THESE DE DOCTORAT de l'UNIVERSITE PARIS 6

Spécialité

Informatique

présentée par

Mme Magali TURREL BARDET

Pour obtenir le grade de
DOCTEUR de l'UNIVERSITE PARIS 6

Sujet de la thèse :

Étude des systèmes algébriques surdéterminés. Applications aux codes correcteurs et à la cryptographie.

Soutenue le 8 Décembre 2004

devant le jury composé de :

Présidente et Rapporteure :	
Mme Brigitte Vallée	DR CNRS, Univ. de Caen
Rapporteur :	
M. Patrick Fitzpatrick	Professeur, Univ. de Cork (Irlande)
Examinateurs :	
M. Jean-Marie Chesneaux	Professeur, Univ. Paris 6
M. Antoine Joux	DGA et Professeur associe à l'UVSQ
M. Daniel Lazard	Professeur, Univ. Paris 6
M. Bruno Salvy	DR, INRIA Rocquencourt
Directeurs :	
M. Daniel Augot	CR, INRIA Rocquencourt
M. Jean-Charles Faugère	CR CNRS, Univ. Paris 6

Table des matières

Remerciements

Mes remerciements vont en premier lieu à Daniel Augot et Jean-Charles Faugère, qui m'ont proposé ce sujet de thèse mêlant calcul formel, codes correcteurs et cryptologie et m'ont efficacement co-encadrée durant mon stage de DEA et mes années de thèse. Je souhaite à tous les thésards de pouvoir bénéficier d'un tel encadrement. J'ai particulièrement apprécié leur disponibilité à tout instant, et la grande liberté qu'ils m'ont laissée dans l'orientation de mes sujets de recherche.

Je remercie vivement Daniel Lazard, avec qui j'ai découvert le calcul formel et qui m'a orientée vers Jean-Charles et Daniel pour démarrer ma thèse. Nous avons eu de nombreuses et stimulantes discussions scientifiques qui m'ont beaucoup aidée à prendre du recul sur mes travaux. J'ai aussi pris grand plaisir à discuter de montagne ou de mer avec lui.

Bruno Salvy a été mon co-auteur pour les résultats d'analyse asymptotique du chapitre 4, et je le remercie de m'avoir fait découvrir la puissance et l'élégance des séries génératrices et des méthodes de cols. Je le remercie également pour les conseils de rédaction et de présentation qu'il m'a prodigués au cours de cette thèse, et pour sa participation à mon jury de thèse.

Je tiens à exprimer ma profonde gratitude à Brigitte Vallée et Patrick Fitzpatrick, qui m'ont fait l'honneur de rapporter ma thèse. Je remercie Brigitte Vallée pour l'intérêt et le soutien chaleureux qu'elle m'a apportés en cette fin de thèse. Je remercie Patrick Fitzpatrick pour les échanges amicaux que nous avons eus par email et ses encouragements.

Je suis très reconnaissante à Antoine Joux et Jean-Marie Chesneaux d'avoir accepté de faire partie de mon jury de thèse, témoignant ainsi de leur intérêt pour ma recherche.

Je remercie profondément Marie-Françoise Roy pour son intérêt pour mes travaux de recherche, et son accueil toujours chaleureux à Rennes.

Ma thèse s'est déroulée au sein de l'équipe CALFOR du LIP6, et j'ai eu beaucoup de plaisir à participer à la vie de cette équipe durant ces années. Je remercie tous les membres de CALFOR, et en particulier David, qui a l'art de la résolution quasi instantanée des tâches administratives ; Jean-Michel et Cyriaque nos ingénieurs système ; mes collègues de bureau successifs : Guénaël, Amir, Philippe, Gwenolé, Louis, Solen ; Mohab pour son dynamisme communicatif ; Fabrice pour tous les services qu'il m'a rendus.

Je remercie les équipes des projets Codes et Algo de l'Inria, qui m'ont accueillie

chaleureusement lors des longues journées de labeur passées à l'Inria avec Daniel ou Bruno. Je remercie également Caroline Fontaine, ancienne du projet Codes, devenue une amie depuis notre séjour au Japon.

Le travail d'enseignante à été une activité importante parallèlement à ce travail de thèse. Je voudrais remercier Michèle Soria, qui m'a beaucoup appris et m'a accordé sa confiance en m'intégrant dans ses équipes enseignantes. Je remercie également tous ceux avec qui j'ai eu l'occasion d'enseigner, entre autres Emmanuel(s), Jean-Charles, Maryse, Michelle, Philippe(s), Stéphane, Valérie, ...

Merci aux amis avec qui j'ai partagé de bons moments pendant ma thèse, Céline et Philippe, Chi-Tuong, Christian et Élise, Arnaud et Éloïse, Anne-Laure, Thierry, et tous les autres... Merci à Marc Moreno-Maza, Daniel et Jean-Charles de m'avoir appris ce qu'était la célèbre cucaracha.

Merci enfin à ma famille, pour son soutien et ses encouragements tout au long de cette thèse, et en particulier : Marinette et Gérard qui m'accueillent sur Paris cette année, Nicolas qui finira par comprendre mon sujet de thèse après avoir relu plusieurs introductions en anglais, Marie et Émilie qui ont partagé un temps notre vie parisienne, Pierre-Luc, Louise, mes parents Mireille et Gérard, mes beaux-parents Bernadette et Claude.

La rédaction de cette thèse aurait sans doute été plus difficile sans le soutien et l'aide de Jean-Baptiste, qui m'a permis de me replonger totalement dans ma rédaction de thèse après la naissance de notre fille Claire en s'occupant d'elle pendant mes séjours à Paris. Je le remercie aussi pour son aide quotidienne et son amour qui m'ont embelli la vie, même durant les moments de doute inévitables de la vie d'une thésarde (où je ne lui rendais pourtant pas la tâche facile !).

à Claire

Résumé

Les bases de Gröbner constituent un outil important pour la résolution de systèmes d'équations algébriques, et leur calcul est souvent la partie difficile de la résolution. Cette thèse est consacrée à des analyses de complexité de calculs de bases de Gröbner pour des systèmes surdéterminés (le nombre m d'équations est supérieur au nombre n d'inconnues).

Dans le cas générique ("aléatoire"), des outils existent pour analyser la complexité du calcul de base de Gröbner pour un système non surdéterminé (suites régulières, borne de Macaulay). Nous étendons ces résultats au cas surdéterminé, en définissant les suites semi-régulières et le degré de régularité dont nous donnons une analyse asymptotique précise. Par exemple dès que $m > n$ nous gagnons un facteur 2 sur la borne de Macaulay, et un facteur 11,65 quand $m = 2n$ (ces facteurs se répercutent sur l'exposant de la complexité globale). Nous déterminons la complexité de l'algorithme F5 (J-C. Faugère) de calcul de base de Gröbner.

Ces résultats sont appliqués en protection de l'information, où les systèmes sont alors considérés modulo 2 : analyse de la complexité des attaques algébriques sur des cryptosystèmes, algorithmes de décodage des codes cycliques. Dans ce dernier cas, une remise en équation complète du problème conduit à utiliser des systèmes de dimension positive dont la résolution est de manière surprenante plus rapide. Nous obtenons ainsi un algorithme de décodage efficace de codes précédemment indécodables, permettant un décodage en liste et applicable à tout code cyclique.

Abstract

Gröbner bases constitute an important tool for solving algebraic systems of equations, and their computation is often the hard part of the resolution. This thesis is devoted to the complexity analysis of Gröbner basis computations for overdetermined algebraic systems (the number m of equations is greater than the number n of variables).

In the generic ("random") case, tools exist to analyze the complexity of Gröbner basis computations for a non overdetermined system (regular sequences, Macaulay bound). We extend these results to the overdetermined case, by defining the semi-regular sequences and the degree of regularity for which we give a precise asymptotic analysis. For example as soon as $m > n$ we gain a factor 2 on the Macaulay bound, and a factor 11,65 when $m = 2n$ (these factors are reflected on the exponent of global complexity). We determine the complexity of the F5 Algorithm (J-C. Faugère) for computing Gröbner bases.

These results are applied in information theory, where the systems are then considered modulo 2 : analysis of the complexity of the algebraic attacks on cryptosystems, algorithms for the decoding of cyclic codes. In this last case, a new equation set-up of this problem leads to use systems of positive dimension for which the resolution is in a surprising way faster. We thus obtain an effective algorithm for decoding codes previously undecodable, allowing list decoding and applicable to any cyclic code.

Introduction

Le problème de la résolution des systèmes d'équations algébriques est un problème fondamental dont les applications, tant académiques qu'industrielles (robotique, codes correcteurs, cryptographie, etc.), sont nombreuses. Les problèmes provenant d'applications industrielles comme la robotique possèdent généralement autant de contraintes que de paramètres, et les solutions intéressantes sont les solutions réelles. Cela signifie que l'on cherche à résoudre un système d'équations polynomiales $f_1(x_1, \ldots, x_n) = 0, \ldots, f_n(x_1, \ldots, x_n) = 0$, où les f_i sont des polynômes à coefficients rationnels (nous omettrons par la suite les égalités à zéro et les variables, et considèrerons le système $f_1, \ldots, f_n$). Un tel système possède généralement un nombre fini de solutions, dont la plupart ne sont pas rationnelles ou réelles mais complexes, et l'expression de ces solutions ne peut se faire explicitement.

Les domaines des codes correcteurs d'erreurs et de la cryptographie constituent un autre champ d'application de la résolution de systèmes d'équations algébriques. Pour les systèmes à coefficients rationnels, le problème de trouver les racines réelles n'est pas algébrique. Par contre, le problème de trouver les solutions d'un système dans un corps fini l'est : il suffit en effet pour obtenir les solutions dans le corps fini $\mathbb{F}_q$ d'ajouter pour chaque variable x l'équation $x^q - x$. Le système obtenu possède alors nécessairement un nombre fini de solutions, et est fortement surdéterminé (il possède plus d'équations que d'inconnues) : il est formé des n équations de corps (pour chacune des n variables), plus les équations de départ qui peuvent être également très nombreuses (par exemple des relations entre des messages en clair et des messages chiffrés en cryptographie).

Les bases de Gröbner constituent un outil important pour la résolution de systèmes algébriques, et leur calcul est souvent la partie difficile de la résolution. Elles servent par exemple à résoudre le problème Ideal Membership d'appartenance à un idéal, qui est EXPSPACE-complet. La complexité du calcul de base de Gröbner dans le cas le pire est $2^{2^{O(n)}}$ (où n est le nombre de variables), mais pour des systèmes possédant un nombre fini de solutions, la complexité n'est plus que simplement exponentielle : $D^{O(n^2)}$ (où D est le maximum des degrés des polynômes engendrant l'idéal).

Il est alors naturel de se demander ce que devient cette complexité théorique pour un système surdéterminé ou pour un système "tiré au hasard" ? Peut-on donner des bornes précises de complexité pour le calcul de la base de Gröbner ? Pour des

systèmes avec équations de corps, le coût de la résolution est au pire celle de la recherche exhaustive, en $O(n)2^n$ sur $\mathbb{F}_2$. Comment se situe le coût du calcul de la base de Gröbner par rapport à cette recherche exhaustive ?

Cette thèse est consacrée à des analyses de complexité de calculs de bases de Gröbner, pour des systèmes surdéterminés en général et en particulier pour des systèmes contenant des équations de corps. Ces résultats sont utilisés pour étudier le décodage algébrique des codes correcteurs d'erreurs, ainsi que pour analyser des attaques algébriques en cryptographie.

Mise en équations. Codes correcteurs. D'une manière générale, il existe pour un problème donné plusieurs mises en équations possibles. Le choix d'une *bonne* mise en équation est crucial, les calculs de bases de Gröbner pour différentes mises en équations pouvant aller de quelques secondes à plusieurs heures, voire être impossibles avec les algorithmes existants. Le choix des variables, l'utilisation des symétries du problème, l'élimination de solutions parasites peuvent avoir un effet déterminant sur la complexité de résolution du problème.

Les chapitres 2 et 6 de cette thèse sont consacrés au décodage algébrique des codes correcteurs. La mise en équation est une étape cruciale pour obtenir un bon algorithme de décodage. Nous récapitulons dans le chapitre 2 tous les systèmes algébriques utilisés pour décoder les codes cycliques, et nous en donnons au chapitre 6 une classification. Tous ces systèmes contiennent des équations de corps, de la forme $x^{2^m} - x$ où m peut être grand. Ces équations permettent d'éliminer toutes les solutions parasites, mais paradoxalement leur présence constitue l'étape bloquante du calcul de la base de Gröbner.

Nous proposons dans le chapitre 6 de nouvelles mises en équations, qui ne contiennent plus les équations de corps. Bien qu'il soit surprenant à priori d'utiliser pour le décodage des systèmes possédant une infinité de solutions, nous montrons que les solutions parasites sont bien contrôlées, et que ces systèmes résolvent bien théoriquement le problème du décodage. En pratique, le calcul de la base de Gröbner est bien plus efficace, et cela nous permet de donner des algorithmes de décodage efficaces, pour de nombreux codes pour lesquels aucun algorithme n'était connu auparavant.

Suites régulières. La complexité dans le cas le pire d'un calcul de base de Gröbner est connue depuis l'exemple de Mayr-Meyer : elle est doublement exponentielle en le nombre de variables. Mais de nombreux travaux ont montré que ce pire cas n'arrive presque jamais, et que, sous des hypothèses vérifiées par "presque tous" les systèmes, et en particuliers ceux contenant des équations de corps, la complexité était simplement exponentielle.

Les suites régulières ont été introduites par Macaulay au début du XX$^{\text{ème}}$ siècle, comme classe naturelle de systèmes "non singuliers". Une bonne mesure de complexité du calcul d'une base de Gröbner est le degré maximal atteint par les polynômes intervenant au cours du calcul. Pour une suite régulière de $m \leq n$ équations

homogènes en n variables, on peut analyser précisément la complexité du calcul de la base de Gröbner : le degré maximal des polynômes intervenant dans le calcul de la base de Gröbner est majoré par $\sum_{i=1}^{m}(d_i - 1) + 1$, appelée borne de Macaulay, où d_i est le degré du $i^{\text{ème}}$ polynôme de la suite. De plus, les suites régulières sont "génériques" : presque toute suite de $m \leq n$ équations en n variables est régulière.

Il n'existe pas de suites régulières surdéterminées, en particulier parce que l'ensemble des solutions d'un système défini par une suite régulière de m équations en n variables est de dimension $n - m$, ce qui impose $m \leq n$.

L'une des contributions principales de cette thèse est la définition et l'analyse des suites *semi-régulières* (Chapitre 3), qui étendent la notion de suites régulières aux systèmes surdéterminés. Nous montrons que les suites semi-régulières homogènes sont exactement celles dont la série de Hilbert est

$$\Big[\prod_{i=1}^{m}(1 - z^{d_i})\Big/(1 - z)^n\Big]$$

et que le degré maximal d'un élément de la base de Gröbner est le premier degré d pour lequel le $d^{\text{ème}}$ coefficient de la série $\prod_{i=1}^{m}(1 - z^{d_i})/(1 - z)^n$ est négatif ou nul.

L'existence de suites semi-régulières est une question très importante. Nous avons vu que presque toute suite est régulière, la conjecture de Fröberg [Frö85] généralise cette propriété aux suites semi-régulières. Cette conjecture est prouvée dans le cas de $n+1$ polynômes en n variables, mais dès que le nombre de variables est $m > n + 1$ il n'existe pas de famille connue de suite semi-régulière (par contre, à m, n et le degré des polynômes fixés, il est facile de construire une suite semi-régulière).

Analyse de F5. L'outil que nous utilisons pour analyser les suites semi-régulières est l'algorithme F5 de Jean-Charles Faugère, ou plus précisément une version matricielle de cet algorithme, appelée F5-matriciel, que nous décrivons chapitre 1. Nous montrons que les suites semi-régulières sont celles pour lesquelles il n'y a pas de réduction à zéro dans l'algorithme F5-matriciel. Cette propriété nous permet de prévoir le comportement de l'algorithme à chaque étape du calcul.

Pour des suites régulières homogènes en position de Noether, nous donnons une borne sur le nombre de polynômes de la base de Gröbner pour l'ordre du degré inverse lexicographique, ce qui nous donne en corollaire une nouvelle preuve de la borne de Macaulay. Nous donnons également une formule explicite bornant le nombre d'opérations élémentaires de l'algorithme F5-matriciel. En comparaison, pour l'algorithme de Buchberger, de telles bornes n'existent que pour des systèmes de 2 variables (Buchberger) et 3 variables (Winkler).

Analyse asymptotique. Le chapitre 4 est consacré à l'analyse asymptotique du degré maximal d'un élément d'une base de Gröbner, lorsque le nombre de variables tend vers l'infini. Nous obtenons des formules généralisant la borne de Macaulay :

par exemple, pour une suite semi-régulière de $m = n + k$ polynômes en n variables, avec $k > 0$ fixé, ce degré maximal est asymptotiquement équivalent à $\frac{\sum_{i=1}^{m}(d_i-1)}{2}$, où d_i est le degré du $i^{\text{ème}}$ polynôme de la suite et $k > 0$ est une constante.

Les méthodes utilisées (méthode du col et des cols coalescents) permettent d'obtenir tous les termes du développement asymptotique de ce degré maximal. Ainsi, pour $2n$ équations quadratiques en n variables, le degré maximal d'un élément de la base de Gröbner se comporte asymptotiquement lorsque $n \to \infty$ comme :

$$d \;=\; 0.0858\, n + 1.04\, n^{\frac{1}{3}} - 1.47 + \frac{1.71}{n^{\frac{1}{3}}} + o(\frac{1}{n^{\frac{1}{3}}})$$

ce qui améliore d'un facteur de l'ordre de 11 la borne de Macaulay $(n + 1)$, et nous donnons la valeur exacte de chaque coefficient.

De plus, les premiers termes de ce développement asymptotique du degré maximal donnent déjà une très bonne estimation de sa valeur exacte pour de petites valeurs de n.

Systèmes avec équations de corps L'analyse de complexité des systèmes à coefficients dans le corps fini $\mathbb{F}_2$ contenant les équations de corps a de nombreuses applications, notamment en cryptographie (nous en présentons certaines au chapitre 5). Dans cette thèse, nous donnons une définition particulière de suites semi-régulières sur $\mathbb{F}_2$, et une version de l'algorithme F5-matriciel incluant un nouveau critère, qui vérifient que les suites semi-régulières sur $\mathbb{F}_2$ sont exactement celles pour lesquelles il n'y a aucune réduction à zéro dans cette version de l'algorithme F5-matriciel. Toutes les propriétés des suites semi-régulières restent vraies : nous montrons en particulier que les suites semi-régulières sur $\mathbb{F}_2$ sont exactement celles dont la série de Hilbert est

$$\Big[(1+z)^n \Big/ \prod_{i=1}^{m}(1+z^{d_i})\Big]$$

pour des suites de m équations en n variables de degrés $d_1, \ldots, d_m$ plus les équations de corps $x_1^2 + x_1, \ldots, x_n^2 + x_n$. Il est possible d'obtenir un développement asymptotique de la régularité en utilisant les mêmes méthodes que dans le cas général.

Plan de lecture

Chaque chapitre contient une introduction présentant brièvement son contenu.

Cette thèse aborde des domaines très différents : codes correcteurs, cryptographie, algèbre commutative, bases de Gröbner, analyse asymptotique. Nous avons choisi de la découper en deux grandes parties, en séparant les rappels des contributions. Les préliminaires nécessaires à la compréhension des chapitres de la seconde partie font l'objet de chapitres dans la première partie.

