=\frac{1}{3}, t_2=\frac{2}{3}$ et $t_3=1$. On note $b_{i,3}(t)=\binom{3}{i}\cdot t^i\cdot(1-t)^{3-i}$ pour $i\in\{0,1,2,3\}$ qui forme la base de Bernstein des polynômes de degré inférieur ou égal à 3. Un résultat classique exprime les courbes de Bézier cubiques dans la base de Bernstein de degré 3 :

PROPOSITION 3.9. *Soit P_0, P_1, P_2 et $P_3\in\mathbb{R}^2$, alors $B([P_0,P_1,P_2,P_3],t)=\sum_{i=0}^{3} P_i\cdot b_{i,3}(t)$.*

Je donne maintenant l'analogue de la matrice de Vandermonde dans la base de Bernstein :

DÉFINITION 3.10. *On appelle matrice de Bernstein-Vandermonde de degré 3 la matrice suivante :*

$$V=\begin{pmatrix} b_{0,3}(t_0) & b_{1,3}(t_0) & b_{2,3}(t_0) & b_{3,3}(t_0)\\ b_{0,3}(t_1) & b_{1,3}(t_1) & b_{2,3}(t_1) & b_{3,3}(t_1)\\ b_{0,3}(t_2) & b_{1,3}(t_2) & b_{2,3}(t_2) & b_{3,3}(t_2)\\ b_{0,3}(t_3) & b_{1,3}(t_3) & b_{2,3}(t_3) & b_{3,3}(t_3)\end{pmatrix}. \tag{3.11}$$

On a alors une formule d'interpolation :

PROPOSITION 3.11. *Soient M_0, M_1, M_2 et $M_3\in\mathbb{R}^2$, alors la seule courbe de degré inférieur ou égal à 3 telle que $\Gamma\left(\frac{i}{3}\right)=M_i$ pour $i\in\{0,1,2,3\}$ est donnée par $F(t)=B([P_0,P_1,P_2,P_3],t)$ où*

$$(\ P_0,P_1,P_2,P_3\ )^T=V^{-1}(M_0,M_1,M_2,M_3)^T. \tag{3.12}$$

De la même façon, si on veut maintenant perturber chacun des M_i d'un vecteur $\overrightarrow{\delta M_i}$ pour $i\in\{0,1,2,3\}$ de façon à trouver une courbe $\tilde{\Gamma}$ telle que $\tilde{\Gamma}(t_i)=M_i+\overrightarrow{\delta M_i}$, alors on dispose de la proposition suivante :

PROPOSITION 3.12. *La courbe de paramétrisation $\tilde{\Gamma}(t)=B([P_0+\overrightarrow{\delta P_0}, P_1+\overrightarrow{\delta P_1}, P_2+\overrightarrow{\delta P_2}, P_3+\overrightarrow{\delta P_3}],t)$ où $(\overrightarrow{\delta P_0},\overrightarrow{\delta P_1},\overrightarrow{\delta P_2},\overrightarrow{\delta P_3})=V^{-1}(\overrightarrow{\delta M_0},\overrightarrow{\delta M_1},\overrightarrow{\delta M_2},\overrightarrow{\delta M_3})$ est la seule courbe de degré inférieur ou égal à 3 tel que $\tilde{\Gamma}(t_i)=M_i+\overrightarrow{\delta M_i}$ pour $i\in\{0,1,2,3\}$.*

Je renvoie à [45] pour une une interprétation géométrique ayant pour cons équence de prouver cette proposition par la linéarité de l'interpolation en interprétant les perturbations comme des « coordonnées » d'un vecteur tangent.

3.3. Applications à l'optimisation de formes

Dans cette section, je vais très brièvement expliquer comment j'ai utilisé les courbes de Bézier par morceaux pour l'optimisation de formes. Je vais donc d'abord décrire le modèle que j'ai utilisé pour l'optimisation de formes puis comment les courbes de Bézier par morceaux s'appliquent à ce modèle. Les publications concernant cette partie sont la thèse de Pierre Bonnelie [9], un article montrant la pertinence de notre approche pour approcher plusieurs composantes grâce à l'algorithme de « flip » [10], l'article tiré du post-doctorat de Satafa Sanogo [11], application à la segmentation d'images catadioptriques dans la thèse de Pauline Merveilleux-Orzekowska [38] et les travaux issus de la thèse d'Ali Dia [14], [15], [16] et [17].

Optimisation de formes

Je me place dans le cadre de formes planes, c'est à dire d'objets géométriques plans codés par leurs frontières ω. Par exemple, la recherche de la surface maximisant l'aire à périmètre fixé (problème isopérimétrique) est le prototype de problème d'optimisation de formes. Je présente maintenant un modèle abstrait permettant de décrire la plupart des problèmes d'optimisation de forme. On note Ω un espace de formes (c'est-à-dire un sous-ensemble de l'ensemble des parties de $\mathbb{R}^2$) qu'on considère comme « constructible ». On considère alors $F: \Omega \longrightarrow \mathbb{R}^+$ une fonction. Le problème d'optimisation de forme associée à Ω et F est de déterminer un $\omega^* \in \Omega$ tel que $F(\omega^*) \leqslant F(w)$ pour tout $\omega \in \Omega$.

Les espaces de formes que nous considérons sont des régions telles que la frontière $\partial\omega$ est une courbe (généralement on souhaite que la forme soit délimitée par une courbe ayant plusieurs composantes connexes sans autointersection bien que d'un point de vue théorique cela ne soit pas vraiment limitatif). Nous nous limiterons au cas des contours qui admettent des paramétrisations continues par morceaux.

Une courbe de Bézier par morceaux est une boucle si le premier point de contrôle du premier polygone de contrôle est confondu avec le dernier point du dernier point de contrôle. Par densité des polynômes dans l'ensemble des fonctions continues, nous chercherons à approcher les formes par des formes dont les contours sont des courbes de Bézier par morceaux.

L'intérêt de cette approche est multiple : généralement, le calcul de $F(\omega)$ n'est pas toujours explicite, il peut requérir de résoudre une équation aux dérivées partielles par exemple - c'est souvent le cas. Si le contour est représenté implicitement, il faudra arriver à donner une estimation de la variation du critère en fonction de paramètres géométriques de la représentation implicite. Globalement, cela revient souvent à enchaîner deux problèmes inverses pour chaque itération d'un algorithme de gradient, un pour le calcul du critère et de ses variations (analyse de sensibilité) et une étape de propagation de front construit à partir du gradient de forme issu de l'analyse de sensibilité (comme la méthode des lignes de niveau avec un problème d'Hamilton-Jacobi).

Je noterai $B^\circ_{l,D}$ l'ensemble des courbes de Bézier à l patchs homogènes de degré D fermées, c'est-à-dire l'ensemble des courbes de la forme $B([[P_{1,0}, \cdots, P_{1,D}], \cdots, [P_{l,0}, \cdots, P_{l,D}]], t)$ avec $P_{i,D} = P_{i+1,0}$ pour $i \in \{0, \cdots, l-1\}$ et $P_{l,D} = P_{1,0}$. Clairement, grâce à l'algorithme de « split » il y a l façons de plonger $B^\circ_{l,D}$ dans $B^\circ_{l+1,D}$. Une première approche est de chercher $\gamma_l^\times \in B^\circ_{l,D}$ telle que $F(\gamma_l^\times) \leqslant F(\gamma)$ pour tout $\gamma \in B^\circ_{l,D}$. Autrement dit, on restreint la recherche d'un minimum à $B^\circ_{l,D}$. On peut alors « raffiner ». On peut le faire de façon systématique en utilisant l'algorithme « split » sur chacun des patchs de $\gamma_l^\times$ pour avoir son encodage dans $B^\circ_{2\cdot l,D}$ et essayer d'améliorer le critère à partir de $\gamma_l^\times$ pour obtenir $\gamma_{2\cdot l}^\times$ dans $B^\circ_{2\cdot l,D}$. Néanmoins le doublement du nombre de paramètres n'est pas toujours souhaitable. C'est pourquoi nous avons privilégié une approche adaptative. On utilise un critère local sur chaque patch pour savoir s'il est souhaitable de raffiner en décomposant le patch. Cette approche permet de maîtriser la complexité mais réclame des efforts pour proposer des techniques de géométrie algorithmique afin de conserver des performances et une cohérence des données.

Application à la segmentation d'images

J'ai présenté les principes de cette méthode permettant de passer de l'étape de propagation de front d'un problème implicite à un problème paramétrique dans [45]. J'ai pu tester cette approche pour trois applications différentes (quatre en réalité puisque pour moi, les problèmes de trajectoires optimales sont des problèmes d'optimisation de formes). Je les prends dans l'ordre où elles ont été traitées. Le premier est le cas de la vision catadioptrique [31] en collaboration avec Ouiddad Labbani-Igbida et Pauline Pauline Merveilleux-Orzekowska qui a constitué une partie de la thèse de cette dernière [38]. Dans cette application, un système de vision sert à un robot autonome à déterminer son « espace de travail », i.e. l'espace au sol qui lui est accessible. Le système de vision

catadioptrique permet d'avoir une vision à 360 degrés autour du robot. Comme représenté sur la figure 3.3, une caméra observe un miroir présentant une symétrie de révolution.

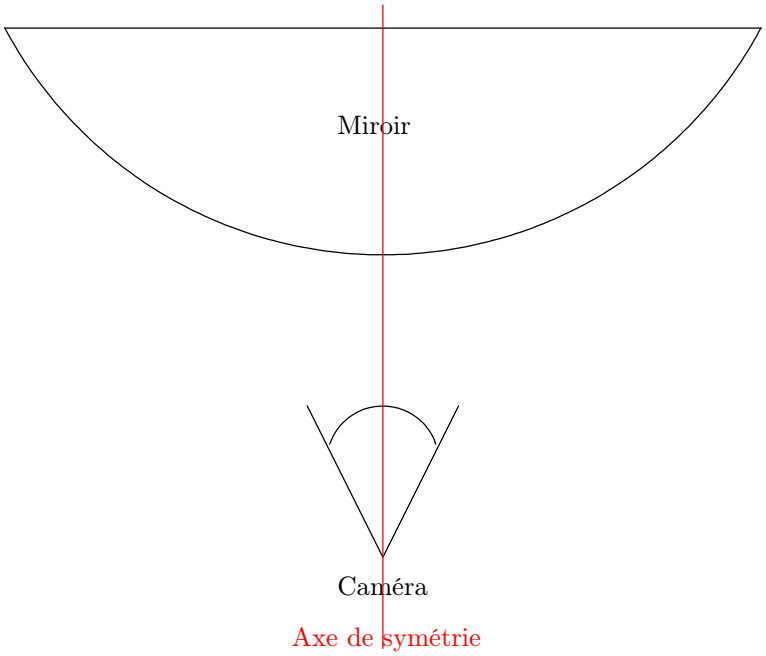

Figure 3.3. Schéma d'un dispositif de vision catadioptrique

Les images produites sont du type de la figure 3.4. Au milieu de l'image, se trouve le robot autonome. On commence par paramétrer une courbe circulaire autour du robot. On construit un champ de déformation suivant la méthode des contours actifs [34].

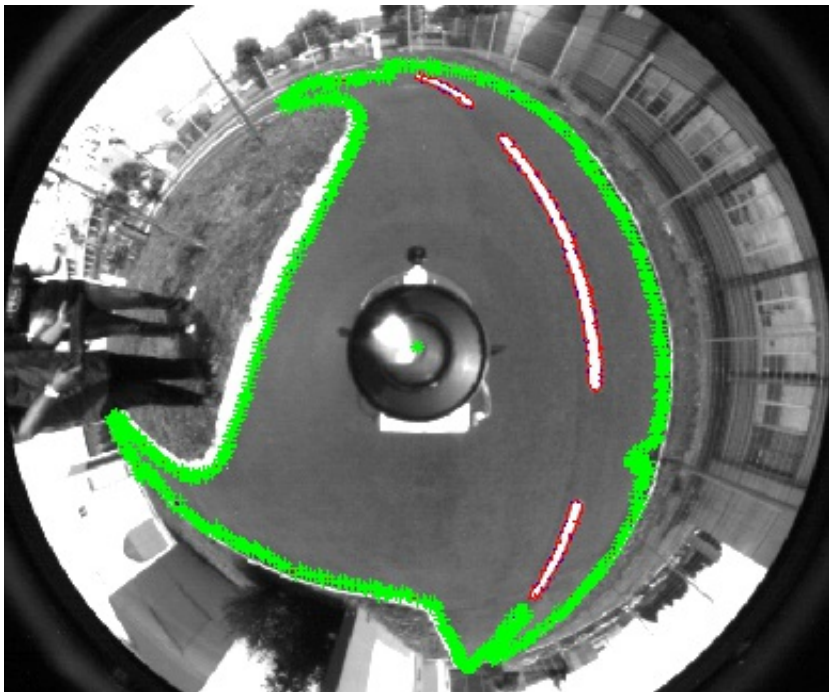

Figure 3.4. Segmentation d'une image catadioptrique

Cette approche a d'ailleurs été testée pour des images $3D$ dans un stage de Master 2 de Tran Duc Minh Phan en 2014 qui a fait un travail important qui n'a hélas pas encore été exploité. Le cas des surfaces est beaucoup plus difficile car l'algorithmique mise en jeu devient très complexe. Il a néanmoins réussi à faire des tests de détection de contours sur des images. Sur la figure 3.5, on cherche à détecter la forme décrite par les voxels noirs, on démarre d'une approximation de sphère par des surfaces de Bézier par morceaux (en bleu) et sur la figure 3.6, on a le résultat de la détection avec une extension de la méthode « snake » en adaptant l'algorithme de détection de Canny [20].