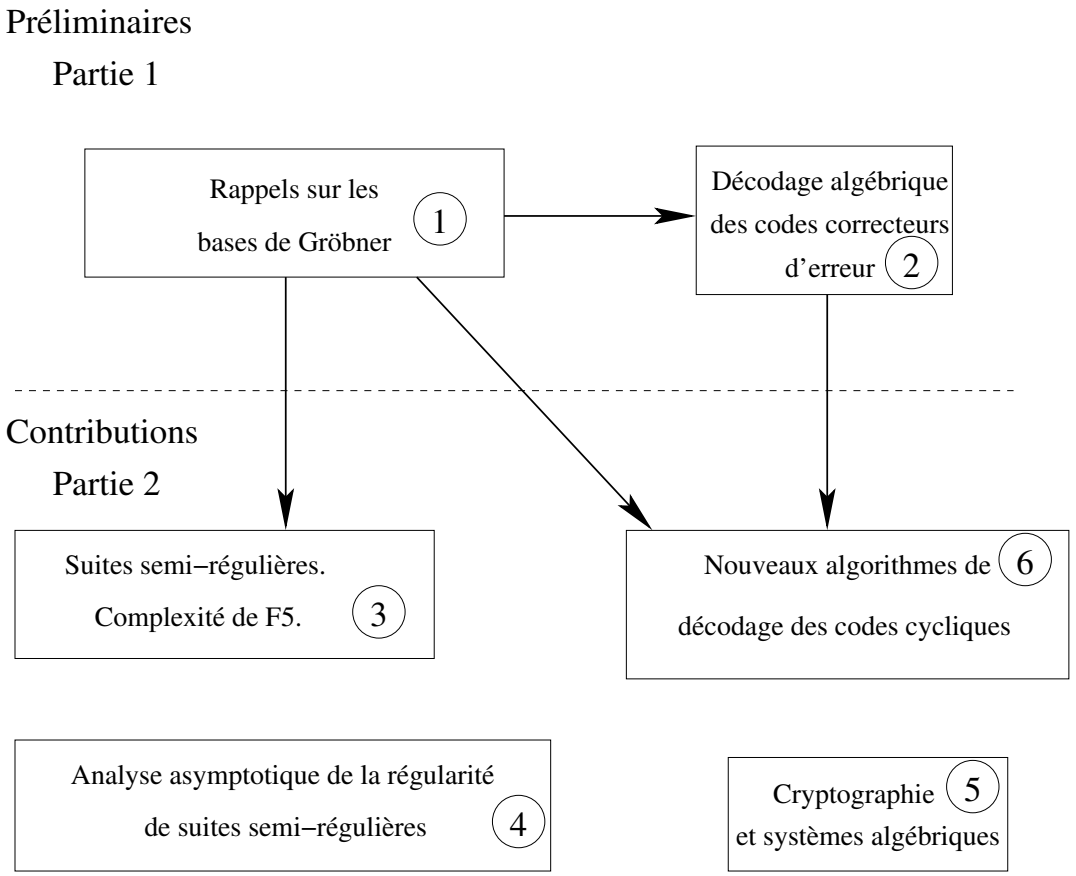


FIG. 1 – Plan de lecture de la thèse

Les rappels sur les bases de Gröbner, Chapitre 1, sont utilisés dans presque tous les chapitres de cette thèse, excepté le chapitre 4. En particulier, le chapitre 3 fait référence aux suites régulières et à l'algorithme F5-matriciel, qui sont définis et présentés dans ce chapitre 1.

Le chapitre 2 présente le décodage algébrique des codes cycliques, il est préférable de le lire avant le chapitre 6 qui décrit de nouveaux algorithmes pour le décodage algébrique des codes cycliques. Les propriétés théoriques (spécialisation, élimination) autant que pratiques (choix de la stratégie de calcul) des bases de Gröbner calculées dans ces chapitres sont rappelées dans le chapitre 1.

Le chapitre 4, consacré à l'analyse asymptotique de l'indice de régularité de la fonction de Hilbert pour des suites semi-régulières, peut être lu de manière indépendante des autres. Le chapitre 5 est lui aussi indépendant : nous montrons comment les résultats des chapitres 3 et 4 pour les systèmes à coefficients et solutions dans le corps fini à deux éléments peuvent s'appliquer à l'analyse de systèmes provenant de problèmes cryptographiques.

Première partie

Préliminaires

Chapitre 1

Rappels sur les bases de Gröbner et les suites régulières

Ce chapitre contient toutes les définitions et propriétés des bases de Gröbner qui seront utilisées dans cette thèse. En particulier, nous récapitulons les propriétés d'élimination et de spécialisation des bases de Gröbner, que nous utiliserons Chapitres 2 et 6 pour le décodage des codes cycliques.

Nous décrivons les liens entre calcul de base de Gröbner et algèbre linéaire. En particulier, nous introduisons une version matricielle, appelée F5-matriciel, de l'algorithme F5 de Jean-Charles Faugère, qui servira de support à l'analyse de complexité des suites semi-régulières au Chapitre 3.

1.1 Introduction

Les bases de Gröbner sont un outil fondamental de l'algèbre commutative pour l'étude de systèmes polynomiaux. Généralisation de la division Euclidienne, de l'algorithme d'Euclide pour le calcul du pgcd, de l'élimination de Gauss (pour des polynômes de degré plus élevé que le degré 1), elles permettent de résoudre de nombreux problèmes concernant les systèmes polynomiaux : appartenance à un idéal, dimension et degré de l'espace des solutions, nombre de solutions dans le cas d'un nombre fini de solutions, calcul de ces solutions, etc.

Dans ce chapitre, nous présentons cet outil et ses propriétés principales, en particulier celles dont nous nous servirons dans cette thèse. Nous énonçons les théorèmes et propriétés sans preuve, pour plus de détails nous renvoyons le lecteur à des livres de référence sur le sujet, comme par exemple [CLO97, BW93].

Résolution d'un système d'équations Une des principales questions concernant les systèmes d'équations polynomiales est la recherche des solutions de ce système. La définition suivante précise la notion de solution d'un système :

Définition 1.1.1 *Soit $\mathbb{L} \supset \mathbb{K}$ une extension de corps. La variété algébrique (affine) associée à un système d'équations $\{f_1, \ldots, f_m\} \subset \mathbb{K}[x_1, \ldots, x_n]$ sur $\mathbb{L}$ est l'ensemble des solutions (ou zéros) de $\{f_1, \ldots, f_m\}$ dans $\mathbb{L}$, c'est-à-dire*

$$V_{\mathbb{L}}(f_1, \ldots, f_m) = \{(z_1, \ldots, z_n) \in \mathbb{L} : f_i(z_1, \ldots, z_n) = 0 \ \forall i = 1 \ldots m\}$$

On appelle ensemble des solutions d'un système la variété $V_{\overline{\mathbb{K}}}(f_1, \ldots, f_m)$ où $\overline{\mathbb{K}}$ est la clôture algébrique de $\mathbb{K}$. Un système est de dimension zéro, ou zéro-dimensionnel, si $V_{\overline{\mathbb{K}}}(f_1, \ldots, f_m)$ est fini.

Nous ne ferons pas de rappels sur les idéaux, les liens entre idéaux et variétés, les notions de dimension, de multiplicité d'une solution, etc. Nous renvoyons le lecteur à des livres généraux comme [CLO97, BW93].

Les bases de Gröbner constituent une première étape vers la résolution d'un système. Une base de Gröbner donne directement des renseignements sur les solutions du système dans $\overline{\mathbb{K}}$: dimension de la variété, nombre de solutions s'il y en a un nombre fini, degré de la variété, etc.

Trouver les solutions dans $\mathbb{L}$ d'un système de polynômes de $\mathbb{K}[x_1, \ldots, x_n]$ avec $\mathbb{L} \supset \mathbb{K}$ une extension de corps distincte de la clôture algébrique de $\mathbb{K}$ constitue un problème qui peut être difficile, comme trouver les solutions réelles d'un système à coefficients rationnels. Dans le cas particulier d'un système à coefficients dans un corps fini, il existe une méthode générale *algébrique* pour trouver les solutions dans $\mathbb{F}_q$ du système : en effet, les éléments du corps fini $\mathbb{F}_q$ sont exactement les solutions du polynôme $x^q - x$. Cette équation s'appelle équation de corps, c'est l'équation qui caractérise l'appartenance au corps $\mathbb{F}_q$. Ainsi, $V_{\mathbb{F}_q}(f_1, \ldots, f_m)$ est exactement l'ensemble des solutions du système de départ auquel on a rajouté les équations de corps. L'idéal considéré devient $I = \langle f_1, \ldots, f_m, x_1^q - x_1, \ldots, x_n^q - x_n \rangle$, et on a donc la relation :

$$V_{\mathbb{F}_q}(f_1, \ldots, f_m) = V_{\overline{\mathbb{F}_q}}(f_1, \ldots, f_m, x_1^q - x_1, \ldots, x_n^q - x_n)$$

De plus, l'idéal I a de bonnes propriétés : grâce aux équations de corps, l'idéal est radical[1] et possède un nombre fini de solutions (cf. lemme de Seidenberg A.2.2 page 144).

Homogénéisation Un polynôme est homogène si tous ses monômes ont même degré. Un idéal homogène est un idéal engendré par des polynômes homogènes. La théorie est très souvent construite pour des idéaux homogènes, pour lesquels les preuves sont plus faciles. Par exemple, pour des polynômes affines, un polynôme de haut degré peut se réduire en un polynôme de bas degré lors d'un calcul de base de Gröbner, et il devient difficile d'estimer par exemple des bornes sur le degré des polynômes. Ces problèmes n'apparaissent pas pour des polynômes homogènes.

Dans le cas général de polynômes affines, il est possible de se ramener à des polynômes homogènes en ajoutant une variable d'homogénéisation.

[1] Un idéal I est radical (ou réduit) si $f^n \in I \Rightarrow f \in I$.

Notations Dans tout ce chapitre, $\mathbb{K}$ désigne un corps quelconque, $\overline{\mathbb{K}}$ sa clôture algébrique et $S_n = \mathbb{K}[x_1, \ldots, x_n]$ l'anneau des polynômes en n variables à coefficients dans $\mathbb{K}$. Dans toutes les applications, nous aurons soit $\mathbb{K} = \mathbb{Q}, \mathbb{R}$ ou $\mathbb{C}$ un corps infini, soit $\mathbb{K} = \mathbb{F}_q$ le corps fini à q éléments.

Pour $s \in \mathbb{N}$ nous notons $(S_n)_s = \{f \in S_n : \deg(f) = s\}$ et $J_s = J \cap (S_n)_s$ si $J \subset S_n$. Nous notons $[i; j]$ l'intervalle des entiers de i à j inclus.

Nous considérerons $f_1, \ldots, f_m \subset S_n$ une suite de m polynômes homogènes en n variables. Nous notons $I = \langle f_1, \ldots, f_m \rangle$ l'idéal homogène engendré par les f_i, et d_i le degré du polynôme f_i.

Plan du chapitre Dans ce chapitre nous rappelons les définitions et propriétés des bases de Gröbner (Section 1.2) et quelques-unes de leurs applications : nous redonnons en particulier Section 1.3 les propriétés d'élimination et de spécialisation qui seront utilisées au chapitre 6 pour le décodage algébrique des codes correcteurs.

Nous introduisons Section 1.6 les séries de Hilbert, qui nous permettent de redonner les propriétés des suites régulières, définies Section 1.7.

La Section 1.8 récapitule les analyses de complexité existantes pour les calculs de bases de Gröbner. Ces analyses de complexité s'appuient sur le lien entre calcul de base de Gröbner et algèbre linéaire, que nous décrivons Section 1.4. Nous donnons Section 1.5 une version matricielle de l'algorithme F5, appelée F5-matriciel, dont nous analyserons la complexité au chapitre 3 pour les suites semi-régulières, généralisant les suites régulières aux systèmes surdéterminés.

1.2 Définitions, premier algorithme

1.2.1 Ordres monomiaux, réduction

Un problème bien connu pour les polynômes en une variable est celui de l'appartenance à un idéal : étant donnés des polynômes $f, g_1, \ldots, g_m \in \mathbb{K}[x]$, décider si f appartient à l'idéal $\langle g_1, \ldots, g_m \rangle$. Ce problème se résout simplement en calculant g le pgcd de $g_1, \ldots, g_m$, puis en effectuant la division euclidienne de f par g. La théorie des bases de Gröbner généralise ce problème aux polynômes multivariés. Cela nécessite de définir la notion d'ordre sur les monômes, généralisant l'ordre naturel par degré croissant pour les polynômes univariés. Un autre problème est de généraliser la notion de division euclidienne, qui devient la réduction d'un polynôme par un ensemble de polynômes.

Définition 1.2.1 *Un ordre monomial admissible sur S_n est une relation d'ordre total $<$ sur l'ensemble des monômes de S_n, vérifiant :*

(i) si $m_1 < m_2$ et m_3 est un monôme alors $m_1 m_3 < m_2 m_3$,

(ii) tout sous ensemble non vide de monômes admet un plus petit élément pour l'ordre $<$.

Le point (ii) peut être remplacé par (ii)' :

(ii)' pour tout monôme m, on a $1 < m$.

Les ordres suivants sont des exemples classiques d'ordres admissibles :
Exemple. L'ordre lexicographique (Lex) avec $x_1 > x_2 > \ldots > x_n$ est défini par $m_1 = x_1^{\alpha_1} \ldots x_n^{\alpha_n} <_{\text{Lex}} m_2 = x_1^{\beta_1} \ldots x_n^{\beta_n}$ si le premier terme non nul dans $(\alpha_1 - \beta_1, \ldots, \alpha_n - \beta_n)$ est négatif.

Le degré d'un monôme $m_1 = x_1^{\alpha_1} \ldots x_n^{\alpha_n}$ est degré$(m_1) = \sum_{i=1}^{n} \alpha_i$.
Exemple. L'ordre du degré inverse lexicographique (grevlex) avec $x_1 > x_2 > \ldots > x_n$ est défini par $m_1 = x_1^{\alpha_1} \ldots x_n^{\alpha_n} <_{\text{grevlex}} m_2 = x_1^{\beta_1} \ldots x_n^{\beta_n}$ si degré$(m_1) <$ degré(m_2) ou degré$(m_1) =$ degré(m_2) et le dernier terme non nul dans $(\alpha_1 - \beta_1, \ldots, \alpha_n - \beta_n)$ est positif.

L'ordre grevlex appartient à la classe des ordres gradués par le degré : on compare d'abord les degrés totaux de deux monômes, et en cas d'égalité on utilise un autre ordre monomial admissible pour décider (par exemple l'inverse de l'ordre Lex pour $x_n > \ldots > x_1$ dans le cas de l'ordre grevlex).
Exemple. Soit $\underline{w}_n = (w_1, \ldots, w_n) \in \mathbb{N}^n$ un n-uplet d'entiers, et $<$ un ordre monomial admissible. L'ordre monomial $<_{\underline{w}_n}$ pondéré par les poids $\underline{w}_n$ est défini par $m_1 = x_1^{\alpha_1} \ldots x_n^{\alpha_n} <_{\underline{w}_n} m_2 = x_1^{\beta_1} \ldots x_n^{\beta_n}$ si $x_1^{w_1\alpha_1} \ldots x_n^{w_n\alpha_n} < x_1^{w_1\beta_1} \ldots x_n^{w_n\beta_n}$.

Nous verrons Section 1.3.2 d'autres exemples d'ordres monomiaux, les ordres d'élimination.

Définition 1.2.2 *Un ordre monomial admissible $<$ étant fixé, soit $p \in S_n$ un polynôme, alors on peut définir les quantités suivantes :*
- *$LM(p)$ le monôme de tête de p (ou Leading Monomial), qui correspond au plus grand monôme apparaissant dans p,*
- *$LC(p)$ le coefficient de tête de p (ou Leading Coefficient), qui correspond au coefficient dans p de son monôme de tête,*
- *$LT(p)$ le terme de tête de p (ou Leading Term), qui vaut $LC(p)LM(p)$,*
- *si $F \subset S_n$ est un ensemble de polynômes, on note $LT(F) = \{LT(f) : f \in F\}$. On définit de même $LM(F)$.*
- *si m_1 et m_2 sont des monômes, on note $ppcm(m_1, m_2)$ le plus petit commun multiple de m_1 et m_2, et $pgcd(m_1, m_2)$ leur plus grand commun diviseur,*
- deg(p) *le degré maximal d'un monôme apparaissant dans p, appelé le degré (ou degré total) de p.*

Il reste maintenant à définir, pour un ordre monomial fixé, la notion de réduction d'un polynôme, qui généralise la division euclidienne.

Définition 1.2.3 *Soit $<$ un ordre admissible, soit $f, g, p \in \mathbb{K}[x_1, \ldots, x_n]$ et $P \subset \mathbb{K}[x_1, \ldots, x_n]$. On définit alors*
- *f se réduit à g modulo p, noté $f \underset{p}{\rightarrow} g$, si il existe un monôme t dans f qui doit divisible par le monôme de tête de p, i.e. il existe deux monômes t et s tels que $g = f - \frac{a}{LC(p)} sp$, $sLM(p) = t$ et $a \neq 0$ soit le coefficient du monôme t dans f,*

- *f se réduit à g modulo P, noté $f \underset{P}{\to} g$, si $\exists p \in P$ tel que $f \underset{p}{\to} g$,*
- *f est réductible modulo p si il existe g tel que $f \underset{p}{\to} g$,*
- *f est réductible modulo P si il existe g tel que $f \underset{P}{\to} g$,*

On note encore $\underset{P}{\to}$ la clôture réflexive et transitive de la relation $\underset{P}{\to}$ (i.e. tout polynôme obtenu après un nombre fini de réductions modulo P). On dit que f est en forme normale modulo P si f n'est pas réductible modulo P.

Une forme normale de f modulo $P = \{f_1, \ldots, f_s\}$ est un polynôme g tel qu'il existe $a_i \in \mathbb{K}[x_1, \ldots, x_n]$ pour $i \in [1; s]$ avec $f = a_1 f_1 + \ldots + a_s f_s + g$ et aucun des termes de tête des f_i ne divise un terme de g, i.e. $f \underset{P}{\to} g$ et g est en forme normale modulo P. O n note alors $g = \overline{f}^P$.

On dit que f est top-réductible modulo P si $\overline{f}^P = g$ et $LT(g) < LT(f)$.

Exemple. Le résultat de la réduction d'un polynôme par un ensemble de polynômes dépend fortement de l'ordre dans lesquels sont faits les calculs, il n'y a pas unicité de la forme normale en général. Considérons par exemple les polynômes $f = x_1^2 x_2$, $g_1 = x_1^2$ et $g_2 = x_1 x_2 - x_2^2$ pour l'ordre Lex $x_1 > x_2$. On obtient $\overline{f}^{g_1} = 0$ alors que $\overline{f}^{g_2} = x_2^3 \neq 0$. Les bases de Gröbner forment des ensembles particuliers par lesquels la forme normale d'un polynôme est unique.

1.2.2 Bases de Gröbner : existence et unicité

Nous pouvons maintenant donner la définition mathématique d'une base de Gröbner.

Définition 1.2.4 (Base de Gröbner) *Soit I un idéal de S_n, $<$ un ordre admissible et $G \subset I$ un sous ensemble fini de I. Alors G est une base de Gröbner de I pour l'ordre $<$ si $\langle LT(G) \rangle = \langle LT(I) \rangle$.*

Théorème 1.2.5 *Tout idéal possède une base de Gröbner pour l'ordre $<$.*

Cette base de Gröbner n'est évidemment pas unique, ainsi si $\{g_1, g_2\}$ est une base de Gröbner d'un idéal I, et si $f \in I$ alors $\{g_1, g_2 + g_1\}$, $\{g_1, g_2, f\}$ et $\{g_1, x g_1, g_2\}$ sont également des bases de Gröbner de I, on peut comme cela construire une infinité de bases de Gröbner pour un idéal donné. En rendant tous les polynômes unitaires et en supprimant successivement les $g \in G$ tels que $\mathrm{LT}(g) \in \langle \mathrm{LT}(G \backslash \{g\}) \rangle$, on obtient une base de Gröbner *minimale*. La définition suivante caractérise l'une de ces bases.

Définition 1.2.6 (Base de Gröbner réduite) *Soit I un idéal de S_n. Un sous ensemble fini $G \subset I$ est une base de Gröbner réduite de I pour l'ordre $<$ si*

- *$LC(g) = 1$ pour tout $g \in G$,*
- *pour tout $g \in G$, aucun monôme de g n'appartient à $\langle LT(G \backslash \{g\}) \rangle$.*

Proposition 1.2.7 *Soit $I \neq \{0\}$ un idéal de S_n, et $<$ un ordre monomial admissible. Alors I possède une unique base de Gröbner réduite G, et G est une base de I. En particulier, $G = \{1\}$ si et seulement si $V_{\overline{\mathbb{K}}}(I) = \emptyset$.*

Dans cette thèse, lorsque nous parlerons de "la" base de Gröbner d'un idéal, il s'agira de la base de Gröbner réduite de cet idéal pour un ordre donné.