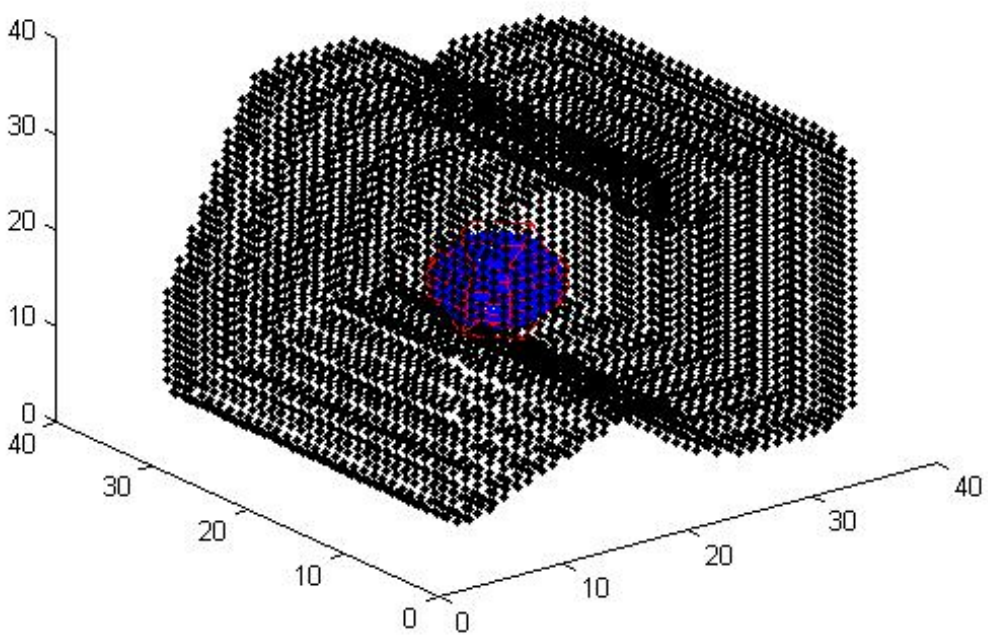

Figure 3.5. Forme à détecter en noir et forme de départ en bleu

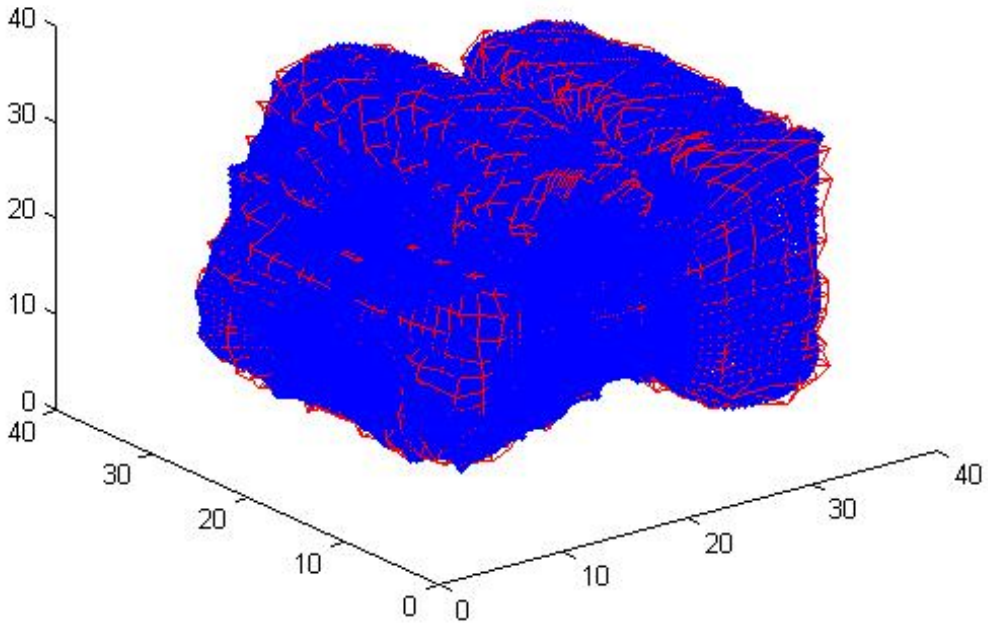

Figure 3.6. En bleu la forme finale et en rouge les polytopes de contrôle

Application à la détection d'inclusion

L'application que nous avons traitée ensuite concerne l'application de la méthode des contours libres à la détection d'inclusions ω_{ex} incluses dans un domaine compact $\Omega \subset \mathbb{R}^2$ à partir de mesures sur la frontière de Ω. Grâce à l'algorithme de « flip » 3.2, nous sommes parvenus avec Pierre Bonnelie, Loïc Bourdoin et Fabien Caubet à détecter des inclusions ayant deux composantes connexes (voir [10]). Le problème est bien plus mal conditionné que dans le cas de la segmentation d'images. Par exemple, on comprend bien qu'il est difficile de « voir » entre les inclusions à partir de mesures aux bords. Pour des raisons de simplicité et pour ne pas à avoir à introduire un ensemble trop large de notions d'analyse fonctionnelle dont je ne suis pas spécialiste, j'omettrai de préciser les esapces fonctionnels mis en jeu. Je noterai $\mathcal{O}$ l'ensemble des ouverts inclus dans Ω avec une fontière suffisamment régulière (disons au moins deux fois dérivables par morceaux) et soit $\omega_{\text{ex}} \in \mathcal{O}$ un ouvert de Ω à frontière régulière. On considère une équation de Laplace définie sur $\Omega \setminus \overline{w_{\text{ex}}}$ avec des conditions aux bords de type Dirichlet homogènes. On s'intéresse à une (unique) solution du

problème :

$$\begin{cases} -\Delta u_{\text{ex}} = 0 \text{ sur } \Omega \setminus \overline{\omega_{\text{ex}}}; \\ u_{\text{ex}} = g \text{ sur } \partial\Omega; \\ u_{\text{ex}} = 0 \text{ sur } \partial\omega_{\text{ex}}. \end{cases} \tag{3.13}$$

On cherche à reconstruire ω_{ex} à partir de g et de la connaissance de Ω. En fait, on suppose que l'on connaît exactement $f_b = \partial_{\boldsymbol{n}} u_{\text{ex}}$ sur $\partial\Omega$. Ainsi, pour toute paire de Cauchy (g, f_b) on s'intéresse au problème inverse suivant :

Problème 3.3. Trouver $\omega \in \mathcal{O}$ et u fonction suffisamment régulière satisfaisant le système surdéterminé suivant :

$$\begin{cases} -\Delta u = 0 \quad \text{sur } \Omega \setminus \overline{\omega}; \\ u = g \quad \text{sur } \partial\Omega; \\ \partial_{\boldsymbol{n}} u = f_b \quad \text{sur } \partial\Omega; \\ u = 0 \quad \text{sur} \quad \partial\omega. \end{cases} \tag{3.14}$$

L'existence d'une solution est assurée par le fait que f_b est supposée connue exactement (on a donc bien un élément de l'image de l'opérateur). Des résultats préalables ont montré le résultat suivant :

Théorème 3.13. *L'ouvert ω et la fonction u satisfaisant 3.14 sont uniquement déterminés par les données de Cauchy $(g, f_b) \neq (0, 0)$.*

Pour résoudre le le problème 3.3, on va considérer le problème d'optimisation de formes :

$$w^* = \underset{\omega \in \mathcal{O}}{\operatorname{argmin}} J(\omega) \tag{3.15}$$

où J est la fonctionnelle positive définie par le moindre-carré suivant :

$$J := \int_{\partial\Omega} |\partial_{\boldsymbol{n}} u_\omega - f_b|^2, \tag{3.16}$$

et u_ω est la solution de :

$$\begin{cases} -\Delta u_w = 0 \text{ sur} \quad \Omega \setminus \overline{\omega}, \\ u_\omega = g \quad \text{sur} \quad \partial\Omega, \\ u_\omega = 0 \text{ sur} \quad \partial\omega. \end{cases} \tag{3.17}$$

Le résultat d'identifiabilité nous garantit que $J(\omega) = 0$ si et seulement si $\omega = \omega_{\text{ex}}$. On utilise alors le théorème d'Hadamard pour calculer le gradient de forme :

Proposition 3.14. *Soient $\omega \in \mathcal{O}$ et V une déformation de w, alors J est différentiable en ω dans la direction V avec $DJ(\omega) \cdot V = \int_{\partial\omega} \partial_{\boldsymbol{n}} u_\omega - \partial_{\boldsymbol{n}} w_w (V \cdot \boldsymbol{n})$, où w_ω est l'unique solution du problème adjoint :*

$$\begin{cases} -\Delta w_\omega = 0 \text{ sur} \quad \Omega \setminus \overline{\omega}, \\ w_\omega = 2 \cdot (\partial_{\boldsymbol{n}} u_\omega - f_b) \text{ sur} \quad \partial\Omega, \\ w_\omega = 0 \quad \text{sur} \quad \partial\omega. \end{cases}$$

Cette approche nous a permis de tester la méthode à partir de simulations de modèles que nous avions construites avec deux inclusions. Même en démarrant avec une seule inclusion, notre algorithme reconstruit les deux inclusions, les déformations des frontières données par la proposition précédente étant remontées directement en déformations des points de contrôle. Il est à remarquer que la méthode est très efficace pour retrouver la géométrie de l'inclusion quand il n'y a qu'une seule inclusion mais que le conditionnement ne permet pas d'en faire autant dans le cas de plusieurs inclusions.

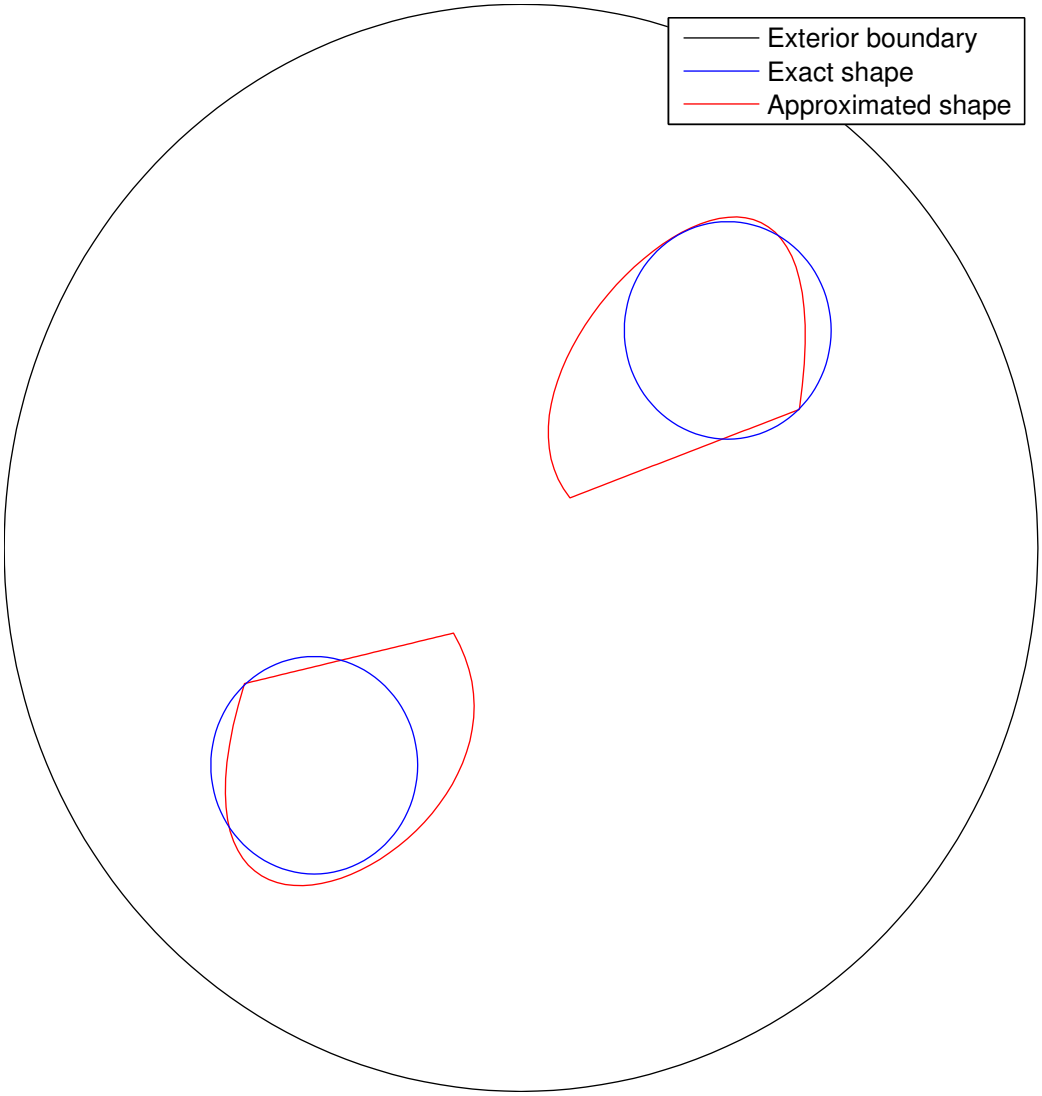

Figure 3.7. Détection d'inclusions dans le cas de deux inclusions

Application à la conception de composants électromagnétiques

Depuis plusieurs années, je collabore avec Stéphane Bila sur l'optimisation de composants électroniques et nous avons co-encadré Pierre Bonnélie (avec Paul Armand également) pour appliquer l'approche des contours libres pour la conception de composants électromagnétiques. Une fois la partie algorithmique avancée et grâce à une action de recherche avec le CNES, nous avons continué, ce qui a conduit à la thèse de Ali Dia qui est en train de s'achever (également co-encadrée par Christophe Durousseau et Cyrille Ménudier). Avec Stéphane Bila, nous avons également co-encadré Satafa Sanogo en post-doctorat, ce qui a conduit à la publication [11] et qui a précédé les travaux dans le cadre de la thèse d'Ali sur la conception de filtres [14], [15], [16], [17] et [18]. Une des grandes difficultés de ce type de sujet est la mutiplicité des outils logiciels utilisés et le fait que certains soient fermés (impossible d'accéder au code ou à des fonctionnalités de base).

3.4. TRAJECTOIRES OPTIMALES

Dans cette section, nous aborderons les problèmes de trajectoires optimales. Une trajectoire optimale est une courbe minimisant un critère, généralement de nature géométrique, c'est-à-dire que le critère dépend uniquement de la courbe et pas de la paramétrisation de celle-ci. C'est cet aspect que je vais commencer par formaliser. Je noterai E un espace affine et je considère une paramétrisation $\gamma: I \longrightarrow E$ d'un intervalle (qu'on supposera fermé) de I dans E de classe C^k. Je noterai $\mathcal{C} = \gamma(I)$ la courbe paramétrée par γ. Si $\varphi \in C^k([0,1], I)$ est une fonction ayant au moins la même régularité que γ et qui réalise un homéomorphisme de $[0,1]$ sur I, alors $\gamma \circ \varphi: [0,1] \longrightarrow E$ est aussi une paramétrisation de $\mathcal{C}$ de même régularité que γ. On peut donc se concentrer sur les paramétrisations définies sur $[0,1]$. On souhaite avoir des courbes simples et donc dont les paramétrisations sont des immersions. On note $\mathcal{E} = \mathrm{Emb}([0,1], E) = \{\gamma \in C^k([0,1], E) | \gamma(t) = \gamma(t') \Rightarrow t = t'\}$ et si on considère des courbes fermées, courbes de Jordan, on aura $\mathcal{F} = \{\gamma \in C^k([0,1], E) | \gamma(t) = \gamma(t') \Rightarrow (t = t' \text{ ou } (t = 0 \text{ et } t' = 1) \text{ ou } (t = 1 \text{ et } t' = 0))\}$. On note $\mathcal{D}^+$ l'ensemble des difféomorphismes croissants de classe C^k de $[0,1]$ dans $[0,1]$. Comme on parle de trajectoire, on souhaite conserver l'orientation, ce qui explique le choix de $\mathcal{D}^+$ au lieu de prendre l'ensemble des difféomorphismes de $[0,1]$ dans $[0,1]$. Beaucoup de résultats de cette section se généralisent assez facilement en considérant $\mathcal{D}$ au lieu de $\mathcal{D}^+$. Si $\mathcal{C} = \gamma([0,1])$ pour $\gamma \in \mathcal{E}$, alors, pour tout $\varphi \in \mathcal{D}^+$, $\mathcal{C} = \gamma \circ \varphi([0,1])$ et $\gamma(0) = \gamma \circ \varphi(0)$ et $\gamma(1) = \gamma \circ \varphi(1)$. Réciproquement, il n'est pas très difficile de montrer que si γ et $\delta \in \mathcal{E}$ sont tels que $\gamma(0) = \delta(0)$ et $\gamma(1) = \delta(1)$ et si de plus $\gamma([0,1]) = \delta([0,1])$ alors il existe $\varphi \in \mathcal{D}^+$ tel que $\delta = \gamma \circ \varphi$. C'est la principale motivation pour représenter l'ensemble des trajectoires dans E comme $\mathcal{T} = \mathcal{E} / \mathcal{D}^+$.

On définit maintenant ce qu'on entend par une fonctionnelle géométrique.

Définition 3.15. *Soit $F: \mathcal{E} \longrightarrow \mathbb{R}^+$, on dit que F est une fonctionnelle géométrique si sa valeur sur une paramétrisation ne dépend que de la courbe et pas de la paramétrisation choisie, i.e. $\forall \gamma \in \mathcal{E}$ et $\varphi \in \mathcal{D}^+$, on a $F(\gamma) = F(\gamma \circ \varphi)$.*

Les constructions précédentes sont faites de telle sorte que les fonctionnelles géométriques passent au quotient par les difféomorphismes croissants, c'est-à-dire que si F est une fonctionnelle géométrique alors elle définit une fonction qu'on notera encore $F: \mathcal{T} \longrightarrow \mathbb{R}^+$. Avec ces notations, je peux maintenant définir ce que j'entends comme problème de trajectoire optimale :

Problème 3.4. Soit F une fonctionnelle géométrique, le problème de trajectoire optimale minimisant associé à F consiste à trouver $\gamma \in \mathcal{T}$ tel que $F(\gamma) \leqslant F(\delta)$ pour tout $\delta \in \mathcal{T}$.

Pour traiter ce type de problèmes, on se propose d'approcher une solution d'un problème de trajectoire optimale à l'aide de courbes de Bézier ou de courbes de Bézier par morceaux. Voyons pourquoi les courbes de Bézier ou les courbes à paramétrisation polynomiale sont intéressantes pour ce problème. Le fait est qu'une courbe de Bézier admettant une paramétrisation de plus petit degré donné n'admet en général qu'une seule paramétrisation de ce degré puisque qu'il n'y a qu'un seul difféomorphisme polynomial de l'intervalle $[0,1]$ conservant le degré qui est l'identité. Bien sûr, si γ est une paramétrisation et φ un difféomorphisme polynomial de $[0,1]$ alors $\gamma \circ \varphi$ aussi. Mais le degré de $\gamma \circ \varphi$ est plus grand que celui de γ. Par exemple, $\begin{pmatrix} t \\ t^2 \end{pmatrix}$ admet aussi comme paramétrisation $\begin{pmatrix} t^3 \\ t^6 \end{pmatrix}$ en composant avec $t \mapsto t^3$.

Motivations du problème

La motivation initiale provient des méthodes d'homotopie pour la résolution des systèmes polynomiaux. Imaginons qu'on veuille résoudre un système algébrique $F(x_1, \cdots, x_n) = 0$ avec $F(x_1, \cdots, x_n) = (f_1(x_1, \cdots, x_n), \cdots, f_m(x_1, \cdots, x_n)) \in \mathbb{C}[x_1, \cdots, x_n]^m$, avec $m \geqslant n$ pour qu'il n'ait génériquement qu'un nombre fini de solutions. Imaginons que nous disposions d'un système $G(x_1, \cdots, x_n) = 0$ mais qu'on connaisse les solutions de ce système et que cet ensemble de solutions ait la même structure que celle de $F(x_1, \cdots, x_n) = 0$.