Par exemple dans le cas de polynômes univariés, $\mathbb{K}[x]$ est principal donc tout idéal est engendré par un unique élément unitaire, qui forme l'unique élément de la base de Gröbner réduite.

Considérons maintenant l'idéal I engendré par $\{f_1, f_2\} = \{x^3 - 2xy, x^2y - 2y^2 + x\}$. Alors $\{f_1, f_2\}$ n'est pas une base de Gröbner de I, puisque le polynôme $xf_2 - yf_1 = x^2$ appartient à I, mais $x^2 \notin \langle \mathrm{LT}(f_1), \mathrm{LT}(f_2)\rangle = \langle x^3, x^2y\rangle$. Notons que la base de Gröbner réduite dépend de l'ordre monomial utilisé : la base réduite pour l'ordre Lex $x < y$ est $\{x - 2y^2, y^3\}$ alors que pour l'ordre grevlex $x < y$, la base réduite est $\{x^2, xy, y^2 - \frac{1}{2}x\}$ (nous verrons Section 1.2.3 comment calculer une base de Gröbner).

Une des propriétés essentielles des bases de Gröbner est l'unicité de la réduction d'un polynôme par une base de Gröbner :

Proposition 1.2.8 *Soit $G = \{g_1, \ldots, g_t\}$ une base de Gröbner d'un idéal I, et soit $f \in S_n$. Alors il existe un unique $r \in S_n$ vérifiant :*

(i) aucun terme de r n'est divisible par l'un des $LT(g_i)$, $i \in [1; t]$, i.e. r est en forme normale modulo G,

(ii) il existe $g \in I$ tel que $f = g + r$.

r est alors la forme normale de f modulo G, noté $r = \overline{f}^G$.

La proposition suivante caractérise les bases de Gröbner :

Proposition 1.2.9 *Un ensemble fini G est une base de Gröbner d'un idéal I pour l'ordre $<$ si, de manière équivalente :*

(i) $\forall f \in I,\ \overline{f}^G = 0$

(ii) tout $0 \neq f \in I$ est réductible modulo G

(iii) tout $0 \neq f \in I$ est top-réductible modulo G

(iv) $\forall f \in I,\ \exists g \in G\ LT(g) | LT(f)$

La caractérisation (i) fournit un test d'appartenance à un idéal : $f \in I$ si et seulement si $\overline{f}^G = 0$.

1.2.3 Algorithme de Buchberger

Le premier algorithme de calcul de base de Gröbner a été donné par Buchberger dans sa thèse[2] [Buc65]. L'un des principaux outils de cet algorithme est la notion de S-polynôme.

[2] C'est Buchberger qui les a appelées bases de Gröbner en hommage à son directeur de thèse

Définition 1.2.10 *Soit $p, q \in S_n$ deux polynômes. Le S-polynôme de p et q par rapport à l'ordre $<$ est la combinaison*

$$S(p,q) = \frac{ppcm(LM(p), LM(q))}{LT(p)} \cdot p - \frac{ppcm(LM(p), LM(q))}{LT(q)} \cdot q$$

Le degré du S-polynôme est $\deg(S(p,q)) \leq \deg(ppcm(LM(p), LM(q)))$. *Le couple (p,q) est appelé une paire critique, et $S(p,q)$ est le S-polynôme associé à la paire critique.*

La proposition suivante caractérise les bases de Gröbner à l'aide des S-polynômes.

Proposition 1.2.11 *Soit $G = \{f_1, \ldots, f_m\} \in S_n$, $0 \notin G$ et soit $S_{i,j} = S(f_i, f_j)$ les S-polynômes associés pour l'ordre $<$. Alors G est une base de Gröbner de $\langle f_1, \ldots, f_m \rangle$ pour $<$ si et seulement si $\overline{S_{i,j}}^G = 0$ pour tous i, j.*

L'algorithme de Buchberger calculant une base de Gröbner de l'idéal $\langle f_1, \ldots, f_m \rangle$ en découle naturellement, et est reproduit Figure 1.1. Partant de $G = \{f_1, \ldots, f_m\}$, il suffit de calculer tous les S-polynômes possibles, de les réduire par rapport à G, et d'ajouter à G tous les restes non nuls. On recalcule alors tous les nouveaux S-polynômes, on les réduit par rapport à G, et ainsi de suite. La preuve de terminaison de l'algorithme utilise le lemme de Dickson, qui assure qu'il n'existe pas de suite infinie strictement croissante d'idéaux monomiaux. Ce lemme n'est pas effectif, et ne donne pas de preuve de complexité pour l'algorithme de Buchberger.

Entrée : $F = (f_1, \ldots, f_m)$ et $<$ un ordre monomial admissible

Sortie : une base de Gröbner $G = (g_1, \ldots, g_t)$ de I pour l'ordre $<$, avec $F \subset G$

$G := F$
$P :=$ Trier$(\{\{p,q\} |\ p, q \in G,\ p \neq q\}, \text{degré})$
Tant que $P \neq \emptyset$ **Faire**
  $\{p,q\} :=$ premier élément de P
  $P := P \backslash \{\{p,q\}\}$
  $S := \overline{S(p,q)}^G$
  Si $S \neq 0$ **Alors**
    $P :=$ Trier$(P \cup \{\{g,S\} |\ g \in G\}, \text{degré})$
    $G := G \cup \{S\}$
  Fin
Fin
Retourner G

FIG. 1.1 – Algorithme de Buchberger

Au cours de l'algorithme, il arrive que la réduction d'un S-polynôme par la base en cours de construction G donne zéro. On parle alors de *réduction à zéro* dans l'algorithme. De même, lorsqu'on utilise un ordre gradué par le degré, si le degré du S-polynôme une fois réduit est strictement inférieur à celui du S-polynôme, on dit qu'il y a eu une *chute de degré* dans l'algorithme. Les chutes de degré n'apparaissent que lors de calculs avec des polynômes affines, et une chute de degré correspond à une réduction à zéro de la partie homogène de plus haut degré du S-polynôme.

Dans l'algorithme tel que décrit Figure 1.1, l'essentiel du temps est perdu à calculer des réductions à zéro de S-polynômes. Il existe des critères, qui permettent de prédire à l'avance de telles réductions à zéro, et ainsi d'améliorer l'efficacité de l'algorithme (voir Section 1.2.4).

On peut définir la notion de d-base de Gröbner pour des polynômes homogènes comme suit :

Définition 1.2.12 *Soit $I = \langle f_1, \ldots, f_m \rangle$ un idéal homogène. Un ensemble fini $G \subset S_n$ est une d-base de Gröbner de I si G engendre I et que, de manière équivalente :*
- *$\overline{S(g_1, g_2)}^{G} = 0 \; \forall g_1, g_2 \in G$ lorsque $\deg(S(g_1, g_2)) \leq d$,*
- *tout $f \in I_s$ avec $s \leq d$ est top-réductible par G.*

Une d-base de Gröbner est aussi appelée une base de Gröbner jusqu'au degré d.

Notons G_d une base de Gröbner jusqu'au degré d, alors il existe un entier d_0 tel que $G_{d_0-1} \subset G_{d_0} = G_{d_0+1} = \ldots$ que l'on note G_∞. Ainsi, à partir d'un certain degré une d-base est une base de Gröbner, et G_∞ est une base de Gröbner.

Nous avons décrit l'algorithme de Buchberger avec une fonction Trier(.,degré) qui choisit en premier les paires critiques de plus petit degré. Cela revient à appliquer la stratégie Normale [Buc65, BW93], qui semble être plutôt efficace en général. L'algorithme de Buchberger se transforme alors aisément en un algorithme qui calcule une d-base de Gröbner : il suffit de ne considérer que les S-polynômes de degré inférieur ou égal à d.

1.2.4 Stratégies de calcul

Il existe de nombreuses stratégies pour calculer une base de Gröbner, mais les coûts du calcul (en temps et en espace) varient énormément d'une stratégie à l'autre. Nous décrivons dans cette section quelques "bonnes" stratégies, qui heuristiquement donnent des calculs plus efficaces dans la plupart des cas.

Choix de l'ordre monomial Il est souvent plus efficace de ne pas calculer directement la base de Gröbner pour l'ordre voulu, mais de faire un calcul intermédiaire pour un autre ordre "plus faible" et d'utiliser ensuite un algorithme de changement d'ordre comme FGLM [FGLM93] (en dimension zéro) ou Gröbner Walk [CKM97] (qui reste valable en dimension positive).

Si le nombre de solutions de l'idéal est fini, il est relativement facile de calculer numériquement ces solutions à partir d'une base de Gröbner de l'idéal pour l'ordre

Lex (voir Section 1.3.1, [CLO97, p. 95] ou [Frö97, p. 84]). Cependant la base Lex est en général difficile à calculer.

L'ordre grevlex semble être le mieux adapté pour le calcul de la base de Gröbner : on le constate aisément expérimentalement, c'est pour cet ordre que les calculs sont les plus rapides. C'est aussi l'ordre pour lequel on obtient les meilleurs bornes de complexité (voir [Laz83] et la section 3.4 de cette thèse) : le degré maximal des polynômes intervenant au cours du calcul est plus petit pour l'ordre grevlex que pour n'importe quel autre ordre monomial admissible (le cas le pire étant celui de l'ordre Lex [Laz83]).

Dans le cas d'un idéal ayant un nombre fini de solutions D (comptées avec multiplicité), une base de Gröbner pour l'ordre Lex peut se calculer à partir d'une base de Gröbner pour n'importe quel autre ordre (en particulier l'ordre grevlex) en $O(nD^3)$ opérations arithmétiques [FGLM93] (n étant le nombre de variables). Pour calculer une base de Gröbner pour l'ordre Lex, on calcule en premier une base pour l'ordre grevlex, puis on utilise un algorithme de changement d'ordre.

Choix des paires critiques Une fois l'ordre monomial fixé, il reste de nombreux choix à faire au cours du calcul, en particulier celui de l'ordre de sélection des paires critiques. Il n'existe pas de bonne stratégie de sélection, mais des critères heuristiques mettent en évidence des stratégies plus efficaces que d'autres en pratique.

La stratégie Normale [Buc65, BW93] semble être plutôt efficace. Les calculs sont faits en sélectionnant en premier les paires critiques de plus petit degré (les choix entre deux paires critiques de même degré sont encore une fois heuristiques). Cette stratégie est particulièrement efficace pour n'importe quel ordre monomial gradué par le degré. Enfin, la stratégie du sucre est une autre stratégie particulière [GMN+91].

Lorsqu'on utilise une stratégie Normale dans l'algorithme F4 de Jean-Charles Faugère [Fau99], l'algèbre linéaire permet de décider entre les paires critiques d'un même degré. Le lien entre algèbre linéaire et calcul de base de Gröbner est explicité dans la section 1.4.

Homogénéisation Il est souvent intéressant, du point de vue théorique, de travailler avec des systèmes homogènes : lorsqu'on calcule une base de Gröbner avec la stratégie Normale, le degré des polynômes intervenant au cours du calcul est croissant, et en supprimant toutes les paires critiques de degré plus grand que d on obtient une d-base de Gröbner.

Dans le cas général, on homogénéise un système en rajoutant une variable d'homogénéisation h, et en considérant les polynômes $f_i^h = h^{\deg(f_i)} f_i(\frac{x_1}{h}, \ldots, \frac{x_n}{h})$. L'inconvénient de cette méthode est qu'elle fait apparaître des solutions "à l'infini", c'est-à-dire les solutions vérifiant $h = 0$, qui ne sont pas solutions du système de départ.

Dans le cas des systèmes provenant de la théorie des codes correcteurs, que nous rencontrerons dans le chapitre 6, les équations sont de la forme $S_i + \sum_{j=1}^{i-1} \sigma_j S_{i-j} + i\sigma_i$

(relations de Newton) ou $S_i + \sum_{j=1}^{v} Z_j^i$ (fonctions puissances), et ne sont donc pas homogènes. Par contre, si l'on utilise un ordre monomial pondéré (voir page 6), en donnant un poids i aux variables S_i et σ_i, elles deviennent homogènes, et l'ensemble des solutions du système est inchangé (il n'y a pas de solutions parasites). Le calcul de la base de Gröbner d'idéaux engendrés par des équations de ce type est souvent bien plus efficace avec un ordre monomial pondéré qu'avec un ordre classique.

Partant d'un système affine, on peut également considéré le système homogène formé des parties homogènes de plus haut degré des polynômes. L'ensemble des solutions de ce nouveau système n'a plus rien à voir avec les solutions du système de départ, mais nous verrons Section 3.5 que la complexité du calcul de la base de Gröbner pour ces deux systèmes est reliée.

Critères évitant les réductions à zéro D'un point de vue pratique, lors d'un calcul de base de Gröbner une grande partie du temps est perdue à calculer zéro, en calculant la réduction de S-polynômes à zéro. Il s'avère que pour certains S-polynômes, il est possible de détecter de manière combinatoire et rapide s'ils vont se réduire à zéro (et les tests de ces critères sont bien plus rapides que le calcul effectif de la réduction à zéro).

Buchberger à donné trois critères, qui permettent d'éviter un certain nombre de réductions à zéro [Buc65], [CLO97, §9] : ainsi, il est inutile de considérer le S-polynôme de deux polynômes ayant des termes de tête étrangers (premier critère de Buchberger). Le second critère de Buchberger dit que, si p, f, g sont des éléments de la base de Gröbner courante, tels que $\mathrm{LM}(p)$ divise $\mathrm{ppcm}(\mathrm{LM}(f), \mathrm{LM}(g))$, et si les paires critiques (f, p) et (g, p) ont déjà été traitées, alors il est inutile de traiter la paire (f, g). Le troisième critère est plus complexe et ne sera pas décrit ici.

Mais ces critères n'évitent pas toutes les réductions à zéro, et il en subsiste même beaucoup dans certains cas. Récemment, Jean-Charles Faugère à proposé un nouvel algorithme, F5 [Fau02], qui contient un critère évitant toutes les réductions à zéro combinatoires provenant des relations $f_i f_j = f_j f_i$ (et ce sont les seules réductions à zéro dans le cas ou la suite $f_1, \ldots, f_m$ est régulière, notion qui sera définie Section 1.7). Nous donnons Section 1.5 une version matricielle de l'algorithme F5, contenant également un nouveau critère dans le cas de systèmes sur $\mathbb{F}_2$ contenant les équations de corps.

1.3 Applications des bases de Gröbner

1.3.1 Résolution de systèmes d'équations polynomiales

Un système de $m = n$ équations en n variables possède génériquement un nombre fini de solutions, borné par la borne de Bezout. Dès que $m > n$, un système de m équations n'a génériquement plus de solution, mais dans de nombreuses applications on a besoin de résoudre des systèmes fortement surdéterminés ayant au moins une

solution (en cryptographie, codes correcteurs, ...).

Dans cette thèse, nous nous intéresserons essentiellement à des systèmes surdéterminés de dimension zéro. Nous ne présenterons donc pas de méthodes de résolution pour des systèmes ayant une infinité de solutions.

Systèmes de dimension zéro Une base de Gröbner donne de nombreux renseignements sur les solutions d'un système polynômial $f_1 = 0, \ldots, f_m = 0$. En particulier, le système possède un nombre fini de solutions si et seulement si la base de Gröbner de $\{f_1, \ldots, f_m\}$ vérifie la propriété suivante : pour toute variable x_i, $i \in [1; n]$, il existe dans la base de Gröbner un polynôme dont le terme de tête est une puissance de x_i.

Supposons maintenant que le système d'équations $f_1 = 0, \ldots, f_m = 0$ possède un nombre fini de solutions. Alors le nombre de solutions (dans la clôture algébrique $\overline{\mathbb{K}}$ de $\mathbb{K}$, comptées avec multiplicité et en incluant les solutions à l'infini) est le cardinal de l'ensemble des monômes qui ne sont pas multiples des monômes de tête des polynômes de la base de Gröbner. On sait aussi que dans ce cas, la base de Gröbner pour l'ordre Lex $x_1 > \ldots > x_n$ a la forme triangulaire de la Figure 1.2.

$$
\begin{array}{llllll}
g_1(x_1, & x_2, & x_3, & \ldots, & x_n) \\
\vdots & & & & \\
g_p(x_1, & x_2, & x_3, & \ldots, & x_n) \\
 & g_{p+1}(x_2, & x_3, & \ldots, & x_n) \\
 & \vdots & & & \\
 & g_q(x_2, & x_3, & \ldots, & x_n) \\
 & & g_{q+1}(x_3, & \ldots, & x_n) \\
 & & \vdots & & \\
 & & & \vdots & \\
 & & & & g_r(x_n)
\end{array}
$$

FIG. 1.2 – Forme générale d'une base Lexicographique

La résolution d'un tel système équation par équation donne l'ensemble des solutions du système dans la clôture algébrique du corps de base.

Sous certaines hypothèses (idéal radical et à un changement de coordonnées près, [GM89, BMMT94, Shape Lemma]), la base de Gröbner de $\{f_1, \ldots, f_m\}$ pour l'ordre Lex $x_1 > \ldots > x_n$ a la structure simplifiée suivante :

$$\{x_1 - g_1(x_n), x_2 - g_2(x_n), \ldots, x_{n-1} - g_{n-1}(x_n), g_n(x_n)\} \tag{1.1}$$

où chaque g_i est un polynôme en une seule variable. Ces hypothèses sont vérifiées par la plupart des systèmes, et dans le cas général, un système de dimension zéro est équivalent à plusieurs systèmes de la forme 1.1.

Systèmes à coefficients et solutions dans un corps fini Si l'on s'intéresse à la résolution dans $\mathbb{F}_2$ d'un système d'équations à coefficients dans $\mathbb{F}_2$, nous avons vu qu'il suffit d'ajouter au système les équations de corps $x_1^2 + x_1, \ldots, x_n^2 + x_n$. On a alors les caractérisations suivantes :

Proposition 1.3.1 *Soit $F = \{f_1, \ldots, f_m\} \subset \mathbb{F}_2[x_1, \ldots, x_n]$, et G la base de Gröbner réduite de l'idéal $I = \langle f_1, \ldots, f_m, x_1^2+x_1, \ldots, x_n^2+x_n\rangle$ pour un ordre quelconque. Alors*

- *l'idéal I est radical, i.e. toutes ses solutions sont de multiplicité 1,*
- *$G = \{1\}$ si et seulement si le système F n'a pas de solutions dans $\mathbb{F}_2$,*
- *$G = \{x_1 - a_1, \ldots, x_n - a_n\}$ si et seulement si le système F a une unique solution $(a_1, \ldots, a_n)$ dans $\mathbb{F}_2$.*

Ces résultats restent valable dans tout corps fini $\mathbb{F}_q$, en ajoutant les équations de corps $x_1^q - x_1, \ldots, x_n^q - x_n$.

1.3.2 Élimination, Spécialisation des bases de Gröbner

Cette section est consacrée à l'utilisation des bases de Gröbner pour l'élimination de variables d'un idéal (ou de manière équivalente pour le calcul de la projection d'une variété), et à leur comportement lors d'une spécialisation (on affecte une valeur particulière à une variable). Ces propriétés seront beaucoup utilisées, en particulier au chapitre 6 pour le décodage de codes correcteurs.

Définition 1.3.2 *Un ordre d'élimination des variables $x_1, \ldots, x_n$ est un ordre monomial admissible sur $\mathbb{K}[x_1, \ldots, x_n, y_1, \ldots, y_m]$ tel que pour tous monômes $x^{\alpha_1}y^{\beta_1}$, $x^{\alpha_2}y^{\beta_2}$ on ait $x^{\alpha_1} > x^{\alpha_2} \rightarrow x^{\alpha_1}y^{\beta_1} > x^{\alpha_2}y^{\beta_2}$. Un tel ordre est aussi appelé ordre par blocs $x_1, \ldots, x_n \gg y_1, \ldots, y_m$.*

Exemple. L'ordre $>_{(\text{Lex},\text{grevlex})}$ pour $x_1 > \ldots > x_n \gg y_1 > \ldots > y_m$ mélange l'ordre lexicographique pour le premier bloc de variables $x_1 > \ldots > x_n$ et l'ordre grevlex pour les variables $y_1 > \ldots > y_m$. Il est défini par $x^{\alpha_1}y^{\beta_1} >_{(\text{Lex},\text{grevlex})} x^{\alpha_2}y^{\beta_2}$ si $x^{\alpha_1} >_{\text{Lex}} x^{\alpha_2}$ ou $x^{\alpha_1} = x^{\alpha_2}$ et $y^{\beta_1} >_{\text{grevlex}} y^{\beta_2}$. Cet ordre d'élimination sera utilisé au chapitre 6 pour traiter des systèmes à paramètres.