Formes normales de paramétrisations

Nous avons choisi de représenter les courbes par des classes d'équivalence de paramétrisation. Nous allons maintenant définir une projection permettant d'avoir un système de formes normales pour les paramétrisations modulo l'action à droite du groupe $\mathcal{D}^+$ sur l'ensemble des immersions. La plupart des étudiants ayant une licence de mathématiques sait qu'il existe une paramétrisation particulière qui se calcule à partir de n'importe quelle paramétrisation de la courbe (si la paramétrisation est une immersion). C'est la paramétrisation par longueur d'arc. On considère $\gamma \in \mathcal{E}$ une immersion paramétrant une courbe $\mathcal{C}$, alors la longueur de $L_{\mathcal{C}}$ de $\mathcal{C}$ est donnée par $L_{\mathcal{C}} = \int_0^1 \|\dot{\gamma}(t)\|_2 \, dt$ et la paramétrisation par longueur d'arc est calculée à l'aide du difféomorphisme $l_\gamma : [0,1] \longrightarrow [0, L_{\mathcal{C}}]$ définit par $l_\gamma(v) = \int_0^v \|\dot{\gamma}(t)\|_2 \, dt$ qui permet d'obtenir $l_\gamma^{-1} : [0, L_{\mathcal{C}}] \longrightarrow [0,1]$ et la paramétrisation par longueur d'arc $\gamma \circ l_\gamma^{-1} : [0, L_{\mathcal{C}}] \longrightarrow E$ telle que $\gamma \circ l_\gamma^{-1}([0, L_{\mathcal{C}}]) = \mathcal{C}$. Mais nous ne considérons que des paramétrisations définies sur $[0,1]$. Considérons $k_{\mathcal{C}} : t \in [0,1] \mapsto L_{\mathcal{C}} \cdot t \in [0, L_{\mathcal{C}}]$. Je définis alors $\mathcal{N}_\gamma = \gamma \circ l_\gamma^{-1} \circ k_{\mathcal{C}} : [0,1] \longrightarrow E$ qui est une paramétrisation de $\mathcal{C}$. De plus, nous avons montré que : γ et $\delta \in \mathcal{E}$ sont deux paramétrisations d'une même courbe $\mathcal{C}$ si et seulement si $\mathcal{N}_\gamma = \mathcal{N}_\delta$. De plus, pour tout $\gamma \in \mathcal{E}$, nous avons $\mathcal{N}_{\mathcal{N}_\gamma} = \mathcal{N}_\gamma$. Ainsi nous avons défini une projection $\mathcal{N}$ de $\mathcal{E}$ dans $\mathcal{E}$ qui permet de donner un système de formes normales et donc une représentation de $\mathcal{T}$ comme l'image de $\mathcal{E}$ par $\mathcal{N}$ en tant que sous-variété de $\mathcal{E}$. Nous avons alors étudié les différentes topologies sur $\mathcal{T}$. L'application $\mathcal{N}$ étant différentiable, nous avons alors étudié la structure de sous-variété différentielle de $\mathcal{T}$.

THÉORÈME 3.16. *L'application* $\mathcal{N} : \left\{ \begin{array}{l} \mathcal{E} \longrightarrow \mathcal{E} \\ \gamma \longmapsto \mathcal{N}_\gamma = \gamma \circ l_\gamma^{-1} \circ k_{\mathcal{C}} \end{array} \right.$ *est une projection sous l'action à droite sur* $\mathcal{E}$ *du groupe des difféomorphismes croissants de* $[0,1]$ *dans* $[0,1]$.

À ma connaissance, c'est la première description aussi complète d'un espace de formes avec une forme normale et la possibilité d'accéder à la structure différentielle explicitement. On trouve divers travaux dans le cas des courbes de Jordan (contours ou formes planes). Un problème important est de pouvoir définir une distance entre courbes. Nous verrons que définir la distance entre deux courbes à partir des formes normales est très efficace. Le premier effet que nous pouvons voir est une situation de factorisation universelle classique. En effet, si $F : \mathcal{E} \longrightarrow \mathbb{R}^+$ est une fonctionnelle géométrique, alors il existe une unique application $\tilde{F} : \mathcal{T} \longrightarrow \mathbb{R}^+$ telle que $F(\gamma) = \tilde{F}(\mathcal{N}_\gamma)$, autrement dit faisant commuter le diagramme suivant :

$$\begin{array}{ccc} \mathcal{E} & \xrightarrow[\mathcal{N}]{} & \mathcal{T} \\ \downarrow F & \swarrow \tilde{F} & \\ \mathbb{R}^+ & & \end{array} \tag{3.18}$$

Ainsi, cette capacité « théorique » d'avoir une forme normale sur les paramétrisations permet de donner un cadre rigoureux à la recherche de trajectoires optimales.

Distance entre courbes

Une des principales applications de la projection $\mathcal{N} : \mathcal{E} \longrightarrow \mathcal{T}$ est de permettre de définir une distance entre courbes. En effet, si $\mathcal{C}$ et $\mathcal{D}$ sont deux courbes dans E de paramétrisation respectivement γ et δ, on peut alors définir une distance $d^{\mathcal{N}}(\mathcal{C}, \mathcal{D}) = \int_0^1 \|\mathcal{N}_\gamma(t) - \mathcal{N}_\delta(t)\| \, dt$. L'indépendance de la paramétrisation choisie vient de l'utilisation des formes normales et on obtient bien une distance sur $\mathcal{T}$. Comme dit précédemment, avoir la capacité de définir une distance est un problème important qui est une des pierres d'achoppement de beaucoup de problèmes algorithmiques comme la segmentation, la reconnaissance de formes, etc. Beaucoup de propositions ont été faites (distance élastique [50], distance géodésique [52], voir [51] pour un point de vue plus complet) mais celle proposée ici semble canonique.

Dans la thèse de Hoang Van Duc [30], nous avons étudié la topologie sur $\mathcal{T}$ associée à cette distance et nous l'avons comparée à la topologie induite de celle de $\mathcal{E}$ puis à la topologie quotient. En fait la topologie associée à $d^{\mathcal{N}}$ est la plus fine que nous ayons étudiée.

Liens avec les courbes de Bézier par morceaux

Si on considère deux courbes de Bézier de même degré, alors elles ont la même forme normale si et seulement si elles sont égales. Il est facile de se convaincre du même résultat pour des courbes de Bézier par morceaux (homogènes) et c'est notamment plus facile pour les courbes de Bézier par morceaux homogènes de degré 3. Ainsi, une courbe de Bézier par morceaux peut également être prise comme représentante de sa propre classe d'équivalence. Donc, avec une fonctionnelle géométrique, on peut travailler directement dans les espaces $B_{N,D}$ sans avoir recours à la forme normale. Pour le calcul de distance, je renvoie à la thèse de Hoang Van Duc [30] puisque nous sommes en dimension finie et que toutes les normes sont équivalentes. Les courbes de Bézier par morceaux offrent donc un cadre particulièrement naturel pour étudier les problèmes de trajectoires optimales.

3.4.0.1. Exemples de problèmes de trajectoires optimales

Nous donnons ici des exemples de problèmes de trajectoires optimales. Il s'agit de problèmes directement formulés en termes de trajectoires. Nous verrons qu'il est parfois possible de se ramener à un tel problème dans des contextes plus généraux.

Géodésiques pour la métrique du conditionnement

Je considère Σ un sous-ensemble semi-algébrique de E (qui représentera un lieu qu'on cherche à éviter comme un lieu singulier à l'image de celui des matrices singulières dans un espace affine de matrices pour la résolution de systèmes linéaires). La métrique du conditionnement veut mesurer la distance à Σ en modifiant la façon de mesurer les longueurs. On souhaiterait que la distance augmente lorsqu'on se rapproche de Σ. L'idée est de pondérer les éléments de longueur par une fonction qui croît quand on se rapproche de Σ et pour la métrique du conditionnement, la mesure est $\frac{dx}{d(x,\Sigma)}$ où $d(x,\Sigma)$ est la distance x à Σ. Ainsi, la longueur d'une courbe γ devient $l_c(\gamma) = \int_0^1 \frac{\|\dot\gamma(t)\|}{d(\gamma(t),\Sigma)}\,dt$. Nous pouvons maintenant formuler le problème de calcul d'une géodésique pour la métrique du conditionnement :

Problème 3.5. Soit A et $B \in E$, trouver $\gamma \in \mathcal{E}$ telle que $\gamma(0) = A$, $\gamma(1) = B$ et pour tout $\delta \in \mathcal{E}$ telle que $\delta(0) = A$ et $\delta(1) = B$, on a $l_c(\gamma) \leqslant l_c(\delta)$.

Une solution du problème 3.5 est appelée arc de géodésique joignant A à B pour la métrique du conditionnement. Ce problème a déjà été étudié : je l'ai rencontré pour la première fois lors d'un exposé de Carlos Beltran. C'est un problème « modèle » car même si beaucoup de questions restent ouvertes, il y a beaucoup de cas qui sont bien connus et certains problèmes de géodésique sont très étudiés même s'ils ne sont pas énoncés en ces termes.

L'approche consiste à approcher la géodésique par des courbes de Bézier par morceaux minimisant la métrique du conditionnement parmi les courbes de Bézier par morceaux. Cela permet de réduire le problème à un problème d'optimisation paramétrique où les paramètres sont les points de contrôle de la courbe de Bézier par morceaux.

Par exemple, sur la figure 3.8, on considère le cas où Σ est constitué de trois points. On cherche à joindre deux points (ici $A = \begin{pmatrix} -2.5 \\ 0 \end{pmatrix}$ et $B = \begin{pmatrix} 2.5 \\ 0 \end{pmatrix}$). La méthode proposée est locale et on optimise ici à partir de chemins initiaux dont on sait qu'ils sont dans des bassins de minima locaux différents (on ne peut pas « traverser » les points de Σ, donc comme pour passer d'une courbe à une autre il faut « traverser » un point de Σ au moins, toutes ces trajectoires sont dans des bassins différents). Ici les approximations des géodésiques sont réalisées avec des courbes de Bézier par morceaux. Il s'agit ici de la concaténation de 3 patchs cubiques.

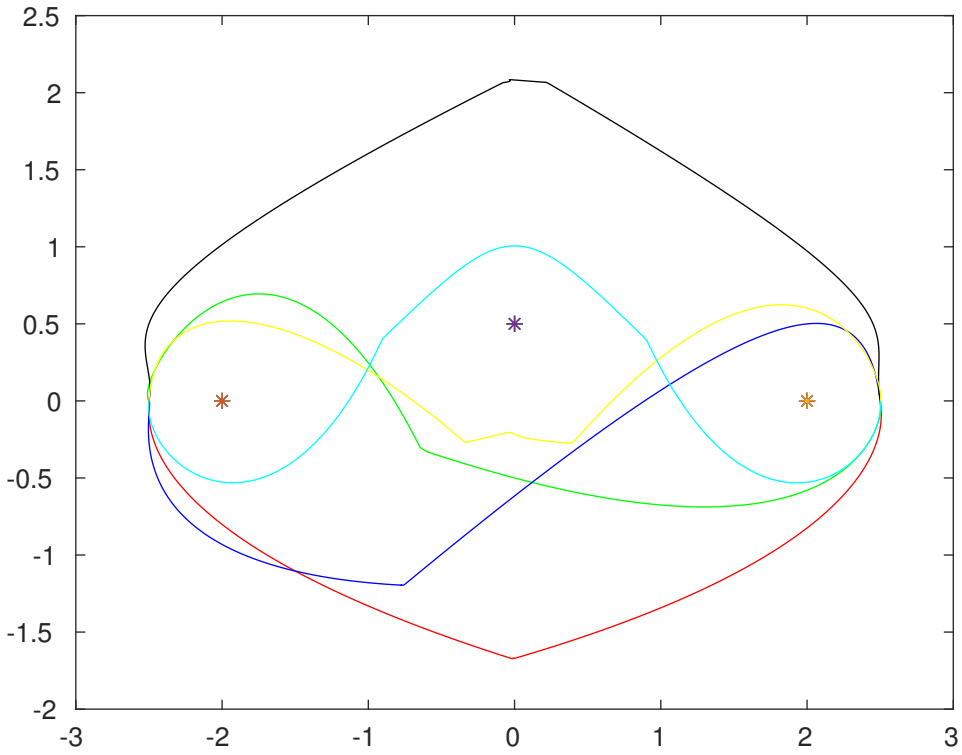

Figure 3.8. Σ est constitué de trois points

3.5. Systèmes différentiels

Dans cette section, nous allons montrer comment réduire le calcul d'orbites d'équations différentielles ordinaires aux calculs de trajectoires optimales particulières. Nous obtiendrons alors une paramétrisation d'un arc d'orbite. Nous montrerons alors comment reparamétrer cette courbe pour avoir une solution du problème de Cauchy associé. Nous étendrons alors notre démarche aux problèmes de contrôle puis nous aborderons d'un point de vu purement théorique l'application de cette approche au contrôle optimal.

3.5.1. Orbites de problèmes autonomes *vs* trajectoires optimales

Soit E un espace affine euclidien de dimension finie, on considère $F: E \longrightarrow TE$ un champ de vecteurs qu'on supposera suffisamment régulier (disons $\mathcal{C}^1$) et $y_0 \in E$, on s'intéresse alors au problème de Cauchy :

$$\begin{cases} \dot{y} = F(y) \\ y(0) = y_0 \end{cases}. \tag{3.19}$$

Soit I un intervalle de $\mathbb{R}$ contenant 0 et $y: I \longrightarrow E$ une fonction satisfaisant 3.19, on appelle arc d'orbite associé à y l'ensemble $\mathcal{O}_{y_0} = \{y(t) | t \in I\}$ qui est la courbe paramétrée par y sur I. On cherche alors à obtenir une paramétrisation de $\mathcal{O}_{y_0}$ en résolvant un problème de trajectoire optimale. C'est l'objet de la proposition suivante :

Proposition 3.17. *Soit E un espace affine euclidien de dimension finie, on considère $F: E \longrightarrow TE$ un champ de vecteurs qu'on supposera suffisamment régulier (disons $\mathcal{C}^1$) et $y_0 \in E$, on considère alors le problème de Cauchy associé :*

$$\begin{cases} \dot{y} = F(y) \\ y(0) = y_0 \end{cases}. \tag{3.20}$$

Soit alors $\gamma: [0,1] \longrightarrow E$ *une fonction lisse (disons* $\mathcal{C}^1$*) telle que* $\dot{\gamma}(v) \neq 0$ *pour tout* $v \in [0,1)$*, telle que :*

$$\begin{cases} \Phi_F(\gamma) := \int_0^1 (\|\dot{\gamma}(v)\|_2 \cdot \|F(\gamma(v))\|_2 - \langle \dot{\gamma}(v), F(\gamma(v)) \rangle)\, dv = 0, \\ \gamma(0) = X_0. \end{cases} \tag{3.21}$$

Alors il existe un intervalle $I = [0,a)$*, une solution* y *du problème 3.20 définie sur* I *et un difféomorphisme* $\varphi: I \longrightarrow [0,1)$ *tels que* $\gamma \circ \varphi = y$*.*

La proposition 3.17 signifie qu'on peut approcher un arc de l'orbite $\mathcal{O}_{y_0}$ de 3.20 en approchant une solution du problème de trajectoire optimale 3.21. Une version un peu simplifiée de cette proposition est issue de la thèse de Hoang Van Duc [30]. On y trouve surtout des expérimentations pour le calcul d'orbites de systèmes différentiels. Le problème de trajectoire optimale 3.21 était déjà abordé dans la thèse de Pierre Bonnélie [9] et elle exprime qu'une solution γ de 3.21 est telle que $\dot{\gamma}$ et $F(\gamma)$ sont colinéaires dans le même sens, autrement dit γ est une reparamétrisation d'une solution de 3.20 puisque la courbe est tout le temps tangente au champ de vecteurs. Dans la thèse de Hoang Van Duc [30], cette approche est comparée aux approches plus classiques et voici ce qu'il en ressort : plus la géométrie est compliquée, plus notre approche est performante comparativement à la méthode d'Euler par exemple mais elle ne permet pas de chercher une solution maximale *a priori*. J'ai proposé une méthode pour chercher une solution maximale (en maximisant la longueur par exemple), mais nous n'avons pas eu le temps de tester sérieusement cette idée, surtout que les équations différentielles ordinaires n'étaient pas notre principal objectif mais uniquement une étape sur les travaux actuels abordés dans [30] mais pas suffisamment développés : l'approximation de trajectoires de problèmes de contrôle qui font l'objet de la sous-section suivante.

3.5.2. Trajectoires optimales pour le contrôle

Dans cette sous-section, je montrerai comment réduire un problème de contrôle autonome en un problème de trajectoires optimales. Je renvoie à [49] pour une introduction générale à la théorie du contrôle optimal.

Je considère donc un problème de contrôle optimal « local », i.e. sur un ouvert d'un espace affine E :

$$\begin{cases} \dot{X}(t) = F(X(t), Y(t)), \\ X(0) = X_0, \\ X(T) = X_1, \end{cases} \tag{3.22}$$

où $F: U \times V \longrightarrow TU$ est $\mathcal{C}^1$ (c'est une hypothèse un peu forte), où X_0 et $X_1 \in U$ sont donnés, Y est une fonction inconnue appartenant à un ensemble d'applications appelés contrôles (généralement inclus dans $L^1_{\mathrm{loc}}(I,V)$) et $X(\cdot)$ et $t_1 \in \mathbb{R}$ sont inconnus. Cela signifie qu'il existe une fonction Y et une solution au problème de Cauchy :

$$\begin{cases} \dot{X}(t) = F(X(t), Y(t)), \\ X(0) = X_0, \end{cases}$$

telle que $X(T) = X_1$. On note $X_Y(\cdot)$ la solution de ce problème de Cauchy quand elle existe et on dit dans ce cas que l'instance est faisable ou satisfaisable ou encore que X_1 est accessible. Pour des raisons de simplicité de l'exposé, je ferai l'hypothèse que les instances auxquelles nous nous intéressons sont faisables.