Il est possible de la même manière de mélanger deux blocs de variables avec l'ordre grevlex pour chacun des deux blocs : on définit l'ordre $>_{(\text{grevlex},\text{grevlex})}$ par $x^{\alpha_1}y^{\beta_1} >_{(\text{grevlex},\text{grevlex})} x^{\alpha_2}y^{\beta_2}$ si $x^{\alpha_1} >_{\text{grevlex}} x^{\alpha_2}$ ou $x^{\alpha_1} = x^{\alpha_2}$ et $y^{\beta_1} >_{\text{grevlex}} y^{\beta_2}$.

Exemple. L'ordre d'élimination de Bayer et Stillman est défini par : $x^{\alpha_1}y^{\beta_1} > x^{\alpha_2}y^{\beta_2}$ si $\deg(x^{\alpha_1}) > \deg(x^{\alpha_2})$ ou $\deg(x^{\alpha_1}) = \deg(x^{\alpha_2})$ et $x^{\alpha_1}y^{\beta_1} >_{\text{grevlex}} x^{\alpha_2}y^{\beta_2}$.

Une propriété essentielle de ces ordres d'élimination est la possibilité d'éliminer des variables d'un système d'équations :

Theorem 1.3.3 (Théorème d'élimination, [CLO97, p. 113]) *Soit I un idéal de $\mathbb{K}[x_1, \ldots, x_n, y_1, \ldots, y_m]$ et G sa base de Gröbner pour un ordre d'élimination $x_1, \ldots, x_n \gg y_1, \ldots, y_m$. Alors l'ensemble*

$$G_n = G \cap \mathbb{K}[y_1, \ldots, y_m]$$

est une base de Gröbner de l'idéal d'élimination $I_n = I \cap \mathbb{K}[y_1, \ldots, y_m]$.

D'un point de vue géométrique, ce théorème d'élimination correspond à un théorème de projection pour les variétés associées : éliminer le premier bloc de variables revient à projeter les variétés associées sur le sous-espace constitué du deuxième bloc de variables.

Par exemple, l'ordre Lex est un ordre d'élimination successive de chaque variable. C'est pour cela que cet ordre est bien adapté au calcul des solutions d'un système de dimension zéro.

Un autre exemple d'utilisation d'un ordre d'élimination est le calcul de l'idéal des relations d'un système de polynômes (ce théorème sera utilisé au chapitre 6) :

Théorème 1.3.4 ([CLO97, chap. 7 §4]) *Soit $\{f_1, \ldots, f_m\} \subset \mathbb{K}[x_1, \ldots, x_n]$. On considère l'idéal $I = \langle f_1 - y_1, \ldots, f_m - y_m \rangle \subset \mathbb{K}[x_1, \ldots, x_n, y_1, \ldots, y_m]$. Alors $J = I \cap \mathbb{K}[y_1, \ldots, y_m]$ est un idéal premier, c'est l'idéal de toutes les relations entre les f_i.*

Ces ordres d'élimination sont très utiles lorsque l'on veut par exemple traiter de systèmes à paramètres : un bloc de variables constitue en fait des paramètres, et l'on cherche à exprimer les autres variables en fonction de ces paramètres. Un autre problème intéressant de ces systèmes à paramètres concerne la spécialisation : étant donné une base de Gröbner G d'un idéal $I \subset \mathbb{K}[x_1, \ldots, x_n, y_1, \ldots, y_m]$, si l'on substitue des valeurs $y_1^*, \ldots, y_m^* \in \mathbb{L}$ (où $\mathbb{L}$ est une extension du corps $\mathbb{K}$) aux variables $y_1, \ldots, y_m$, que peut-on dire du système $G^* = G(y_i = y_i^*)$? Il est évident que ce système engendre l'idéal $I^* = I(y_i = y_i^*)$, mais à quelles conditions G^* est-il une base de Gröbner de I^* ?

Nous donnons ci-dessous trois cas particuliers pour lesquels cette propriété est vraie. Le premier concerne l'homogénéisation d'un système : si nous notons $f^h = h^{\deg(f)} f(\frac{x_1}{h}, \ldots, \frac{x_n}{h})$ l'homogénéisation d'un polynôme f, et $f_a = f(x_1, \ldots, x_n, 1)$ la déshomogénéisation d'un polynôme f, alors la proposition suivante est vérifiée :

Proposition 1.3.5 ([BW93]) *Soit $F \subset \mathbb{K}[x_1, \ldots, x_n]$, et G une base de Gröbner du système homogénéisé $F^h = \{f^h : f \in F\} \subset \mathbb{K}[x_1, \ldots, x_n, h]$. Alors $G_a = \{g_a : g \in G\}$ est une base de Gröbner de F.*

Dans les deux autres cas toutes les variables sauf une sont spécialisées. Si l'idéal est de dimension zéro, le théorème est vrai et il n'y a pas de chute de degré dans la base de Gröbner lors de la spécialisation, dans le cas général il peut y avoir une chute de degré mais elle est bien contrôlée.

On appelle variable principale d'un polynôme P pour un ordre donné la plus grande variable (pour cet ordre) qui apparaît dans ce polynôme, et l'initial de P est son coefficient dominant en tant que polynôme en sa variable principale.

Pour $\underline{y}_n^* = (y_1^*, \ldots, y_n^*) \in \mathbb{L}^n$ (où $\mathbb{L} \supset \mathbb{K}$ est une extension de corps) notons $\phi_{\underline{y}_n^*}$ la fonction de spécialisation des variables $\underline{y}_n$:

$$\begin{array}{rcl} \phi_{\underline{y}_n^*} : \mathbb{K}[x, y_1, \ldots, y_n] & \to & \mathbb{L}[x] \\ p(x, \underline{y}_n) & \mapsto & \phi_{\underline{y}_n^*}(p) = p(x, \underline{y}_n^*) \end{array}$$

Théorème 1.3.6 (Gianni [Gia89]) *Soit $I \subset \mathbb{K}[x, y_1, \ldots, y_n]$ un idéal de dimension zéro, et G une base de Gröbner de I pour un ordre d'élimination de la variable x. Soit $(y_1^*, \ldots, y_n^*) \in \mathbb{L}^n$ et $I^* = \phi_{\underline{y}_n^*}(I) \subset \mathbb{L}[x]$.*
Notons g un polynôme de G de plus petit degré en x dont l'initial ne s'annule pas en $(y_1^*, \ldots, y_n^*)$ (un tel g existe). Alors $\phi_{\underline{y}_n^*}(g) \in \mathbb{L}[x]$ engendre I^* (ce qui implique que $G^* = \phi_{\underline{y}_n^*}(G)$ est une base de Gröbner de I^*).

Théorème 1.3.7 (Fortuna-Gianni-Trager[FGT01]) *Soit $I \subset \mathbb{K}[x, y_1, \ldots, y_n]$ un idéal, et G une base de Gröbner de I pour un ordre d'élimination de la variable x. Soit $(y_1^*, \ldots, y_n^*) \in \mathbb{L}^n$ et $I^* = \phi_{\underline{y}_n^*}(I)$. Pour $p \in I$ on définit $\delta(p) = \deg_x(p) - \deg(\phi_{\underline{y}_n^*}(p))$ la chute de degré et $\delta_I = \min\{\delta(p),\ p \in I\}$.*

Notons g un polynôme de G de plus petit degré en x tel que $\phi_{\underline{y}_n^*}(g) \neq 0$, alors

- *$\phi_{\underline{y}_n^*}(g)$ engendre I^* (ce qui implique que $G^* = \phi_{\underline{y}_n^*}(G)$ est une base de Gröbner de I^*), et $\delta_I = \delta(g)$ (i.e. la chute de degré de g est minimale),*
- *s'il existe $h \in I$ dont le terme de tête ne s'annule pas par $\phi_{\underline{y}_n^*}$, alors $\deg_x(g) = \deg(\phi_{\underline{y}_n^*}(g))$ et $LT(\phi_{\underline{y}_n^*}(I)) = \phi_{\underline{y}_n^*}(LT_x(I))$*

1.4 Bases de Gröbner et algèbre linéaire

Le but de cette section est de décrire le lien entre un calcul de base de Gröbner et l'algèbre linéaire. Nous nous limitons dans un premier temps au cas de polynômes homogènes $f_1, \ldots, f_m \in S_n = \mathbb{K}[x_1, \ldots, x_n]$, f_i de degré d_i. Fixons $<$ un ordre admissible gradué, et notons $I = \langle f_1, \ldots, f_m \rangle$ l'idéal engendré par les f_i. Pour tout entier d, nous rappelons que $I_d = \{f \in I;\ \deg(f) = d\}$ est muni d'une structure d'espace vectoriel (c'est un sous-espace vectoriel de $(S_n)_d$ l'ensemble des polynômes de degré d, qui possède lui-même comme base l'ensemble des monômes de degré d). L'espace vectoriel I_d est de dimension finie, notée $\dim(I_d)$, et $0 \leq \dim(I_d) \leq \dim((S_n)_d) = \binom{n+d-1}{d}$.

1.4.1 Matrice de Macaulay

L'idée de Macaulay [Mac16] est de représenter l'espace vectoriel I_d via une matrice, dont les colonnes sont indexées par les monômes de S_n de degré d, et les lignes

par les multiples par un monôme des m polynômes engendrant I, ces multiples étant de degré d. L'espace vectoriel I_d est ainsi représenté par un tableau de coefficients, et cette représentation matricielle est utilisée pour ramener les opérations faites dans l'idéal à des opérations d'algèbre linéaire sur une matrice. Cette matrice généralise la matrice de Sylvester pour calculer le résultant de deux polynômes en une variable.

Revenons plus précisément sur la construction de la matrice de Macaulay en degré d, notée $\mathcal{M}_{d,m}^{\text{acaulay}}$, qui est décrite dans [Mac02]. Les colonnes de la matrice correspondent aux monômes de degré d, il y a donc $\mu = \binom{n+d-1}{d}$ monômes, notés $\omega_1^{(d)}, \ldots, \omega_\mu^{(d)}$. Pour chaque f_j, $j \in [1; m]$ considérons tous les produits tf_j de degré d, avec t un monôme de degré $d - d_j$. Ces polynômes particuliers de l'idéal sont appelés polynômes élémentaires[3] par Macaulay, ils engendrent l'espace vectoriel I_d.

Chaque polynôme élémentaire est associé à une ligne de la matrice : pour un produit tf_j, il suffit d'écrire dans la colonne correspondant à un monôme $\omega_i^{(d)}$ le coefficient du polynôme tf_j en ce monôme. Chaque coefficient non nul de la matrice est l'un des coefficients d'un polynôme f_i. Pour chaque ligne, le monôme indexant la première colonne non nulle est le monôme de tête de la ligne, c'est exactement le monôme de tête du polynôme correspondant. Chaque ligne de la matrice associée

$$\mathcal{M}_{d,m}^{\text{acaulay}} = \begin{array}{c} \\ \vdots \\ tf_j \\ \vdots \end{array} \begin{array}{c} \text{monômes } \omega_i^{(d)} \text{ de degré } d \\ \begin{pmatrix} \\ \cdots \text{coeff}(tf_j, \omega_i^{(d)}) \cdots \\ \\ \end{pmatrix} \end{array}$$

au polynôme f_i contient les mêmes éléments (les coefficients de f_i et des zéros) mais dans des colonnes différentes.

Tout polynôme de degré d de l'idéal I peut s'écrire $f = g_1 f_1 + \ldots + g_m f_m$, et est donc une combinaison linéaire de polynômes élémentaires de degré d : $f = \lambda_1\omega_1 f_1 + \lambda_2\omega_2 f_1 + \ldots + \lambda_p\omega_p f_i + \ldots + \lambda_\rho\omega_\rho f_m$. Le polynôme f est donc représenté par une combinaison linéaire des lignes de la matrice.

Macaulay utilise ces matrices pour définir le Résultant de n polynômes homogènes $F_1, \ldots, F_n$ en n variables, qui permet de résoudre le système d'équations $F_1 = F_2 = \cdots = F_n = 0$. Il suppose "each polynomial being complete in all its terms with literal coefficients, all different", ce que l'on traduirait maintenant par des polynômes génériques au sens de la définition suivante.

Définition 1.4.1 *Un polynôme F homogène de degré d est générique s'il s'écrit $F = \sum_{i_1+\ldots+i_n=d} U_{i_1,\ldots,i_n} x_1^{i_1} \cdots x_n^{i_n}$ avec $U_{i_1,\ldots,i_n}$ des variables. C'est un polynôme à coefficient dans $\mathbb{K}[\{U_{i_1,\ldots,i_n}\}_{i_1+\ldots+i_n=d}]$.*

Le Résultant est défini dans le cas général pour des polynômes génériques, le

[3] Ces polynômes élémentaires dépendent de la base $(f_1, \ldots, f_m)$ de I

résultant de n polynômes donnés en n variables étant le résultat de la spécialisation du résultant générique pour ces polynômes particuliers.

Définition 1.4.2 ([Mac02]) *Soit $F_1, \ldots, F_n$ des polynômes homogènes en n variables, de degrés $d_1, \ldots, d_n$ respectivement, chaque polynôme F_i étant générique. Le* Résultant *R de $F_1, \ldots, F_n$ est le plus grand facteur commun des déterminants de la matrice de Macaulay en degré $d = \sum_{i=1}^{n}(d_i - 1) + 1$, i.e. de la matrice des coefficients des polynômes élémentaires de $\langle F_1, \ldots, F_n \rangle$ pour le degré d.*

Proposition 1.4.3 ([Mac02]) *Le Résultant R est un polynôme homogène, irréductible (i.e. il ne peut pas s'écrire comme le produit de deux polynômes non triviaux en les coefficients des F_i) et de degré $D_i = d_1 d_2 \cdots d_n / d_i$ en les coefficients de F_i ($i \in [1; n]$).*

Une condition nécessaire et suffisante pour que le système $F_1 = F_2 = \ldots = F_n = 0$ ait une solution non triviale est l'annulation de R.

Nous voyons ici que pour des polynômes génériques, la matrice de Macaulay intéressante pour la résolution du système est celle en degré $d = \sum_{i=1}^{n}(d_i - 1) + 1 = d_1 + d_2 + \ldots + d_n - n + 1$. Cette borne s'appelle la borne de Macaulay, nous verrons qu'elle est une borne de complexité intrinsèque à l'idéal (calcul d'une base de Gröbner, du résultant, indice de régularité, etc.).

1.4.2 Algorithme de Lazard, complexité

Il est possible d'effectuer un calcul de base de Gröbner en appliquant un algorithme d'élimination de Gauss à la matrice de Macaulay [Laz83, Laz01]. En effet, notons $\tilde{\mathcal{M}}_{d,m}^{\text{acaulay}}$ la matrice obtenue à partir de $\mathcal{M}_{d,m}^{\text{acaulay}}$ après application d'un algorithme de Gauss tel que les seules opérations élémentaires autorisées soient l'addition d'une ligne et d'une combinaison linéaires des précédentes. Les lignes de $\tilde{\mathcal{M}}_{d,m}^{\text{acaulay}}$ conservent le mêmes ordre que dans $\mathcal{M}_{d,m}^{\text{acaulay}}$ (voir la notion d'étiquette d'une ligne Section 1.5.1). Considérons l'ensemble des polynômes correspondant à une ligne de $\tilde{\mathcal{M}}_{d,m}^{\text{acaulay}}$ dont le terme de tête n'est pas le même que celui de la ligne correspondante dans $\mathcal{M}_{d,m}^{\text{acaulay}}$, pour tout $d \leq D$. Alors cet ensemble de polynômes est une base de Gröbner de $\langle f_1, \ldots, f_m \rangle$ jusqu'au degré D, et pour D suffisamment grand c'est une base de Gröbner de $\langle f_1, \ldots, f_m \rangle$.

Du point de vue de la complexité, si $D_{\max}$ est le degré maximal d'un polynôme apparaissant au cours du calcul, et $N_{D_{\max}}$ la taille de la plus grande matrice $\mathcal{M}_{D_{\max},m}$, alors la complexité globale du calcul de la base de Gröbner est dominée par le coût de l'algèbre linéaire sur cette matrice, qui peut être estimé à $N_{D_{\max}}^{\omega}$ où ω est le coefficient de la complexité de l'algèbre linéaire. La meilleur borne connue est $\omega = 2.376$ [CW90]. Cependant, les matrices que nous considérons sont très creuses : pour des polynômes quadratiques f_i, il y a au plus $\frac{n(n-1)}{2}$ coefficients non nuls par lignes, et considérer $\omega = 2$ n'est pas déraisonnable.

Cependant, la taille de la matrice de Macaulay en degré $D_{\max}$ est bien plus grande que son rang, en effet de nombreuses lignes valent zéro dans la matrice réduite $\tilde{\mathcal{M}}^{\text{acaulay}}_{D_{\max},m}$. La section suivante présente une version matricielle de l'algorithme F5, qui utilise les critères de l'algorithme F5 pour réduire considérablement la taille de la matrice utilisée. Nous verrons Chapitre 3 la définition des suites semi-régulières, qui sont exactement les suites pour lesquelles les matrices apparaissant dans l'algorithme F5-matriciel seront de rang plein.

1.4.3 Cas affine

Si les polynômes $f_1, \ldots, f_m$ sont affines, les colonnes de la matrice de Macaulay $\mathcal{M}^{\text{acaulay}}_{d,m}$ en degré d sont indexées par tous les monômes de S_n de degré $\leq d$, les lignes représentant toujours tous les multiples des polynômes f_i $(1 \leq i \leq m)$ par des monômes tels que le produit soit de degré exactement d. Pour calculer une base de Gröbner de $I = \langle f_1, \ldots, f_m \rangle$, il suffit comme précédemment de calculer une forme Échelon des matrices $\mathcal{M}^{\text{acaulay}}_{d,m}$ pour tout $d \leq D$ et D suffisamment grand, en ajoutant une opération élémentaire : à une ligne de $\mathcal{M}^{\text{acaulay}}_{d,m}$ correspondant à un polynôme f_i dont le terme de tête est de degré $d' < d$ on peut ajouter une combinaison linéaire des lignes de $\mathcal{M}^{\text{acaulay}}_{d',i}$. Cela revient à homogénéiser le système et à calculer une forme Echelon de la matrice de Macaulay des polynômes homogènes en degré D.

1.5 Versions matricielles de l'algorithme F5

Dans cette section, nous décrivons un algorithme, appelé *F5-matriciel* , qui étant donné une suite $f_1, \ldots, f_m$ de polynômes homogènes de degrés $d_1 \leq d_2 \leq \ldots \leq d_m$ et un degré $d_{\max}$, calcule une base de Gröbner de $I = \langle f_1, \ldots, f_m \rangle$ jusqu'au degré $d_{\max}$ en utilisant de l'algèbre linéaire. Cet algorithme est une transcription simple de l'algorithme F5 [Fau02] en version matricielle, et utilise le même critère pour éviter les réductions à zéro provenant des relations $f_i f_j = f_j f_i$.

Notre algorithme fonctionne pour toute suite de polynômes homogènes, mais nous en décrivons une version simple : par exemple, nous évitons de maintenir à jour une liste de paires critiques, comme dans la plupart des algorithmes de calcul de base de Gröbner, ce qui impose l'utilisation d'un degré maximal en entrée de l'algorithme (sans liste de paires critiques l'algorithme n'a plus de critère d'arrêt). L'algorithme est également adapté aux systèmes denses : le nombre de colonnes de la matrice en degré d est exactement le nombre de monômes de degré d.

Dans le cas de polynômes affines, nous appliquerons l'algorithme après homogénéisation. Nous donnons Section 1.5.2 un nouveau critère dans le cas où l'idéal contient en plus les équations de corps $x_1^2 - x_1 h, \ldots, x_n^2 - x_n h$, évitant les réductions à zéro provenant des relations $f_i^2 = f_i h^{\deg(f_i)}$ où h est la variable d'homogénéisation (et $h < x_i$ pour tout i). Ce critère est identique si l'idéal contient les équations

$x_1^2, \ldots, x_n^2$, il évite les réductions à zéro provenant des relations $f_i^2 = 0$.

1.5.1 Description de l'algorithme F5-matriciel et du critère

L'idée de [Fau02] est de construire une sous-matrice $\mathcal{M}_{d,m}$ de la matrice de Macaulay en retirant les lignes se réduisant à zéro à cause des relations $f_i f_j = f_j f_i$. Cette matrice sera de rang plein pour des suites régulières (voir Section 1.7).

Les lignes susceptibles d'être retirées sont caractérisées au moyen d'un *critère*. Pour le décrire, nous ordonnons les colonnes de $\mathcal{M}_{d,m}^{\text{acaulay}}$ suivant l'ordre $<$, et nous

$$\mathcal{M}_{d,m}^{\text{acaulay}} = \begin{array}{c} \\ (t, f_1) \\ (u, f_2) \\ \vdots \\ (v, f_m) \end{array} \begin{array}{c} \text{monômes } \omega_i^{(d)} \text{ de degré } d \\ \begin{pmatrix} \\ \text{coeff}(u f_2, \omega_i^{(d)}) \\ \\ \end{pmatrix} \end{array}$$

étiquetons et ordonnons les lignes : la ligne tf_j est étiquetée (t, f_j) et précède la ligne (u, f_i) si $j < i$ ou $(j = i$ et $t < u)$. Avec cette convention, à cause des relations $f_i f_j = f_j f_i$ pour tout $1 \leq i < j \leq m$, une ligne d'étiquette (t, f_j) est combinaison linéaire des précédentes si t est le terme de tête d'un élément de $\langle f_1, \ldots, f_{j-1} \rangle$.

Pour appliquer ce critère, nous avons besoin de construire les matrices $\tilde{\mathcal{M}}_{d,m}$ qui sont le résultat de l'application d'un algorithme d'élimination de Gauss sur $\mathcal{M}_{d,m}$, dans lequel les seules opérations élémentaires permises pour la $i^{\text{ème}}$ ligne sont : $\text{ligne}_i \leftarrow c \times \text{ligne}_i + c' \times \text{ligne}_{i-j}$ avec $j > 0$ et $c \neq 0 \in \mathbb{K}, c' \in \mathbb{K}$.

Le critère s'exprime alors comme suit :

Proposition 1.5.1 (Critère général) *[Fau02] : pour tout $j < m$, si une ligne d'étiquette (t, f_j) dans la matrice $\tilde{\mathcal{M}}_{d-d_m,m-1}$ a pour terme de tête t', alors la ligne (t', f_m) dans la matrice $\mathcal{M}_{d,m}$ est combinaison linéaire des précédentes.*

L'algorithme F5-matriciel est décrit figure 1.3. Il est incrémental en d et m.

1.5.2 Algorithme F5-matriciel dans le cas $\mathbb{F}_2$ homogène

Supposons que l'on ajoute à l'idéal I les équations x_i^2. Alors de nouvelles réductions à zéro apparaissent, provenant des relations $f_i^2 = 0$. Le critère suivant supprime toutes les réduction à zéro provenant des relations $f_i^2 = 0$ (le même critère reste valable si l'on ajoute les équations de corps homogénéisées $x_i^2 = x_i h$, il supprime les réductions à zéro provenant des relations $f_i^2 = f_i h^{\deg(f_i)}$) :

Proposition 1.5.2 (Critère de Frobenius) *si une ligne d'étiquette (t, f_m) dans la matrice $\tilde{\mathcal{M}}_{d-d_m,m}$ a pour monôme de tête t' alors la ligne (t', f_m) dans la matrice $\mathcal{M}_{d,m}$ est combinaison linéaire des précédentes.*

Input : $\begin{cases} f_1, \ldots, f_m \text{ polynômes homogènes de degrés } d_1 \leq d_2 \leq \ldots \leq d_m, \\ d_{\max} \text{ un entier.} \end{cases}$

Output : $\{$Une base de Gröbner de $f_1, \ldots, f_m$ jusqu'au degré $d_{\max}$

POUR d de d_1 à $d_{\max}$ **FAIRE**
  $\mathcal{M}_{d,0}$:= matrice avec 0 lignes
  POUR i de 1 à m **FAIRE**
    construire $\mathcal{M}_{d,i}$ en ajoutant à $\mathcal{M}_{d,i-1}$ les lignes :
    (1) si $d_i = d$ ajouter les lignes f_i d'étiquette $(1, f_i)$
    (2) sinon, pour toute ligne f de $\tilde{\mathcal{M}}_{d-1,i}$ d'étiquette (e, f_i) telle que la plus grande variable de e soit x_λ, ajouter les $n-\lambda+1$ lignes $x_\lambda f, x_{\lambda+1} f, \ldots, x_n f$ (dans cet ordre) avec les étiquettes $(x_\lambda e, f_i), \ldots, (x_n e, f_i)$, sauf celles qui vérifient que $x_{\lambda+k} e$ est un terme de tête dans $\tilde{\mathcal{M}}_{d-d_i,i-1}$,
  calculer $\tilde{\mathcal{M}}_{d,m}$ la forme de Gauss de $\mathcal{M}_{d,m}$ (sans pivots)
RETOURNER tous les polynômes correspondant à une ligne dont le terme de tête n'est pas le même dans $\mathcal{M}_{d,m}$ et $\tilde{\mathcal{M}}_{d,m}$, pour tout $d \leq d_{\max}$.

FIG. 1.3 – Algorithme F5-matriciel

L'algorithme F5-matriciel se comporte comme précédemment, il suffit de modifier l'étape (2) comme suit :

(2) sinon, pour toute ligne f dans $\tilde{\mathcal{M}}_{d-1,i}$ d'étiquette (e, f_i) telle que la plus grande variable apparaissant dans e soit x_λ, ajouter les $n - \lambda$ lignes $x_{\lambda+1} f, x_{\lambda+2} f, \ldots, x_n f$ (dans cet ordre) d'étiquette $(x_{\lambda+1} e, f_i), \ldots, (x_n e, f_i)$, sauf celles vérifiant que $x_{\lambda+k} e$ est un terme de tête dans $\tilde{\mathcal{M}}_{d-d_i,i}$ (et non $\tilde{\mathcal{M}}_{d-d_i,i-1}$ comme précédemment).

Nous appelerons F5-matriciel/2 l'algorithme F5-matriciel dédié au calcul de bases de Gröbner pour les systèmes à coefficients et solutions dans $\mathbb{F}_2$, modifié comme ci-dessus.

1.6 Fonction et série de Hilbert

1.6.1 Définition

Un outil essentiel dans l'étude des idéaux de polynômes est la fonction de Hilbert d'un idéal. Cette fonction est une donnée intrinsèque de l'idéal, et ne dépend pas en particulier du système de générateurs choisi. Nous rappelons brièvement la définition et les propriétés de la fonction de Hilbert, le lecteur peut se reporter à [CLO97] pour plus de détails.

Rappels. Pour $s \in \mathbb{N}$ l'ensemble $(S_n)_s = \{f \in S_n : \deg(f) = s\}$ est un espace vectoriel sur $\mathbb{K}$, de dimension $\binom{n+s-1}{s}$. Si I est un idéal, alors $I_s = I \cap (S_n)_s$ est aussi un espace vectoriel sur $\mathbb{K}$.

Définition 1.6.1 *La fonction de Hilbert d'un idéal homogène $I = \langle f_1, \ldots, f_m \rangle$ en degré s est définie par*

$$HF_{s,m,\underline{d}_m}(n) = \dim\left(S_n/I\right)_s = \dim\left((S_n)_s/I_s\right) = \dim(S_n)_s - \dim(I_s)$$

où $\underline{d}_m = (d_1, \ldots, d_m)$ et d_i est le degré du polynôme homogène f_i.

À partir d'un certain degré, appelé la régularité de Hilbert, ou indice de régularité, cette fonction de s est égale à un polynôme en s, appelé polynôme de Hilbert. Le degré de ce polynôme est exactement la dimension de l'idéal. L'indice de régularité de l'idéal I est noté $H(I)$.

La série de Hilbert est définie par $HS_{n,m,\underline{d}_m}(z) = \sum_{s\geq 0} HF_{s,m,\underline{d}_m}(n) z^s$. Cette série est une fraction rationnelle, qui peut s'écrire $\frac{P(z)}{(1-z)^d}$, avec $P(1) \neq 0$. Alors d est la dimension de I, et $P(1)$ est le degré de la variété définie par I.

1.6.2 Fonction de Hilbert "générique"

Nous redonnons ici les résultats présentés dans [MS91], et provenant essentiellement de [Frö85, Ani86, FH94], pour le calcul de fonctions et séries de Hilbert d'idéaux génériques.

La définition d'idéal générique est la même que celle utilisée par Macaulay (Définition 1.4.1 page 17) : un idéal J de S_n est dit (homogène) générique s'il s'écrit $J = \langle g_1, \ldots, g_m \rangle$ avec pour $j \in [1; m]$, $g_j = \sum_{i_1+\ldots+i_n=d_j} U_{j,i_1,\ldots,i_n} x_1^{i_1} \cdots x_n^{i_n}$ est un polynôme à coefficient dans $\mathbb{K}(\{U_{j,i_1,\ldots,i_n}\}_{i_1+\ldots+i_n=d_j,\, j\in[1;m]})$. La série de Hilbert générique est la série de Hilbert associée à un idéal générique J, notée $G_{n,m,\underline{d}_m}(z)$.

On définit l'ordre $\leq$ sur les séries par $\sum a_i z^i \leq \sum b_i z^i$ si $a_i \leq b_i\ \forall i$, et on note $\left[\sum_{i\geq 0} a_i z^i\right] = \sum_{i\geq 0} b_i z^i$ où $b_i = a_i$ si $a_j > 0\ \forall 0 \leq j \leq i$ et $b_i = 0$ sinon.

Lemme 1.6.2 *Pour tout idéal I on a $HS_{n,m,\underline{d}_m}(z) \geq G_{n,m,\underline{d}_m}(z)$.*

Un idéal est dit H-générique si $HS_{n,m,\underline{d}_m}(z) = G_{n,m,\underline{d}_m}(z)$. On conjecture une valeur explicite pour cette borne inférieure : notons

$$F_{n,m,\underline{d}_m}(z) = \left[\prod_{i=1}^{m}(1-z^{d_i})\Big/(1-z)^n\right] \tag{1.2}$$

Conjecture 1.6.3 $G_{n,m,\underline{d}_m}(z) = F_{n,m,\underline{d}_m}(z)$

Théorème 1.6.4 *La conjecture 1.6.3 est prouvée dans les cas suivants :*

1. *$m \leq n$ (suite régulière),*
2. *$n = 2$,*
3. *$n = 3$ et le corps $\mathbb{K}$ est infini (ou "suffisamment gros"),*

4. $m = n + 1$ *en caractéristique 0,*
5. $d_i = 2\ \forall i \in [1; m]$ *et* $n \leq 11$ *;* $d_i = 3\ \forall i \in [1; m]$ *et* $n \leq 8$ *;*

Il est relativement facile de montrer que l'ensemble des idéaux H-génériques forme un ouvert de Zariski dans l'ensemble des idéaux, le problème étant de montrer que cet ouvert est non vide. Dans tous les cas du théorème précédent, la preuve consiste à exhiber un exemple d'idéal H-générique ayant $F_{n,m,\underline{d}_m}(z)$ comme série de Hilbert.

Dans le cas $m \leq n$, on utilise l'idéal monomial $I = \langle x_1^{d_1}, \ldots, x_n^{d_n} \rangle$. En fait, nous allons voir dans la section suivante que, plus généralement, toute suite régulière a pour série de Hilbert $F_{n,m,\underline{d}_m}(z)$. Mais les suites régulière n'existent que pour $m \leq n$. Dans le chapitre 3, nous définissons les suites semi-régulières, qui étendent la notion de suites régulière aux suites surdéterminées, et qui sont exactement les suites ayant pour série de Hilbert $F_{n,m,\underline{d}_m}(z)$. Malheureusement nous n'avons pas pu prouver d'autres cas généraux de la conjecture 1.6.3, le problème difficile étant ici de trouver un exemple explicite (ce qui est paradoxal, puisqu'il suffit de trouver *un* exemple explicite pour prouver que *presque tous* les systèmes marchent !).

1.7 Suites régulières

1.7.1 Définition des suites régulières

Les suites régulières ont été introduites par Macaulay [Mac02, Mac16]. Ce sont des suites ayant un comportement très prédictible, qui est également le "comportement moyen" des suites de polynômes.

Définition 1.7.1 (Suites régulières) *Une suite $f_1, \ldots, f_m \in S_n$ de m polynômes homogènes est régulière si les conditions suivantes sont vérifiées :*

- $\langle f_1, \ldots, f_m \rangle \neq S_n$
- *Pour tout $i \in [1; m]$, si $g_i f_i = 0$ dans $S_n/\langle f_1, \ldots, f_{i-1} \rangle$ alors $g_i = 0$ dans $S_n/\langle f_1, \ldots, f_{i-1} \rangle$*

Pour des suites homogènes, l'ordre n'a pas d'importance : toute permutation des polynômes donne encore une suite régulière [Eis95], ce qui est faux pour des suites affines. Les suites régulières affines sont généralement définies exactement de la même façon que pour les suites homogènes, et par exemple la suite $x, (x-1)z, (x-1)y$ est une suite régulière, alors que $(x-1)z, (x-1)y, x$ ne l'est pas (dans le premier cas, $(x-1)z = -z$ dans $\mathbb{K}[x, y, z]/(x)$ est non diviseur de zéro, et $(x-1)y = -y$ est non diviseur de zéro dans $\mathbb{K}[x, y, z]/(x, (x-1)z) = \mathbb{K}[x, y, z]/(x, z)$. Dans l'autre cas, $(x-1)y$ est diviseur de zéro dans $\mathbb{K}[x, y, z]/((x-1)z)$). De nombreuses autres propriétés des suites régulières homogènes ne sont plus valables pour les suites régulières affines définies ainsi, comme le calcul de la série de Hilbert ou la complexité du calcul de la base de Gröbner.

Nous considérons dans cette thèse une autre définition de suite régulière affine, plus restrictive :

Définition 1.7.2 *Une suite affine $f_1, \ldots, f_m$ est régulière si la suite homogène $f_1^h, \ldots, f_m^h$ l'est, où f_i^h est la partie homogène de f_i de plus grand degré.*

Cette définition nous permet de conserver toutes les propriétés des suites régulières homogènes (en particulier les calculs de complexité) pour les suites régulières affines. Dans toute la suite, une suite régulière affine s'entendra au sens de la définition 1.7.2.

Nous allons maintenant voir, de manière théorique, que partant d'une suite quelconque, on peut se ramener à une suite régulière. Cette proposition ne nous servira pas dans la suite de cette thèse, mais elle montre la portée des suites régulières.

Proposition 1.7.3 *Soit $I = \langle f_1, \ldots, f_m \rangle$ un idéal de codimension h. Alors il existe des $\lambda_{i,j}$ tels que la suite $(g_1, \ldots, g_m) = (f_1, f_2 + \lambda_{2,3} f_3 + \ldots + \lambda_{2,m} f_m, f_3 + \lambda_{3,4} f_4 + \ldots + \lambda_{3,m} f_m, \ldots, f_h + \lambda_{h,h+1} f_{h+1} + \ldots + \lambda_{h,m} f_m, f_{h+1}, \ldots, f_m)$ vérifie :*

- *$\{g_1, \ldots, g_m\}$ engendre I,*
- *$(g_1, \ldots, g_h)$ est une suite régulière*

et cette proposition est vraie pour tous $\lambda_{i,j} \in \mathbb{K}$ sauf un nombre fini.

Démonstration [Mat89] La preuve utilise le lemme suivant : si a, b sont deux polynômes de l'anneau de polynômes $\mathbb{K}[x_1, \ldots, x_n]$, et $\mathcal{P}$ un idéal premier, $b \notin \mathcal{P}$, alors

$$\#\{\lambda \in \mathbb{K} : a + \lambda b \in \mathcal{P}\} \leq 1$$

En effet, soit $\mathcal{M}$ un idéal maximal contenant $\mathcal{P}$ et ne contenant pas b (a vérifier : un idéal premier est l'intersection des idéaux maximaux qui le contiennent). Alors dans le corps $\mathbb{K}[x_1, \ldots, x_n]/\mathcal{M}$, on a $a + \lambda b = 0$ et $b \neq 0$, ce qui donne $\lambda = -a/b \in \mathbb{K}$ et il y a au plus une solution (et aucune si le polynôme b ne divise pas le polynôme a).

On a $g_1 = f_1$. Supposons que l'on ait construit la suite $g_1, \ldots, g_{i-1}$, régulière, construisons g_i non diviseur de zéro dans $\mathbb{K}[x_1, \ldots, x_n]/\langle g_1, \ldots, g_{i-1} \rangle$. On considère $\min(I_{i-1})$ l'ensemble des idéaux premiers minimaux (pour l'inclusion) associés[4] à $I_{i-1} = \langle g_1, \ldots, g_{i-1} \rangle$. Il suffit de trouver g_i qui n'appartienne à aucun de ces idéaux premiers, alors g_i sera non diviseur de zéro modulo les premiers polynômes. Si f_i n'appartient à aucun de ces premiers associés, alors $g_i = f_i$ convient. Supposons que $f_i \in \mathcal{P}_j$, alors il suffit par le lemme précédent de trouver un $f_{i+i_j} \notin \mathcal{P}_j$ pour que pour presque tout λ, $g_i = f_i + \lambda_j f_{i+i_j} \notin \mathcal{P}_j$. Or, si pour tout $l \geq 1$ on a $f_{i+l} \in \mathcal{P}_j$, alors $\mathcal{P}_j$ est un idéal premier associé à I, or une suite régulière est équidimensionnelle, donc $\dim(I) < \dim(I_{i-1}) = \dim(\mathcal{P}_j)$ donc $\mathcal{P}_j$ ne peut pas être un idéal associé à I. Si f_i est dans plusieurs idéaux associés à I_{i-1}, alors $g_i = f_i + \sum_{f_i \in \mathcal{P}_j} \lambda_j f_{i+i_j}$ n'est dans aucun $\mathcal{P}$. □

[4] Tout idéal I peut s'écrire $I = Q_1 \cap \cdots \cap Q_s$ avec $\sqrt{Q_i} = P_i$ un idéal premier. Les Q_i sont appelés idéaux primaires associés à I, et les P_i sont les idéaux premiers associés à I.

1.7.2 Caractérisations des suites régulières

Proposition 1.7.4 *Soit $f_1, \ldots, f_m$ une suite régulière homogène, et d_i le degré de f_i. Alors les propriétés suivantes sont vérifiées :*

1. *l'idéal $\langle f_1, \ldots, f_m \rangle$ est de dimension $n - m$,*
2. *la série de Hilbert de $f_1, \ldots, f_m$ est*

$$\sum_{d \geq 0} HF_{d,m,\underline{d}_m}(n) z^d = \prod_{i=1}^{m} (1 - z^{d_i}) \Big/ (1 - z)^n \tag{1.3}$$

   et réciproquement, toute suite $g_1, \ldots, g_m$ de degrés $d_1, \ldots, d_m$ ayant pour série de Hilbert (1.3) est une suite régulière,
3. *toute permutation $f_{\sigma(1)}, \ldots, f_{\sigma(m)}$ est une suite régulière,*
4. *l'indice de régularité est la borne de Macaulay $\sum_{i=1}^{m} (d_i - 1) + 1$,*
5. *presque toute suite est une suite régulière : l'ensemble des suites de n variables et degrés $d_1, \ldots, d_m$ qui sont régulières est un ouvert non vide de Zariski,*

Démonstration La propriété 1 se trouve dans [CLO97]. Il n'est pas évident de trouver une référence précise pour la propriété 2, elle est par exemple citée en exercice dans [Frö97, p. 137] ou peut être obtenue comme un corollaire de [Lan02, Théorème 6.6 p. 436]. Nous en donnons ici une preuve courte due à D. Lazard[5] : considérons la suite exacte $0 \to S_n/\langle f_1, \ldots, f_{i-1} \rangle \xrightarrow{f_i} S_n/\langle f_1, \ldots, f_{i-1} \rangle \to S_n/\langle f_1, \ldots, f_i \rangle \to 0$, alors les séries de Hilbert vérifient la relation $z^{d_i}\, HF(f_1, \ldots, f_{i-1}) - HF(f_1, \ldots, f_{i-1}) + HF(f_1, \ldots, f_i) = 0$, et $HF() = 1/(1-z)^n$ ce qui donne

$$HF(f_1, \ldots, f_m) = \prod_{i=1}^{m} (1 - z^{d_i}) \Big/ (1 - z)^n.$$

Pour la réciproque, considérons la suite exacte $0 \to K \to S_n/\langle f_1, \ldots, f_{i-1} \rangle \xrightarrow{f_i} S_n/\langle f_1, \ldots, f_{i-1} \rangle \to S_n/\langle f_1, \ldots, f_i \rangle \to 0$ où K est le noyau de la multiplication par f_i, alors la série de Hilbert de K est nécessairement nulle, donc $K = \{0\}$ et la suite est exacte. La propriété 3 est prouvée dans [Eis95], la propriété 4 dans [Laz83]. Quant à la dernière, il est facile de voir que l'ensemble des suites régulières est un ouvert de Zariski, qui est non vide puisqu'il contient la suite $x_1^{d_1}, \ldots, x_m^{d_m}$. □

1.8 Études de complexité existantes

Complexité théorique Le problème Ideal Membership (appartenance à un idéal) est un problème EXPSPACE-complet, qui peut être résolu en calculant une base

[5] La même preuve se trouve dans [CLO97] sans utiliser la notion de suite exacte, les propriétés des suites exactes utilisées ici sont redémontrées.

de Gröbner. Dans le pire cas, le calcul d'une base de Gröbner est doublement exponentiel en le nombre de variables, mais le comportement générique est bien plus efficace. Pour les systèmes de dimension zéro, le calcul de la base de Gröbner est simplement exponentiel en le nombre de variables.