Méthode de tir

Je décris ici brièvement la méthode de tir pour traiter le problème 3.22. C'est l'approche générale qui semble la plus utilisée pour ce type de problème et qui peut s'adapter au contrôle optimal. Pour traiter 3.22, on considère le problème de Cauchy suivant :

$$\begin{cases} \dot{X}_Y(t) = F(X_Y(t), Y(t)), \\ X_Y(0) = X_0, \end{cases} \tag{3.23}$$

et on note $G(Y) = X_Y(T) - X_1$. La méthode indirecte consiste à chercher Y tel que $G(Y) = 0$. C'est généralement fait en discrétisant F et Y pour avoir un nombre fini de variables puis en utilisant la méthode de Newton (il faut résoudre une ODE et calculer une variation en fonction de Y à chaque itération). Cette méthode ne s'applique pas telle quelle sur le contrôle optimal car on utilise généralement la contrainte d'optimalité pour éliminer Y de la définition de G. Le problème est que X_1 peut donner un équilibre instable (point répulsif) ce qui interdirait la convergence de la méthode de Newton.

Approche par trajectoires optimales

Pour l'approche par trajectoire optimale, comme pour le cas des équations différentielles ordinaires, on va faire abstraction de l'intervalle de paramétrisation en se concentrant sur la géométrie du problème. On cherche $\gamma\colon [0,1] \longrightarrow U$ et $\beta\colon [0,1] \longrightarrow V$ continuement différentiables telles que $\dot{\gamma}(v) \neq 0$ pour tout $v \in [0,1]$ et satisfaisant :

$$\begin{cases} \Phi_F(\gamma,\beta) = \int_0^1 (\|\dot{\gamma}(v)\|_2 \cdot \|F(\gamma(v),\beta(v))\|_2 - \langle \dot{\gamma}(v), F(\gamma(v),\beta(v))\rangle)\, dv = 0, \\ \gamma(0) = X_0, \\ \gamma(1) = X_1. \end{cases} \tag{3.24}$$

Si on a un couple (γ,β) solution de 3.24, alors il existe un difféomorphisme $\varphi\colon [0,T] \longrightarrow [0,1]$ tel que $(\gamma\circ\varphi, \beta\circ\varphi)$ soit solution de 3.22. Durant la thèse de Hoang Van Duc, nous avons pu tester que cette approche fonctionne et cela même si X_1 est répulsif (car on force la solution à y aboutir). On utilise les courbes de Bézier par morceaux pour approcher à la fois γ et β.

3.6. Conclusions

Dans ce chapitre, j'espère avoir montré que je poursuis des projets scientifiques à longue échéance, n'hésitant pas à investir de nouveaux domaines soit pour des raisons du contexte du laboratoire, soit pour avancer sur des sujets qui me permettront de revenir sur des questions que je n'ai pas réussi à traiter de prime abord. Le point de vue géométrique pour l'étude des systèmes algébriques est une motivation profonde et continue. L'investissement sur les thématiques d'optimisation de formes a donné lieu à l'encadrement de trois thèses (Pierre Bonnélie, Hoang Van Duc et Ali Dia) deux post-doctorats (Marie-Ève Modolo et Satafa Sanogo) et de très nombreux mémoires de M2 ACSYON. Les échanges avec des doctorants est une priorité, même si je ne prends pas part à leur encadrement (comme pour la thèse de Pauline Merveilleux-Orzekowska par exemple). J'ai assumé un rôle d'animateur scientifique en proposant des collaborations avec d'autres équipes de l'axe MATHIS, comme avec l'équipe MODE (Loïc Bourdin et Paul Armand) ou dans d'autres axes (Stéphane Bila, Christophe Durousseau et Cyrille Menudier). Encore une fois, motivé par des questions parfois très théoriques, je prends soin d'aller étudier les applications. Cet échange entre théorie et applications permet de maintenir une cohérence et un intérêt sur mes thématiques de recherche. Enfin, des questions nouvelles (comme sur le contrôle) viennent parfois de considérations issues des applications.

Chapitre 4

Conclusions

Dans ce mémoire, j'ai présenté une partie importante de mes recherches en les contextualisant en termes de collaborations scientifiques, d'encadrement doctoral et d'animation scientifique du laboratoire. J'ai montré la cohérence de ce parcours scientifique avec le recours à l'algèbre des polynômes commutatifs ou non-commutatifs pour la représentaion et la manipulation d'objets mathématiques. J'ai illustré l'importance que je donne au fait de pouvoir aller de sujets très théoriques jusqu'à des applications concrètes. La partie scientifique de ce mémoire n'a pas pour objectif d'être exhaustive sur les thématiques que j'aborde dans mes recherche, ni d'aileurs sur les résultats des sujets qui y sont abordés. Il a pour objectif d'illustrer ma méthode de travail et l'équilibre que j'essaie de maintenir entre recherche et formation. J'ai illustré la variété des sujets, des projets et des productions scientifiques de ma recherche. J'espère que ce texte permet de confirmer que j'ai acquis une certaine maturité scientifique, une capacité à prendre des responsabilités et à mener à bien des projets. Mais j'epère aussi avoir convaincu que j'ai encadré avec bienveillance et ouverture d'esprit de jeunes chercheurs. Je suis conscient des limites de ce texte et j'espère qu'il ne sera qu'une étape dans ma carrière d'enseignant-chercheur pour aller plus loin et mener de nouveaux projets.

Bibliographie

[1] François Apéry et Jean-Pierre Jouanolou. *Résultant et sous-résultants: le cas d'une variable avec exercices corrigés*. Collection Méthodes. Hermann, Paris, 2006.

[2] Nicolas Aragon, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Adrien Hauteville, Olivier Ruatta, Jean-Pierre Tillich, et Gilles Zémor. LAKE-Low rAnk parity check codes Key Exchange. 2017.

[3] Magali Bardet, Pierre Briaud, Maxime Bros, Philippe Gaborit, Vincent Neiger, Olivier Ruatta, et Jean-Pierre Tillich. An algebraic attack on rank metric code-based cryptosystems. Dans *Advances in cryptology—EUROCRYPT 2020. Part III*, volume 12107 de *Lecture Notes in Comput. Sci.*, pages 64–93. Springer, Cham, 2020.

[4] Magali Bardet, Maxime Bros, Daniel Cabarcas, Philippe Gaborit, Ray Perlner, Daniel Smith-Tone, Jean-Pierre Tillich, et Javier Verbel. Improvements of Algebraic Attacks for Solving the Rank Decoding and MinRank Problems. Dans Shiho Moriai et Huaxiong Wang, éditeurs, *Advances in Cryptology – ASIACRYPT 2020*, pages 507–536. Cham, 2020. Springer International Publishing.

[5] Anne-Mercedes Bellido. *Construction de fonctions d'itération pour le calcul simultané des solutions d'équations et de systèmes d'équations algébriques*. These de doctorat, Toulouse 3, jan 1992.

[6] Thierry P. Berger, Philippe Gaborit, et Olivier Ruatta. Gabidulin matrix codes and their application to small ciphertext size cryptosystems. Dans *Progress in cryptology—INDOCRYPT 2017*, volume 10698 de *Lecture Notes in Comput. Sci.*, pages 247–266. Springer, Cham, 2017.

[7] Thierry P. Berger, Cheikh Thiécoumba Gueye, Jean Belo Klamti, et Olivier Ruatta. Designing a Public Key Cryptosystem Based on Quasi-cyclic Subspace Subcodes of Reed-Solomon Codes. Dans Cheikh Thiecoumba Gueye, Edoardo Persichetti, Pierre-Louis Cayrel, et Johannes Buchmann, éditeurs, *Algebra, Codes and Cryptology*, volume 1133, pages 97–113. Springer International Publishing, Cham, 2019. Series Title: Communications in Computer and Information Science.

[8] P. Boito et O. Ruatta. Extended companion matrix for approximate GCD. Pages 74–80. 2011.

[9] Pierre Bonnelie. *Déformations libres de contours pour l'optimisation de formes et application en électromagnétisme*. Theses, Université de Limoges, feb 2017. Issue: 2017LIMO0006.

[10] Pierre Bonnelie, Loïc Bourdin, Fabien Caubet, et Olivier Ruatta. Flip procedure in geometric approximation of multiple-component shapes—application to multiple-inclusion detection. *SMAI Journal of Computational Mathematics*, 2:255–276, 2016.

[11] Pierre Bonnelie, Olivier Ruatta, Satafa Sanogo, et Stéphane Bila. Free-form method for designing an optimal microwave filter. Dans *International Symposium on Electric and Magnetic Fields (EMF 2016)*. Lyon, France, 2016.

[12] Delphine Boucher, Philippe Gaborit, Willi Geiselmann, Olivier Ruatta, et Felix Ulmer. Key exchange and encryption schemes based on non-commutative skew polynomials. Dans *Post-quantum cryptography*, volume 6061 de *Lecture Notes in Comput. Sci.*, pages 126–141. Springer, Berlin, 2010.