Degré maximal apparaissant dans un calcul de base de Gröbner Une bonne mesure de complexité est le degré maximal d'un polynôme apparaissant au cours d'un calcul de base de Gröbner. Nous donnons ici quelques résultats connus, cette liste étant loin de l'exhaustivité. Un premier résultat concerne le cas le pire :

Proposition 1.8.1 ([MM82, BS88]) *Pour tout corps $\mathbb{K}$, il existe un système de n polynômes en n variables de degrés D tel que pour tout ordre admissible, la base de Gröbner de ce système contienne des polynômes de degré D^{2^n}, et le calcul de cette base de Gröbner est polynômial en D^{2^n} (les polynômes intervenant au cours du calcul restent creux).*

Ce cas est le pire qui puisse apparaître, et ne se rencontre pas dans le cas général. La proposition suivante concerne des systèmes plus "généraux", c'est un cas particulier du résultat de Lazard [Laz83], redémontré dans [Giu84] :

Théorème 1.8.2 ([Laz83, Giu84]) *Soit $f_1, \ldots, f_m$ un système de $m \leq n$ polynômes en n variables à coefficients dans un corps $\mathbb{K}$ quelconque. Si le système homogénéisé admet un nombre fini de solutions, alors pour l'ordre grevlex, et pour presque tout changement linéaire de variable, le degré maximal de n'importe quel polynôme intervenant au cours du calcul de la base de Gröbner est la borne de Macaulay $\sum_{i=1}^{n}(d_i - 1) + 1$, où d_i est le degré de f_i avec $d_1 \geq d_2 \geq \ldots d_m$, et le coût du calcul de la base de Gröbner est polynômial en D^n (avec $D = \max\{d_i\}$).*

("presque tout changement de variable" s'applique sur les variables $x_0, \ldots, x_n$, et implique en particulier que l'hyperplan à l'infini est en position générique).

Cas des suites régulières Nous avons vu que pour les suites régulières, l'indice de régularité est la borne de Macaulay. Cependant, les exemples suivants montrent que le degré maximal des éléments d'une base de Gröbner réduite varie considérablement selon l'ordre monomial utilisé.

Considérons d'abord la suite de $\mathbb{K}[x_0, x_1, \ldots, x_n]$ suivante :

$$\{x_1^d, x_0^{d-1}x_1 - x_2^d, \ldots, x_0^{d-1}x_{n-1} - x_n^d\} \tag{1.4}$$

C'est une suite de n polynômes en $n+1$ variables qui est régulière. La variété associée est donc de dimension 1.

Pour l'ordre grevlex$(x_1, \ldots, x_n, x_0)$, la suite est déjà une base de Gröbner, et le degré maximal est d. Considérons maintenant l'ordre grevlex$(x_0, x_1, \ldots, x_n)$. Dans ce cas, l'idéal contient les polynômes x_1^d, $x_2^{d^2}$, $\ldots$, $x_n^{d^n}$. La base de Gröbner contient $x_n^{d^n}$, et le degré maximal atteint lors du calcul est d^n.

La définition classique d'une suite en position de Noether est donnée par exemple dans [Eis95], nous en donnons une autre équivalente (d'après [BG01, lemme 4.1]), mais plus algorithmique :

Définition 1.8.3 *On dit que les variables $x_1, x_2, \ldots, x_n$ mettent la suite $f_1, \ldots, f_m$ en position de Noether si, pour l'ordre grevlex $x_1 > x_2 > \ldots > x_n$, pour tout $1 \leq j \leq m$ il existe n_j tel que $x_j^{n_j} \in LT(\langle f_1, \ldots, f_j \rangle)$.*

Ainsi, les variables $x_1, \ldots, x_n, x_0$ mettent la suite (1.4) est en position de Noether, alors que ce n'est pas le cas pour les variables $x_0, x_1, \ldots, x_n$.

En utilisant l'ordre grevlex, et à un changement linéaire près de coordonnées, le degré maximal d'un polynôme intervenant dans le calcul de la base de Gröbner de I est inférieur ou égal à la borne de Macaulay [Laz83]. Nous remontrerons ce résultat dans le chapitre 3 dans le cas où toute sous-suite $f_1, \ldots, f_i$ est en position de Noether, toujours pour l'ordre grevlex (mais sans changement de coordonnées).

Cas des systèmes surdéterminés On peut déduire de la proposition 1.8.2 une borne pour les systèmes surdéterminés, mais en pratique la complexité pour ces systèmes est très en dessous de cette borne. Ainsi, si l'on ajoute simplement un polynôme (i.e. on considère $n+1$ polynômes en n variables), la borne déduite de la proposition précédente est $\sum_{i=1}^{n}(d_i - 1) + 1$, qui est a peu près le double de la borne minimale estimée à partir des méthodes de résultants :

Proposition 1.8.4 ([Sza01]) *Soit $f_1, \ldots, f_{n+1}$ un système générique (i.e. les coefficients des f_i sont des paramètres, et f_i est vu comme un polynôme à coefficient dans l'anneau de polynôme de ses coefficients) de degrés $d_1, \ldots, d_{n+1}$ en n variables. Alors le résultant projectif du système homogénéisé peut être calculé à partir d'une matrice dont les lignes correspondent à des polynômes de l'idéal de degré au plus $(\sum_{i=1}^{n+1}(d_i - 1) + 1)/2$.*

Cela signifie que, dans le cas d'un système surdéterminé de $n+1$ polynômes *génériques* en n variables, de degrés $d_1, \ldots, d_{n+1}$, on peut calculer les solutions du système en considérant des polynômes de l'idéal de degré inférieur ou égal à la moitié de la borne de Macaulay, soit $(\sum_{i=1}^{n+1}(d_i - 1) + 1)/2$.

Le chapitre 3 est consacré à la définition et à l'analyse du comportement des suites semi-régulières, qui généralisent les suites régulières aux systèmes surdéterminés. Nous montrerons que l'on peut calculer le degré maximal d'un élément d'une base de Gröbner, et que par exemple pour tout système de $n+k$ équations en n variables, où $k > 0$ fixé, ce degré maximal est asymptotiquement équivalent à $(\sum_{i=1}^{n+1}(d_i - 1) + 1)/2$ lorsque $n \to \infty$, ce qui généralise le résultat de la proposition 1.8.4.

Chapitre 2

Décodage algébrique de codes correcteurs d'erreurs

Dans ce chapitre nous présentons rapidement la théorie des codes correcteurs d'erreurs, et nous explicitons le décodage algébrique des codes cycliques et le lien avec les calculs de bases de Gröbner. Nous rappelons les différents systèmes algébriques utilisés, et les algorithmes de décodage qui en découlent. Nous montrons les limites de ces systèmes pour une utilisation effective.

2.1 Introduction

La théorie des codes correcteurs a été introduite dans les années 40 par les travaux de Golay, Hamming et Shannon. Cette théorie a été développée pour répondre à un besoin (transmission de données sur un canal de communication bruité, compression de données, ...), et trouve de nos jours de nombreuses applications à d'autres domaines (par exemple en cryptographie, avec le système de chiffrement de McEliece [McE78]). Elle est fondée sur le théorème de Shannon de 1948 (dont on peut par exemple trouver une description dans [vL99]) qui assure l'existence de "bons" codes correcteurs d'erreurs. Un problème important de la théorie des codes reste la construction et l'analyse de "bons" codes.

Dans cette thèse, nous ne nous intéressons qu'à des codes en blocs (par opposition aux codes convolutifs, qui traitent les symboles les uns après les autres), dont le principe est décrit figure 2.1 : on veut transmettre un message par blocs $u = u_1 u_2 \dots u_k$ sur un canal bruité (le modèle utilisé est celui du canal binaire symétrique). Pour cela on encode ce message $u_1 u_2 \dots u_k$ en un mot du code $c = c_0 c_1 \dots c_{n-1}$ qui comporte des bits de redondance par rapport au message d'origine ($n > k$), ce qui va permettre de corriger les éventuelles erreurs de transmission.

Pouvoir *détecter* une erreur, c'est être capable de répondre à la question : le vecteur reçu $\tilde{c} = \tilde{c}_0 \tilde{c}_1 \dots \tilde{c}_{n-1}$ est-il égal à c ? Pouvoir *corriger* cette erreur, c'est être capable, après détection, d'obtenir par correction le vecteur $c' = c$ et donc après

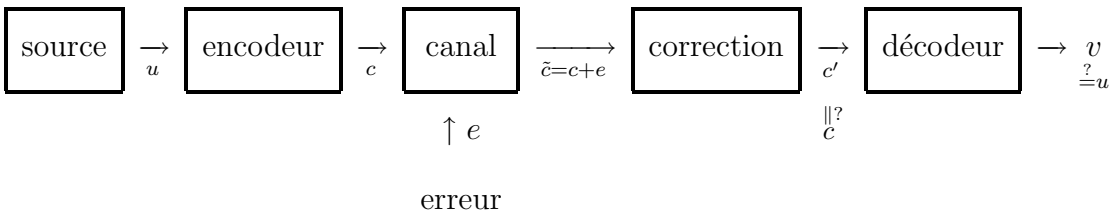


FIG. 2.1 – Schéma de transmission d'un message sur un canal bruité

décodage $v = v_1 v_2 \dots v_k = u = u_1 u_2 \dots u_k$.

Le taux d'efficacité du code (ou rendement) est $R = \frac{k}{n} = \frac{\text{nombre de bits d'information}}{\text{nombre de bits transmis}}$. Les autres paramètres importants d'un code correcteur sont sa capacité de correction, qui est le nombre d'erreurs que le code peut corriger, et sa capacité de détection. La *distance de Hamming* entre deux mots de code est le nombre de bits qui diffèrent, $d_H(x,y) = \{i \in [1;n] \mid x_i \neq y_i\}$, et la distance minimale d'un code correcteur est la distance minimale entre deux mots du code : $d = \min\{d_H(x,y) : x \neq y, (x,y) \in \mathcal{C}\}$. Le poids d'un mot est $w_H(c) = d_H(c,0)$, c'est le nombre de bits non nuls de ce mot. Le décodage au *maximum de vraisemblance* consiste à trouver le mot du code le plus proche d'un mot donné au sens de la distance de Hamming. Dans ce cas, la capacité de correction du code est la moitié de sa distance minimale. Un code de longueur n, de dimension k et de distance minimale d est dit de type $[n,k,d]$, et on note $t = \lfloor \frac{d-1}{2} \rfloor$ sa capacité de correction.

Les codes les plus simples sont les codes à répétitions :

Exemple 2.1.1 *Le code à trois répétitions :* 0 *s'encode en* 000 *et* 1 *s'encode en* 111*. Par exemple* 01011 *s'encode en* 000111000111111*. On décode par décision majoritaire (maximum de vraisemblance) :*

$$\underbrace{010}_{0}\,\underbrace{011}_{1}\,\underbrace{001}_{0}\,\underbrace{111}_{1}\,\underbrace{110}_{1} \longrightarrow 01011$$

Le taux d'efficacité du code est de $\frac{1}{3}$*, sa distance minimale est 3 et il corrige 1 erreur.*

Plus généralement, pour un code à n répétitions, le taux d'efficacité du code est $\frac{1}{n}$, sa capacité de détection est $n-1$, sa distance minimale est n et il corrige donc $\lfloor \frac{n}{2} \rfloor$ erreurs.

Les paramètres associés à un code permettent de définir plus précisément la notion de "bon" code correcteur d'erreur. On attend d'un bon code qu'il ait un bon rendement k/n et une grande capacité de correction t, ces deux critères étant, comme on peut s'y attendre, contradictoires. Dans le cas des codes linéaires (un code linéaire de type $[n,k,d]$ sur $\mathbb{F}_q$ est un sous espace vectoriel de $\mathbb{F}_q^n$), l'inégalité des empilements de sphères donne le nombre maximal de mots décodables à n et k fixés. Un code pour lequel cette borne est atteinte est appelé code *parfait*. Les

codes à répétitions de longueur impaire sur $\mathbb{F}_2$ sont des codes parfaits. Tous les codes parfaits existants sont répertoriés : les seuls vérifiant $t > 1$ et $n > 2$ sont, outre les codes à répétition de longueur impaire sur $\mathbb{F}_2$, de type $[n \text{ impair}, 1, n]$, les deux codes de Golay, de type $[23, 12, 7]$ sur $\mathbb{F}_2$ et $[11, 6, 5]$ sur $\mathbb{F}_3$ [MS77, page 179].

L'existence de bons codes est assurée par la borne de Varshamov-Gilbert, qui est une borne inférieure sur t pour les meilleurs codes. La plupart des codes linéaires atteignent cette borne, en particulier les codes choisis au hasard. Cependant, un bon code pris au hasard a peu de chances de posséder un algorithme de décodage performant. Le problème général de décodage d'un code linéaire est un problème NP-complet [BMvT78].

Une classe importante de codes correcteurs est celle des codes cycliques, qui sont des cas particuliers de codes linéaires. Une sous-classe importante de codes cycliques à été découverte par R. C. Bose et D. K. Ray-Chaudhury en 1960 et indépendamment par A. Hocquenghem en 1959. Ces codes, connus sous le nom de codes BCH, sont très utilisés en pratique, notamment parce qu'il existe pour cette famille de codes cycliques un algorithme de décodage très efficace dû à Berlekamp-Massey [Ber68]. Par exemple, le C.C.S.D.S. (Consultative Committee for Space Data System[1]) recommande d'utiliser, dans les systèmes de télémesure par satellite, le code de Reed-Solomon (les codes de Reed-Solomon sont des codes BCH particuliers) de type $[255, 223, 33]$ sur $\mathbb{F}_{2^8}$ (il est de rendement $R = 0.87$, sa distance minimale est 33, donc il corrige 16 erreurs). Le code de Reed-Solomon $[31, 15, 17]$ sur $\mathbb{F}_{32}$ est utilisé dans les communications militaires.

Il existe de nombreuses autres familles de codes cycliques, comme par exemple les codes à résidus quadratiques (codes RQ). Ces codes RQ sont de bons codes - leur rendement est proche de $1/2$ et on connaît de bonnes bornes sur leur capacité de correction - mais il n'existe pas à l'heure actuelle d'algorithme de décodage générique et efficace des codes RQ. La famille des codes RQ contient en particulier les deux codes parfaits de Golay. Plus généralement, étant donnés un corps de base et une dimension n, on peut construire une large variété de codes cycliques, mais on ne sait ni décoder, ni déterminer la capacité de correction de la plupart d'entre eux.

Décodage algébrique. Les bases de Gröbner sont un outil très utilisé pour le décodage de codes correcteurs [MS03a, CM02, Aug96, BF01, RH99, LVY97, CRHT94b]... Nous présentons dans ce chapitre les méthodes existantes de décodage algébrique des codes cycliques à l'aide de calculs de base de Gröbner.

L'idée du décodage algébrique est de réécrire le problème du décodage en un système d'équations algébriques, ayant les caractéristiques suivantes :

1. *correction de la mise en équation* : le calcul des solutions du système permet de retrouver l'erreur qui s'est produite,

[1] http ://www.ccsds.org/

2. *effectivité du calcul des solutions* : le calcul des solutions de ce système peut être fait en un temps raisonnable (temps polynômial par exemple).

Un tel système est un système à paramètres : une partie des variables, les paramètres, correspondent à des quantités qui peuvent être calculées pour chaque erreur, et l'on s'intéresse aux solutions du système dans lequel les paramètres ont été spécialisés.

Un algorithme de décodage comporte deux phases : la première phase est la phase de *pré-calcul*, qui peut nécessiter une grande puissance de calcul sur un long terme. La seconde est la phase de *décodage*, où l'on reçoit un flot de mots codés à décoder. On calcule alors pour chaque mot la valeur des paramètres, et on obtient les solutions du système dans lequel on a remplacé les paramètres par leur valeur calculée pour ce mot précis. Cette seconde phase doit être la plus rapide possible, et peut comporter des limitations liées à la puissance de calcul autorisée ou à la quantité de mémoire disponible.

Le calcul de base de Gröbner peut s'utiliser dans chacune de ces deux phases. Si l'on considère les paramètres comme des variables, on peut calculer une base de Gröbner du système avec paramètres. Ce pré-calcul donne des formules, et la phase de décodage consiste simplement, pour chaque mot à décoder, à calculer les paramètres correspondants et à le substituer dans les formules. On parle dans ce cas de *décodage formel*. La seconde solution consiste, pour chaque mot à décoder, à calculer les paramètres correspondants, à les substituer dans le système algébrique et à calculer une base de Gröbner du système ainsi spécialisé. On parle alors de *décodage en ligne*. Pour le décodage formel, la base de Gröbner est calculée une fois pour toutes, mais le système contient beaucoup plus de variables que pour un décodage en ligne, et la base de Gröbner est souvent impossible à calculer. Dans le cas d'un décodage en ligne, il faut recalculer une base de Gröbner pour chaque mot, mais chaque base est beaucoup plus facile à calculer.

Syndrome, localisateurs et fonctions symétriques élémentaires. À partir d'un mot bruité reçu, il est possible de calculer le *syndrome*, qui est l'ensemble de paramètres associé à l'erreur transmise. Décoder une erreur, c'est pouvoir retrouver l'erreur à partir de son syndrome. Pour cela, on introduit un ensemble de variables, les *localisateurs* : si les positions non nulles de l'erreur $e = e_0 e_1 \dots e_{n-1}$ sont les $i_1, \dots, i_v$ (c'est-à-dire $e_{i_j} \neq 0$), les localisateurs sont les puissances $\alpha^{i_1}, \dots, \alpha^{i_v}$ où α est une racine primitive n-ème de l'unité. Avec ces notations, le syndrome de l'erreur correspond à un ensemble de fonctions puissances des localisateurs. Pour décoder une erreur, il suffit donc de retrouver les localisateurs à partir du syndrome, ou de manière équivalente de retrouver les fonctions symétriques élémentaires des localisateurs à partir du syndrome.

Les systèmes algébriques et leurs difficultés sur $\mathbb{F}_2$. De nombreuses relations algébriques existent entre les syndromes et les localisateurs, ainsi que les fonctions

puissances et les fonctions symétriques élémentaires des localisateurs, en particulier les relations de Newton. Si l'on se place dans le corps des rationnels, les relations de Newton permettent d'exprimer les fonctions symétriques élémentaires en fonction des fonctions puissances, mais modulo 2 cette propriété n'est plus vérifiée, et les équations de Newton sont plus difficiles à résoudre. Une autre difficulté du calcul modulo 2 est que les localisateurs sont des racines n-ème de l'unité, et donc les syndromes et les fonctions symétriques élémentaires des localisateurs vérifient des équations de corps (de la forme $x^{2^m} + x$, où m est l'ordre multiplicatif de 2 modulo n). Ces équations sont vite de très haut degré, et les prendre en compte constitue une vraie difficulté pour le calcul de la base de Gröbner. Inversement, les ignorer revient à chercher les solutions du système dans la clôture algébrique de $\mathbb{F}_2$, ce qui peut augmenter de manière drastique le nombre de solutions du système, et rendre le calcul de la base de Gröbner plus difficile.

Tous les systèmes étudiés jusqu'ici contiennent ces équations de corps. Les premiers systèmes qui ont été proposés [CRHT94a] utilisent l'expression des syndromes en fonction des localisateurs de l'erreur, et les résultats ont été démontrés par Loustaunau et Von York [LVY97], fournissant un algorithme de décodage formel basé sur un pré-calcul de base de Gröbner. On trouve également une étude de ces systèmes dans [CM02]. Il est cependant souvent impossible de calculer la base de Gröbner du système formel, même pour le code RQ de longueur 41.

En parallèle, de nombreux travaux ont été menés à partir des identités de Newton, en essayant de trouver des formules pour les fonctions symétriques élémentaires des localisateurs en fonction des syndromes. Ceci a été appliqué notamment aux codes à résidus quadratiques - par exemple, [RYT90, RTCY92, RRTC01, CRT94] pour des codes binaires, et [Hum92, HH93] pour des codes ternaires. Dans ces papiers, les auteurs construisent pour chaque code RQ particulier un algorithme de décodage spécifique à partir des identités de Newton, et ils ne donnent aucune méthode générale. Notons que, grâce aux équations de corps, tous les systèmes considérés sont de *dimension zéro* (ils possèdent un nombre fini de solutions), mais le degré des polynômes engendrés par les équations de corps rendent généralement le calcul de la base de Gröbner impossible.

Organisation du chapitre. Dans une première partie (section 2.2), nous définissons mathématiquement les codes cycliques, plus particulièrement les codes à résidus quadratiques et les codes BCH. Les sections 2.3 et 2.4 présentent les diverses approches du décodage algébrique par bases de Gröbner, pour des systèmes de dimension zéro.

2.2 Codes cycliques : définitions et propriétés

Dans cette section, p désigne un nombre premier, q une puissance de ce nombre premier et n un entier tel que $\text{pgcd}(n, q) = 1$ (cette hypothèse sera justifiée plus

loin). La construction et les performances d'un code cyclique sont fortement basées sur les propriétés des polynômes à coefficients dans un corps fini. Nous renvoyons le lecteur à l'annexe A.1 ou à l'ouvrage [LN97] pour des rappels sur les corps finis et polynômes à coefficients dans un corps fini, et aux ouvrages [LN97, MS77, vL99] pour une étude détaillée des codes cycliques.
Notation. Si r est un entier, nous notons $[1;r]$ l'intervalle des entiers de 1 à r.

2.2.1 Codes linéaires, définition des bons codes

Définition 2.2.1 *Un code linéaire $\mathcal{C}$ de longueur n et de dimension k sur F_q est un sous-espace vectoriel de $(\mathbb{F}_q)^n$ de dimension k.*

Soit $\mathcal{C}$ un code linéaire de type $[n,k,d]$ et $t = \lfloor \frac{d-1}{2} \rfloor$ sa capacité de correction. Notons $V_t^{(n)} = \sum_{i=0}^{t} \binom{n}{i}(q-1)^i$ le nombre de points de l'espace à distance au plus t d'un mot du code. La relation suivante, appelée borne des empilements de sphères, donne le nombre maximal de mots décodables en fonction de n et t : $V_t^{(n)} \leq q^{n(1-R)}$. Un code pour lequel cette inégalité est une égalité est appelé code parfait, c'est un code qui possède le maximum de mots décodables à n et k fixés.

Le théorème ci-dessous explicite une notion de "bon" code :

Théorème 2.2.2 *Si $0 \leq \delta \leq \frac{1}{2}$, alors il existe une famille infinie de codes linéaires de type $[n,k,d]$ avec $n \to \infty$, vérifiant[2] $\frac{d}{n} \geq \delta$ et $R = \frac{k}{n} \gtrsim 1 - H_2(\frac{d}{n})$ lorsque $n \to \infty$, avec $H_2(x) = -x\log_2(x) - (1-x)\log_2(1-x)$. Cette borne s'appelle la borne de Varshamov-Gilbert.*

Un code linéaire tiré au hasard (on tire au hasard les vecteurs engendrant l'espace vectoriel du code) est un bon code, au sens ou il atteindra cette borne [Bar97, Bar98].

2.2.2 Codes cycliques

Définition 2.2.3 (Code cyclique) *Un code cyclique $\mathcal{C}$ de longueur n sur $\mathbb{F}_q$ est un idéal de l'algèbre*

$$R_n = \mathbb{F}_q[x]/(x^n - 1) \simeq \mathbb{F}_q^n.$$

(On identifie $\mathbb{F}_q^n$ à R_n par $(c_1, \ldots, c_{n-1}, c_n) \to c_1 + c_2 x + \ldots + c_n x^{n-1}$). Si $q = 2$ on parle de code binaire.

Nous avons choisi $\text{pgcd}(n,q) = 1$, donc $x^n - 1$ n'a que des facteurs simples. Notons $\alpha \in \mathbb{F}_{q^m}$ une racine primitive $n^{\text{ème}}$ de l'unité, où m est l'ordre multiplicatif de q modulo n et $\mathbb{F}_{q^m}$ est le corps de décomposition de $x^n - 1$ (cf. annexe A.1).

L'anneau R_n est principal, donc un code cyclique $\mathcal{C}$ est défini par $g(x)$ son polynôme générateur unitaire de degré minimal. C'est un diviseur de $x^n - 1$, qui

[2] où l'on note $f(n) \gtrsim g(n)$ lorsque $n \to \infty$ si il existe une fonction $\epsilon(n)$ avec $|\epsilon(n)| \to 0$ quand $n \to \infty$ et $f(n) \geq g(n)(1+\epsilon(n))$.

s'écrit donc $g(x) = \prod_{i \in Q}(x - \alpha^i)$. La classe cyclotomique d'un entier i est $\mathcal{C}l_i = \{q^j i \mod n, j \in [1; m]\}$. L'ensemble Q est une réunion de classes cyclotomiques, et s'appelle l'ensemble des zéros ou *l'ensemble de définition* du code. Réciproquement, un code cyclique est entièrement défini par la donnée d'une réunion de classes cyclotomiques. La dimension du code est $k = n - \deg(g)$, on code un message $f(x) \in \mathbb{F}_q[x]$ de degré au plus $k-1$ par $f(x)g(x)$.

On dit que $\mathcal{C}$ est un code cyclique de type $[n, k, d]$ s'il est de longueur n, de dimension k et de distance minimale d.

Exemple 2.1.1 (suite) Le code à trois répétitions est un code cyclique binaire de générateur $g(x) = x^2 + x + 1$ et d'ensemble de définition $\{1, 2\}$. On code 0 en 0 et 1 en $g(x)$. C'est un code de type $[3, 1, 3]$, qui corrige donc une erreur et en détecte deux.

2.2.3 Codes à résidus quadratiques

Les codes à résidus quadratiques sont des codes possédant de très bons paramètres, étudiés pour la première fois par Assmus et Mattson [AM72] en 1972, et qui ont fait l'objet de nombreuses recherches théoriques.

Définition 2.2.4 *Soit n et q deux entiers premiers tels que q soit un carré modulo n. Le code à résidus quadratiques de longueur n sur $\mathbb{F}_q$ est le code cyclique dont l'ensemble de définition Q_n est l'ensemble des carrés modulo n. Son générateur est*

$$g(x) = \prod_{i \in Q_n}(x - \alpha^i) = \prod_{r^2 \in [1; n-1]}(x - \alpha^{r^2}).$$

Exemple 2.2.5 *Le code binaire à résidus quadratiques de longueur 31 est défini par $Q_{31} = \{1, 2, 4, 5, 7, 8, 9, 10, 14, 16, 18, 19, 20, 25, 28\}$. Il est de type $[31, 16, 7]$, de générateur $g(x) = x^{15} + x^{14} + x^{13} + x^9 + x^8 + x^3 + 1$ et $\alpha \in \mathbb{F}_{2^5}$ est une racine de $x^5 + x^2 + 1$. Les classes cyclotomiques de Q_{31} sont $\mathcal{C}l_1 = \{1, 2, 4, 8, 16\}$, $\mathcal{C}l_5 = \{5, 10, 20, 9, 18\}$ et $\mathcal{C}l_7 = \{7, 14, 28, 25, 19\}$.*

Exemple 2.2.6 *Les codes à résidus quadratiques sur $\mathbb{F}_2$ de longueur ≤ 50 ont pour ensembles de définition Q_7, Q_{17}, Q_{23}, Q_{31}, Q_{41} et Q_{47}.*

Proposition 2.2.7 ([MS77]) *Le code à résidus quadratiques Q_n de longueur n sur $\mathbb{F}_q$ est de dimension $k = \frac{n+1}{2}$ et a une distance minimale impaire.*

Proposition 2.2.8 ([vL99, p. 87]) *Dans le cas binaire ($q = 2$), nécessairement $n = \pm 1 \mod 8$, et la distance minimale d du code Q_n vérifie $d^2 \geq n$. Si de plus on a $n = -1 \mod 4$, alors $d^2 - d + 1 \leq n$ et $d = 3 \mod 4$.*

2.2.4 Codes BCH

Définition 2.2.9 *Soit q une puissance d'un nombre premier, n et $\delta \leq n$ deux entiers. Le code BCH au sens strict de longueur n et de distance construite δ sur $\mathbb{F}_q$ est le code cyclique de générateur $g(x) = \prod_{i \in Q}(x - \alpha^i)$ où $Q = Cl(1) \cup Cl(2) \cup \ldots \cup Cl(\delta - 1)$. On le note $BCH(n, \delta)$.*

Proposition 2.2.10 *La distance minimale d d'un code $BCH(n, \delta)$ vérifie $d \geq \delta$.*

Théorème 2.2.11 *Les codes BCH de longueur $n = 2^m - 1$ sur $\mathbb{F}_2$ ont une distance minimale impaire. On les appelle les codes BCH binaires primitifs.*

2.3 Aspects algébriques du décodage, relations de Newton

Dans tout ce qui suit, nous nous limiterons au cas de codes binaires ($q = 2$). Soit $\mathcal{C}$ un code cyclique binaire de type $[n, k, d]$, d'ensemble de définition Q. On note $t = \lfloor \frac{d-1}{2} \rfloor$ la capacité de correction du code. On se donne $\alpha \in \mathbb{F}_{2^m}$ une racine primitive $n^{\text{ème}}$ de l'unité, où $\mathbb{F}_{2^m}$ est le corps de décomposition de $x^n - 1$.

Considérons $c(x) \in \mathcal{C}$ un mot de code, et $\tilde{c} = c + e$ le mot bruité reçu après transmission, où $e(x) = \sum_{r=0}^{n-1} e_r x^r$ est l'erreur. Si $i_1, \ldots, i_v$ correspondent aux positions non nulles de cette erreur, on appelle $(Z_j^* = \alpha^{i_j})_{j \in [1;v]}$ les *localisateurs* de e. On dit que l'erreur e est de poids v. Comme un mot c est un mot du code si et seulement si $c(\alpha^i) = 0$ pour tout $i \in Q$, on peut calculer les paramètres $S_i^* = e(\alpha^i) = \tilde{c}(\alpha^i)$ pour $i \in Q$. Remarquons que $S_{2\,i}^* = (S_i^*)^2$.

Définition 2.3.1 (Syndrome) *L'ensemble $\underline{S}^* = \{S_i^*,\, i \in Q\} \subset \mathbb{F}_{2^m}$ s'appelle le syndrome de l'erreur e. Il est entièrement défini par $\underline{S}_{rep}^* = \{S_i^*,\, i \in Q_{rep}\}$ où Q_{rep} est un ensemble de représentants des classes cyclotomiques de Q. Il est calculable à partir du mot bruité reçu.*

Convention. Dans ce chapitre, nous distinguerons une valeur particulière d'un paramètre de la variable correspondante par l'ajout d'une étoile $*$. Ainsi, S_1^* est une valeur particulière de la variable S_1.

Un outil important dans l'étude des codes cycliques est la transformée de Fourier d'un mot. On préférera la définir pour des polynômes à coefficients dans $\overline{\mathbb{F}}_2$ la clôture algébrique de $\mathbb{F}_2$, en suivant [Aug93] :

Définition 2.3.2 *La transformée de Fourier de $a = \sum_{r=0}^{n-1} a_r x^r \in \overline{\mathbb{F}}_2[x]/(x^n - 1)$ est le polynôme $S(Z) = \sum_{i=1}^{n} S_i Z^{n-i} \in \overline{\mathbb{F}}_2[Z]$ de degré au plus $n - 1$, où $S_i = a(\alpha^i) = \sum_{r=0}^{n-1} a_r \alpha^{ir}$. On l'appelle aussi le polynôme de Mattson-Solomon de a.*

Proposition 2.3.3 *La transformée de Fourier est une application bijective de $\overline{\mathbb{F}}_2^n$ dans lui-même.*

Ainsi, connaître la transformée de Fourier d'une erreur revient à connaître l'erreur elle-même. Cependant, pour un mot bruité reçu $\tilde{c} = c + e$, on ne peut calculer qu'une partie de la transformée de Fourier de e, le syndrome. En effet, si $i \in Q$ alors $S_i^* = e(\alpha^i) = \sum_{j=1}^{v} (Z_j^*)^i$ est bien l'un des coefficients de la transformée de Fourier de e. Les autres coefficients sont les $S_i^* = \sum_{j=1}^{v} (Z_j^*)^i$ pour $i \notin Q$. La proposition suivante prouve que cette connaissance du seul syndrome est tout de même suffisante lorsque le poids de l'erreur est inférieur à la capacité de correction du code :

Proposition 2.3.4 *L'application suivante est injective :*

$$\left(\begin{array}{ccc} \{\textit{mots de poids} \leq t = \lfloor \frac{d-1}{2} \rfloor\} & \longrightarrow & \mathbb{F}_{2^m}^{\#Q} \\ e & \longmapsto & \underline{S}^* \end{array} \right)$$

Démonstration Soit e_1 et e_2 deux erreurs ayant même syndrome $\{e_1(\alpha^i),\, i \in Q\}$. Alors $e = e_1 - e_2$ vérifie $e(\alpha^i) = e_1(\alpha^i) - e_2(\alpha^i) = 0$ pour tout $i \in Q$ et donc $e \in \mathcal{C}$. Or, $w(e) \leq w(e_1) + w(e_2) \leq 2t \leq d-1$ et par définition de la distance minimale du code on a $e = 0$ et $e_1 = e_2$. □

On définit le *polynôme localisateur* de e par :

$$\begin{array}{rcl} L_v(Z) = & \prod_{j=1}^{v} (Z - \alpha^{i_j}) = \prod_{j=1}^{v} (Z - Z_j^*) & \\ = & Z^v - \sigma_1^* Z^{v-1} + \cdots + (-1)^{v-1} \sigma_{v-1}^* Z + (-1)^v \sigma_v^* & \end{array} \tag{2.1}$$

où les $\sigma_i^* = (-1)^i \sum_{j_1 < j_2 < \cdots < j_i} Z_{j_1}^* \cdots Z_{j_i}^*$ sont les fonctions symétriques élémentaires des Z_j^*.

Convention. Nous utiliserons les notations $\underline{Z}_v = (Z_j)_{j \in [1;v]}$ pour les variables et $\underline{Z}_v^* = (Z_j^*)_{j \in [1;v]}$ pour des valeurs particulières de ces variables. Nous notons de même $\underline{\sigma}_v = (\sigma_j)_{j \in [1;v]}$, $\underline{S} = (S_i)_{i \in Q}$ et $\overline{S} = (S_i)_{i \notin Q}$, ainsi que $\underline{\sigma}_v^*$, $\underline{S}^*$ et $\overline{S}^*$.

Décoder $\tilde{c}$, c'est pouvoir retrouver les valeurs des localisateurs $\underline{Z}_v^*$ à partir du syndrome de l'erreur $\underline{S}^*$. Or, le polynôme localisateur contient toute cette information, et la méthode de Chien ("Chien search", [MS77, p. 276]) permet de trouver les racines du localisateur en un temps négligeable par rapport au temps nécessaire au décodage. C'est une technique de recherche exhaustive (on essaie toutes les racines $n^{\text{èmes}}$ de l'unité) qui peut être implantée de manière très efficace. Pour connaître l'erreur, il est donc équivalent, sur le plan théorique comme sur le plan algorithmique, de retrouver soit ses localisateurs soit les fonctions symétriques élémentaires de ces derniers.

Les localisateurs, les syndromes et les fonctions symétriques élémentaires des localisateurs d'une erreur vérifient de nombreuses équations algébriques, résumées dans la proposition suivante :

Proposition 2.3.5 *Soit $\mathcal{C}$ un code cyclique d'ensemble de définition Q. Soit $e \in \mathbb{F}_2[x]/(x^n - 1)$ un mot de poids w avec $w \leq v \leq n$, et notons $\underline{Z}_w^*$, $\underline{\sigma}_w^*$, $\underline{S}^*$ et $\overline{S}^*$*

respectivement les localisateurs, les fonctions symétriques élémentaires des localisateurs, les syndromes et les fonctions puissances inconnues de e. Alors ces quantités satisfont les relations algébriques suivantes :
L'expression des syndromes comme fonctions puissances des localisateurs :

$$\textsc{Syndrom}_w = \left\{ S_i - \sum_{j=1}^{w} Z_j^i, \qquad \forall\, i \in Q \right\}. \tag{2.2}$$

ainsi que l'expression des fonctions symétriques élémentaires des localisateurs :

$$\textsc{Sym}_w = \left\{ \sigma_j - \sum_{1 \le l_1 < \cdots < l_j \le w} Z_{l_1} \cdots Z_{l_j}, \qquad \forall j \in [1; w] \right\}. \tag{2.3}$$

Les relations de Newton généralisées pour le poids v (avec $\sigma^*_{w+1} = 0 = \cdots = \sigma^*_v$) :

$$\textsc{NewGen}_v = \left\{ \; S_{(v+i \mod n)} + \sum_{j=1}^{v} \sigma_j S_{(v+i-j \mod n)} \quad \forall i \in [1; n] \; \right\}. \tag{2.4}$$

Comme le code considéré est un code binaire, les $\underline{S}^*$, $\overline{S}^*$ et $\underline{\sigma}^*_v$ vérifient de plus la partie triangulaire des relations de Newton :

$$\textsc{NewTri}_v = \left\{ S_i + \sum_{j=1}^{i-1} S_{i-j}\sigma_j + i\sigma_i \quad \forall i \in [1; v] \right\}. \tag{2.5}$$

Les équations de corps et de longueur :

$$\left\{ \begin{array}{ll} S_i^{2^m} + S_i & \forall\, i \in [1; n] \\ \sigma_i^{2^m} + \sigma_i & \forall\, i \in [1; w] \\ Z_j^{n+1} + Z_j & \forall\, j \in [1; w] \end{array} \right\}. \tag{2.6}$$

Nous présentons dans les deux sections suivantes différents systèmes construits à partir de ces équations qui permettent de retrouver l'erreur à partir de son syndrome. Nous verrons que ces systèmes ont des comportements très différents vis-à-vis des bases de Gröbner.

2.4 Décodage à partir d'idéaux de dimension zéro

Dans cette section, nous présentons les différents systèmes étudiés, ainsi qu'un nouveau système $\textsc{SynSym}^+_v$, obtenu en symétrisant le système $\textsc{Syndrom}^+_v$, et dont le comportement est similaire à celui des autres systèmes. Nous récapitulons les résultats existants, donnant des algorithmes de décodage algébrique des codes cycliques.

Tous les systèmes utilisés sont composés d'équations provenant de la proposition 2.3.5, et tous contiennent des équations de longueur ou de corps qui les rendent de dimension zéro.