[13] F. Chabaud et J. Stern. The cryptographic security of the syndrome decoding problem for rank distance codes. Dans Kwangjo Kim et Tsutomu Matsumoto, éditeurs, *Advances in Cryptology — ASIACRYPT '96*, pages 368–381. Berlin, Heidelberg, 1996. Springer.

[14] A. Dia, C. Durousseau, C. Menudier, L. Carpentier, O. Ruatta, et S. Bila. Gradient Descent Shape Optimization of Microwave Circuits using Bézier Curves Parametrization. Pages 158–161. 2018.

[15] A. Dia, C. Durousseau, C. Menudier, L. Carpentier, O. Ruatta, et S. Bila. Shape Optimization Methods for the Design of Microwave Circuits and Antennas. 2018.

[16] Ali Dia, Christophe Durousseau, Cyrille Menudier, Ludovic Carpentier, Olivier Ruatta, et Stéphane Bila. Bézier Curve Parametrization for Gradient Descent Shape Optimization of Microwave Circuits. Dans *International Workshop on Microwave Filters (7th IWMF)*. Noordwijk, Netherlands, apr 2018. European Space Agency (ESA) and Centre National D'Etudes Spatiales (CNES).

[17] Ali Dia, Christophe Durousseau, Cyrille Menudier, Ludovic Carpentier, Olivier Ruatta, et Stéphane Bila. Optimisation de formes de circuits hyperfréquences par un paramétrage utilisant des courbes de bézier couplées à une méthode de gradients. Dans *XXIèmes Journées Nationales Microondes*. Caen, France, may 2019.

[18] Ali Dia, Christophe Durousseau, Cyrille Menudier, Ludovic Carpentier, Olivier Ruatta, et Stéphane Bila. Shape Optimization of a Compact Dual-mode Filter Using Bézier Curves Parametrization. Dans *2021 IEEE MTT-S International Microwave Filter Workshop (IMFW)*, pages 137–139. Nov 2021.

[19] Daouda Niang Diatta. *Calcul effectif de la topologie de courbes et surfaces algébriques réelles*. Theses, Université de Limoges, sep 2009.

[20] Lijun Ding et Ardeshir Goshtasby. On the Canny edge detector. *Pattern Recognition*, 34(3):721–725, mar 2001.

[21] Gerald E. Farin et Gerald Farin. *Curves and Surfaces for CAGD: A Practical Guide*. Morgan Kaufmann, 2002. Google-Books-ID: 5HYTP1dIAp4C.

[22] Jean-Charles Faugère, Françoise Levy-dit-Vehel, et Ludovic Perret. Cryptanalysis of MinRank. Dans David Wagner, éditeur, *Advances in Cryptology – CRYPTO 2008*, pages 280–296. Berlin, Heidelberg, 2008. Springer.

[23] P. Gaborit et O. Ruatta. Efficient erasure list-decoding of Reed-Muller codes. Pages 148–152. 2006.

[24] Philippe Gaborit, Gaétan Murat, Olivier Ruatta, et Gilles Zémor. Low rank parity check codes and their application to cryptography. Dans *Proceedings of the Workshop on Coding and Cryptography WCC*, volume 2013. 2013.

[25] Philippe Gaborit et Olivier Ruatta. Improved Hermite multivariate polynomial interpolation. Dans *2006 IEEE International Symposium on Information Theory*, pages 143–147. Jul 2006. ISSN: 2157-8117.

[26] Philippe Gaborit, Olivier Ruatta, et Julien Schrek. On the complexity of the rank syndrome decoding problem. *Institute of Electrical and Electronics Engineers. Transactions on Information Theory*, 62(2):1006–1019, 2016.

[27] M. Giesbrecht. Factoring in Skew-polynomial Rings over Finite Fields. *Journal of Symbolic Computation*, 26(4):463–486, oct 1998.

[28] V. Guruswami et M. Sudan. Improved decoding of Reed-Solomon and algebraic-geometric codes. Dans *Proceedings 39th Annual Symposium on Foundations of Computer Science (Cat. No.98CB36280)*, pages 28–37. Nov 1998. ISSN: 0272-5428.

[29] Adrien Hauteville. *Nouveaux protocoles et nouvelles attaques pour la cryptologie basée sur les codes en métrique rang*. Theses, Université de Limoges, dec 2017. Issue: 2017LIMO0088.

[30] Van Duc Hoang. *Distance and geometry of the set of curves and approximation of optimal trajectories*. Theses, Université de Limoges, feb 2020. Issue: 2020LIMO0013.

[31] Ouiddad Labbani I., Pauline Merveilleux O, et Olivier Ruatta. Free Form based active contours for image segmentation and free space perception. *ArXiv:1606.04774 [cs]*, jun 2016. ArXiv: 1606.04774.

[32] Nathan Jacobson. *The theory of rings*. Nombre 2 dans Mathematical surveys. American Mathematical Soc, Providence, RI, 10. print edition, 1998.

[33] N. Karmarkar et Y. N. Lakshman. Approximate polynomial greatest common divisors and nearest singular polynomials. Dans *Proceedings of the 1996 international symposium on Symbolic and algebraic computation*, ISSAC '96, pages 35–39. New York, NY, USA, oct 1996. Association for Computing Machinery.

[34] Michael Kass, Andrew Witkin, et Demetri Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, jan 1988.

[35] Ziming Li. A subresultant theory for Ore polynomials with applications. Dans *Proceedings of the 1998 international symposium on Symbolic and algebraic computation - ISSAC '98*, pages 132–139. Rostock, Germany, 1998. ACM Press.

[36] Pierre Loidreau. *Metrique rang et cryptographie*. Thesis, Université Pierre et Marie Curie - Paris VI, jan 2007.

[37] Robert J McEliece. A public-key cryptosystem based on algebraic. *Coding Thv*, 4244:114–116, 1978.

[38] Pauline Merveilleux-Orzekowska. *Exploration et navigation de robots basées vision omnidirectionnelle*. These de doctorat, Amiens, jan 2012.

[39] Bernard Mourrain et Olivier Ruatta. Relations between roots and coefficients, interpolation and application to system solving. Dans *Journal of Symbolic Computation*, volume 33, pages 679–699. 2002. ISSN: 0747-7171 Journal Abbreviation: J. Symbolic Comput.

[40] Gaetan Murat. *Résultants de polynômes de Ore et Cryptosystèmes de McEliece sur des Codes Rang faiblement structurés*. Theses, Université de Limoges, dec 2014. Issue: 2014LIMO0061.

[41] Oystein Ore. On a Special Class of Polynomials. *Transactions of the American Mathematical Society*, 35(3):559–584, 1933. Publisher: American Mathematical Society.

[42] Oystein Ore. Theory of Non-Commutative Polynomials. *Annals of Mathematics*, 34(3):480–508, 1933. Publisher: Annals of Mathematics.

[43] A. V. Ourivski et T. Johansson. New Technique for Decoding Codes in the Rank Metric and Its Cryptography Applications. *Problems of Information Transmission*, 38(3):237–246, jul 2002.

[44] Olivier Ruatta. *Dualité algébrique, structures et applications*. PhD thesis, Université d'Aix-Marseille 2, 2002.

[45] Olivier Ruatta. On the geometry and the deformation of shapes represented by piecewise continuous Bézier curves with application to shape optimization. Dans *Geometric science of information*, volume 8085 de *Lecture Notes in Comput. Sci.*, pages 112–119. Springer, Heidelberg, 2013.

[46] Olivier Ruatta, Mark Sciabica, et Agnes Szanto. Overdetermined Weierstrass iteration and the nearest consistent system. *Theoretical Computer Science*, 562:346–364, 2015.

[47] David Rupprecht. *Elements de geometrie algebrique approchee : etude du pgcd et de la factorisation*. These de doctorat, Nice, jan 2000.

[48] Madhu Sudan. Decoding of Reed Solomon Codes beyond the Error-Correction Bound. *Journal of Complexity*, 13(1):180–193, mar 1997.

[49] Emmanuel Trélat. *Contrôle optimal: théorie & applications*. Vuibert, Paris, 2005. OCLC: 749708838.

[50] Laurent Younes. Computable Elastic Distances Between Shapes. *SIAM Journal on Applied Mathematics*, 58(2):565–586, apr 1998. Publisher: Society for Industrial and Applied Mathematics.

[51] Laurent Younes. *Shapes and Diffeomorphisms*, volume 171 de *Applied Mathematical Sciences*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2019.

[52] Laurent Younes, Peter W. Michor, Jayant M. Shah, et David B. Mumford. A metric on shape space with explicit geodesics. *Rendiconti Lincei*, 19(1):25–57, mar 2008.