Nous considérons les trois systèmes suivants :

$$\textsc{Syndrom}_v^+ = \left\{ \begin{array}{ll} S_i - \sum_{j=1}^{v} Z_j^i & \forall\ i \in Q \\ Z_j^{n+1} - Z_j & \forall\ j \in [1;v] \end{array} \right\} \subset \mathbb{F}_2[\underline{Z}_v, \underline{S}]$$

$$\textsc{Newton}_v^+ = \left\{ \begin{array}{ll} \sigma_j^{q^m} - \sigma_j, & j \in [1;v] \\ S_j^{q^m} - S_j, & j \in [1;n] \\ S_i + \sum_{j=1}^{i-1} \sigma_j S_{i-j} + i\sigma_i, & i \in [1;v] \\ S_{v+i} + \sum_{j=1}^{v} \sigma_j S_{v+i-j}, & i \in [1;n] \end{array} \right\} \subset \mathbb{F}_2[\underline{S}, \overline{S}, \underline{\sigma}_v]$$

$$\textsc{SynSym}_v^+ = \left\{ \begin{array}{ll} S_i - \sum_{j=1}^{v} Z_j^i & \forall i \in Q \\ \sigma_j - \sum_{l_1<\cdots<l_j} Z_{l_1} \cdots Z_{l_j} & \forall j \in [1;v] \\ Z_j^{n+1} - Z_j & \forall j \in [1;v] \end{array} \right\} \subset \mathbb{F}_2[\underline{Z}_v, \underline{S}, \underline{\sigma}_v]$$

Convention. Un système comportant les équations de longueur ou de corps pour ses variables est noté avec un $^+$ ajouté au nom du système sans ces équations. Ainsi, le système $\textsc{Syndrom}_v^+$ correspond au système $\textsc{Syndrom}_v$ auquel on a ajouté les équations de longueur[3] pour les variables Z_j.

Notons que $S_{n+i} = S_i$, i.e. S_i est en fait la variable $S_{i \bmod n}$. Les deux premiers systèmes sont étudiés par Chen et al. [CRHT94c, CRHT94b]. Les auteurs étudient le décodage en ligne. Le décodage formel est introduit par ces mêmes auteurs dans [CRHT94a] à partir du système $\textsc{Syndrom}_v^+$, et les preuves, basées sur une affirmation fausse[4], sont corrigées par Loustaunau et Von York dans [LVY97]. L'idéal $\textsc{SynSym}_v^+$ n'est pas considéré par ces auteurs, mais il est naturel d'introduire les fonctions symétriques élémentaires dans ces systèmes, et les résultats et preuves sont similaires à celles produites pour le système $\textsc{Syndrom}_v^+$.

Nous redonnons les propriétés de ces trois systèmes algébriques et expliquons en particulier comment le calcul d'une base de Gröbner fournit un algorithme de décodage. Des exemples détaillés sont donnés, qui illustrent le comportement de ces systèmes pour différents codes à résidus quadratiques (longueur 23, 31 et 41). Nous verrons au chapitre 6, section 6.2, une classification de tous ces systèmes.

2.4.1 Préliminaires

Un point commun à tous ces systèmes est qu'ils contiennent des équations de corps ou de longueur ($Z_j^{n+1} - Z_j$, $\sigma_j^{q^m} - \sigma_j$ ou $S_j^{q^m} - S_j$), qui n'ont que des solutions

[3] Il est suffisant d'ajouter les équations de longueur des variables Z_j pour que l'idéal contienne également les équations de corps pour les variables S_i

[4] Les auteurs utilisent la "propriété" qu'une base de Gröbner d'un idéal de dimension zéro pour un ordre lexicographique est triangulaire, c'est-à-dire comporte autant de polynômes que de variables. La figure 2.3 page 49 donne un exemple d'une telle base, qui est loin d'être triangulaire.

distinctes. D'après le lemme de Seidenberg A.2.2 page 144, cela implique que les idéaux étudiés sont tous radicaux[5] :

Lemme 2.4.1 *Les idéaux* $\langle \text{NEWTON}_v^+ \rangle$, $\langle \text{SYNDROM}_v^+ \rangle$ *et* $\langle \text{SYNSYM}_v^+ \rangle$ *sont radicaux et possèdent un nombre fini de solutions.*

Ces deux propriétés restent vraies pour toute spécialisation, i.e. pour tout $\underline{S}^* \subset \overline{\mathbb{F}}_2$ *les idéaux* $\langle \text{NEWTON}_v^+(\underline{S}^*) \rangle$, $\langle \text{SYNDROM}_v^+(\underline{S}^*) \rangle$ *et* $\langle \text{SYNSYM}_v^+(\underline{S}^*) \rangle$ *sont radicaux et de dimension zéro.*

Démonstration Chaque idéal contient une équation en chaque variable, donc l'idéal est bien de dimension zéro. La radicalité provient du lemme de Seidenberg, car chacune de ces équations admet des solutions distinctes. Ces propriétés persistent lorsqu'on spécialise certaines variables. □

La proposition suivante traduit les propositions de bijectivité de la transformée de Fourier (propositions 2.3.3) et d'unicité de l'erreur de poids au plus t (proposition 2.3.4) pour les idéaux :

Proposition 2.4.2 *Soit* $\underline{S}^*$ *le syndrome d'un mot d'erreur de poids* $v \leq t$*, alors les idéaux* $\langle \text{NEWTON}_v^+(\underline{S}^*) \rangle$, $\langle \text{SYNDROM}_v^+(\underline{S}^*) \rangle$ *et* $\langle \text{SYNSYM}_v^+(\underline{S}^*) \rangle$ *possèdent une unique solution (à permutation près des* Z_j*).*

Démonstration Il est évident que si $\langle \text{SYNDROM}_v^+(\underline{S}^*) \rangle$ possède une unique solution, alors $\langle \text{SYNSYM}_v^+(\underline{S}^*) \rangle$ aussi. Or, si $(\underline{Z}_v^*)$ est une solution du premier système, alors par bijectivité de la transformée de Fourier (proposition 2.3.3), les Z_j^* sont non nuls et sont exactement les localisateurs de l'erreur de syndrome $\underline{S}^*$.

Pour l'idéal $\langle \text{NEWTON}_v^+(\underline{S}^*) \rangle$, la preuve ne nous paraît pas simple, comme dans [CRHT94b], et nous le redémontrons dans la proposition 6.3.1 page 118. □

Ces deux propriétés vont permettre de donner des algorithmes de décodage, à l'aide du lemme suivant :

Lemme 2.4.3 *Soit* $I \subset \mathbb{K}[x_1, \ldots, x_n]$ *un idéal radical possédant une unique solution* $x_1^*, \ldots, x_n^*$*. Soit* G *une base de Gröbner réduite de* I *pour un ordre quelconque, alors* $G = \{x_1 - x_1^*, \ldots, x_n - x_n^*\}$.

Démonstration Comme I possède une unique solution, on a $I \cap \mathbb{K}[x_i] = \langle (x_i - x_i^*)^\lambda \rangle$ pour un entier $\lambda \in \mathbb{N}$, et par radicalité on a $x_i - x_i^* \in I$ pour tout $i \in [1; n]$, donc $I = \langle x_1 - x_1^*, \ldots, x_n - x_n^* \rangle$. □

Corollaire 2.4.4 *Soit* $\underline{S}^*$ *le syndrome d'un mot d'erreur de poids* $v \leq t$*, alors on a les égalités suivantes :*

$$\begin{aligned} \langle \text{NEWTON}_v^+(\underline{S}^*) \rangle &= \langle \sigma_j - \sigma_j^*,\ j \in [1; v],\ S_i - S_i^*,\ i \notin Q \rangle \\ \langle \text{SYNSYM}_v^+(\underline{S}^*) \rangle \cap \mathbb{F}_{2^m}[\underline{\sigma}_v] &= \langle \sigma_j - \sigma_j^*,\ j \in [1; v] \rangle \end{aligned}$$

[5] i.e. $f^\lambda \in I$ implique $f \in I$.

2.4.2 Décodage en ligne

Décodage en ligne avec le système SYNDROM_v^+**.** Ce système est étudié par Chen et al. [CRHT94c], nous en donnons ici une preuve simple à partir des résultats de la section précédente.

Théorème 2.4.5 ([CRHT94c]) *Soit e une erreur de poids $1 \leq v \leq t$, $L(Z)$ le polynôme localisateur de e, $\underline{S}^*$ le syndrome de e. Alors*

$$\langle \text{SYNDROM}_w^+(\underline{S}^*)\rangle \cap \mathbb{F}_{2^m}[Z_1] \quad = \quad \begin{cases} \langle 1\rangle & si\ w < v, \\ \langle L(Z_1)\rangle & si\ w = v, \\ \langle Z_1 \cdot L(Z_1)\rangle & si\ w = v+1, \\ \langle Z_1^{n+1} - Z_1\rangle & si\ w \geq v+2. \end{cases}$$

Démonstration Notons $I = \langle \text{SYNDROM}_w^+(\underline{S}^*)\rangle$ et $Z_1^*, \ldots, Z_v^*$ les localisateurs de e. Soit $(z_1^*, \ldots, z_w^*)$ une solution quelconque de SYNDROM_w^+. Notons u le nombre de z_i^* non nuls, à permutation près on peut supposer que ce sont les u premiers, et on a alors $z_1^* = \alpha^{i_1}, \ldots, z_u^* = \alpha^{i_u}, z_{u+1}^* = 0, \ldots, z_w^* = 0$. L'erreur $a = x^{i_1} + \ldots + x^{i_u}$ est ainsi de poids au plus t et a pour syndrome $\underline{S}^*$, donc d'après la proposition 2.3.4 on a $a = e$, $v \leq u \leq w$ et les z_i^* qui ne sont pas égaux à un Z_j^* sont égaux deux à deux.

On en déduit que, si $w < v$, alors I n'a aucune solution. Si $w = v$, les solutions de I sont exactement les racines de $L(Z)$. Pour $w = v+1$ les solutions de I sont exactement les racines de $L(Z)$ et 0, et enfin pour $w \geq v+2$ tout Z^* racine de $Z^{n+1} + Z$ est solution de I. Les égalités d'idéaux proviennent du fait que I est un idéal radical. □

Le système des équations de Newton, décodage en ligne. L'égalité pour le système $\langle \text{NEWTON}_v^+(\underline{S}^*)\rangle$ du corollaire 2.4.4 est donné par Chen et al. [CRHT94b], mais avec une preuve différente. Cette proposition est complétée par C. Rong dans [RH99], ce qui permet de déterminer le poids de e :

Proposition 2.4.6 *Si e est une erreur de poids*[6] *$1 \leq v \leq t$ et de syndrome $\underline{S}^*$ alors :*

$$\begin{aligned} \langle \text{NEWTON}_w^+(\underline{S}^*)\rangle \cap \mathbb{F}_{2^m}[\underline{\sigma}_w] &= \langle 1\rangle,\ si\ w < v \\ \langle \text{NEWTON}_t^+(\underline{S}^*)\rangle \cap \mathbb{F}_{2^m}[\underline{\sigma}_v] &= \langle \sigma_1 - \sigma_1^*, \ldots, \sigma_v - \sigma_v^*\rangle, \\ si\ v \leq \lfloor \frac{t-1}{2} \rfloor\ alors\ \langle \text{NEWTON}_t^+(\underline{S}^*)\rangle \cap \mathbb{F}_{2^m}[\sigma_{v+1}] &= \langle \sigma_{v+1}^{2^m} - \sigma_{v+1}\rangle \\ et\ \langle \text{NEWTON}_t^+(\underline{S}^*)\rangle \cap \mathbb{F}_{2^m}[\sigma_{2v+1}] &= \langle \sigma_{2v+1}^{2^m} - \sigma_{2v+1}\rangle, \\ si\ v > \lfloor \frac{t-1}{2} \rfloor\ alors\ \langle \text{NEWTON}_t^+(\underline{S}^*)\rangle \cap \mathbb{F}_{2^m}[\sigma_j] &= \langle \sigma_j\rangle,\ \forall\ v+1 \leq j \leq t. \end{aligned}$$

[6] la proposition n'est plus vraie pour $v = 0$

Algorithmique Pour décoder pratiquement une erreur de syndrome $\underline{S}^*$, il suffit de calculer une base de Gröbner réduite G du système spécialisé $\textsc{Newton}_t^+(\underline{S}^*)$ pour un ordre quelconque (en pratique on utilise l'ordre grevlex, pour lequel les calculs sont les plus rapides). On peut déterminer v de la manière suivante : on pose $w = 0$, et tant que $G = \langle 1 \rangle$ on incrémente w. On obtient finalement $v = w$ et $G \cap \mathbb{F}_{2^m}[\sigma_1, \ldots, \sigma_v] = \{\sigma_1 - \sigma_1^*, \ldots, \sigma_v - \sigma_v^*\}$ (avec $\sigma_v^* \neq 0$). Les fonctions symétriques élémentaires des localisateurs sont les σ_j^*. Il est également possible de partir de $w = t$ et de décrémenter w tant que $\sigma_w = 0$ annule les polynômes de G.

De la même manière si l'on veut utiliser le système $\langle \textsc{Syndrom}_w^+(\underline{S}^*) \rangle$, il suffit d'en calculer une base de Gröbner G pour un ordre d'élimination $\{Z_j, j > 1\} > Z_1$. On détermine v en incrémentant w à partir de $w = 0$ tant que $G = \langle 1 \rangle$ ou en décrémentant w à partir de $w = t$ tant que $Z_1 = 0$ est solution du système, et pour $w = v$ on obtient le polynôme localisateur.

Nous obtenons ainsi un algorithme de décodage, décrit ici pour $\textsc{Newton}_v^+$ mais valable de la même manière pour $\textsc{Syndrom}_v^+$ ou $\textsc{SynSym}_v^+$:

Algorithme 1 (Décodage en ligne à partir des équations de Newton)
Pour chaque mot $\tilde{c}$ reçu,

- *calculer le syndrome de l'erreur $\underline{S}^* = \{\tilde{c}(\alpha^i) \ : \ i \in Q\}$,*
- *calculer la base de Gröbner réduite de $\langle \textsc{Newton}_w^+(\underline{S}^*) \rangle$ pour un ordre grevlex et pour $w = 0, \ldots, v$, en déduire v le poids de l'erreur et les polynômes $\sigma_j - \sigma_j^*$ pour $1 \leq j \leq v$,*
- *calculer les racines du polynôme localisateur de l'erreur.*

Exemple Prenons l'exemple du code RQ $[31, 16, 7]$, et essayons de décoder successivement les erreurs $e_1 = x^2$, $e_2 = x^2 + x^4$ et $e_3 = x^2 + x^4 + x^7$. Calculons pour ces trois erreurs e_k, $k \in [1; 3]$ une base de Gröbner réduite G_N^k de $\langle \textsc{Newton}_3^+(\underline{S}^*) \rangle$, G_S^k de $\langle \textsc{Syndrom}_k^+(\underline{S}^*) \rangle \cap \mathbb{F}_{2^m}[Z_1]$ et G_σ^k de $\langle \textsc{SynSym}_3^+(\underline{S}^*) \rangle \cap \mathbb{F}_{2^m}[\underline{\sigma}_3]$. On obtient (avec α racine de $x^5 + x^2 + 1$) :

$$\left\{ \begin{array}{rcl} G_N^1 = G_\sigma^1 & = & \{\sigma_1 + \alpha^2, \sigma_2 + (1 + \alpha^3)\sigma_3, \sigma_3^{32} + \sigma_3\} \\ G_N^2 = G_\sigma^2 & = & \{\sigma_1 + \alpha^2 + \alpha^4, \sigma_2 + \alpha + \alpha^3, \sigma_3\} \\ G_N^3 = G_\sigma^3 & = & \{\sigma_1, \sigma_2 + 1 + \alpha + \alpha^2 + \alpha^4, \sigma_3 + \alpha^2 + \alpha^3 + \alpha^4\} \end{array} \right.$$

et

$$\left\{ \begin{array}{rcl} G_S^1 & = & \{Z_1 + \alpha^2\} \\ G_S^2 & = & \{Z_1^2 + (\alpha^2 + \alpha^4)Z_1 + \alpha + \alpha^3\} \\ G_S^3 & = & \{Z_1^3 + (1 + \alpha + \alpha^2 + \alpha^4)Z_1 + \alpha^2 + \alpha^3 + \alpha^4\} \end{array} \right.$$

On vérifie bien par exemple que la seule solution pour l'erreur e_3 est $\sigma_1 = 0$, $\sigma_2 = 1 + \alpha + \alpha^2 + \alpha^4$, $\sigma_3 = \alpha^2 + \alpha^3 + \alpha^4$ ce qui donne comme polynôme localisateur $Z^3 + (1 + \alpha + \alpha^2 + \alpha^4)Z + \alpha^2 + \alpha^3 + \alpha^4 = (Z - \alpha^2)(Z - \alpha^4)(Z - \alpha^7)$.

2.4.3 Décodage formel

Décodage formel avec le système $\textsc{Syndrom}_v^+$**.** L'approche est introduite par Chen et al. [CRHT94a] et les preuves sont données par Loustaunau et York [LVY97]. Une étude détaillée de la forme de la base de Gröbner de l'idéal $\langle \textsc{Syndrom}_v^+ \rangle$ est donnée dans [CM02].

Théorème 2.4.7 ([LVY97]) *Soit G une base de Gröbner de $\langle \textsc{Syndrom}_t^+ \rangle$ pour l'ordre lexicographique $Z_1 > \cdots > Z_{t-1} > Z_t > \underline{S}$. Soit $G_k = G \cap \mathbb{F}_2[Z_k, \ldots, Z_t, \underline{S}]$ la base de Gröbner de l'idéal $\langle \textsc{Syndrom}_t^+ \rangle \cap \mathbb{F}_2[Z_k, \ldots, Z_t, \underline{S}]$ d'élimination des $k-1$ premières coordonnées et 0_k le vecteur nul de longueur k.*

Alors une erreur de syndrome $\underline{S}^*$ est de poids $v \leq t$ si et seulement si

$$\forall k \geq v+1,\ \forall g \in G_k,\ g(0_{t-k+1}, \underline{S}^*) = 0 \quad et \quad \exists g \in G_v,\ g(0_{t-v+1}, \underline{S}^*) \neq 0$$

Si $G_v = \{g_1, \ldots, g_u\}$ alors l'idéal principal

$$\langle G_v(Z_v, 0_{t-v}, \underline{S}^*) \rangle = \langle g_1(Z_v, 0_{t-v}, \underline{S}^*), \ldots, g_u(Z_v, 0_{t-v}, \underline{S}^*) \rangle \subset \mathbb{F}_{2^m}[Z_v]$$

est engendré par le polynôme localisateur $L(Z_v)$ de l'erreur, qui est (à un coefficient près dans $\mathbb{F}_{2^m}$) l'un des $g_j(Z_v, 0_{t-v}, \underline{S}^*)$.

Démonstration Nous ne réécrivons pas la preuve qui est bien faite dans [LVY97], et que nous réutiliserons pour prouver le théorème 2.4.8, mais nous donnons les ingrédients qui font marcher la preuve : (1) unicité de la solution, (2) propriété d'élimination des bases de Gröbner, (3) propriété de radicalité des idéaux (que les auteurs oublient de prouver dans [LVY97]), (4) théorème de spécialisation d'une base de Gröbner. □

Remarque. L'idéal $\langle \textsc{Syndrom}_v^+ \rangle$ contient les polynômes linéaires $S_{2i \mod n} + S_i^2$. Il suffit donc de considérer le système $\textsc{Syndrom}_v^+$ en ne prenant qu'un représentant S_i par classe cyclotomique, le théorème 2.4.7 reste valable, mais le système possède beaucoup moins de variables.

Remarque. $\textsc{Syndrom}_v^+$ est une base de Gröbner de $\langle \textsc{Syndrom}_v^+ \rangle$ pour l'ordre lexicographique $S > Z$, et l'idéal $\langle \textsc{Syndrom}_v^+ \rangle$ est zéro dimensionnel. Il suffit donc d'utiliser un algorithme de changement d'ordre du type FGLM [FGLM93] pour calculer G pour l'ordre lexicographique $Z > S$.

L'algorithme de décodage issu de ce théorème est détaillé à la fin de cette section sur les exemples des codes à résidus quadratiques de longueurs 23 et 31.

Le système $\textsc{SynSym}_t^+$ **avec fonctions symétriques élémentaires, décodage formel.** Nous avons vu que l'étude du système $\textsc{Syndrom}_v^+$ permet d'obtenir une formule symbolique pour le polynôme localisateur de l'erreur. Or, il parait plus facile d'obtenir des formules séparées pour chaque coefficient du polynôme localisateur plutôt qu'une seule formule, ce qui est bien vérifié en pratique. Nous donnons un résultat semblable au théorème 2.4.7 pour le système $\textsc{SynSym}_v^+$, en utilisant les mêmes méthodes de preuve que dans [LVY97].

Theorem 2.4.8 *Soit G une base de Gröbner de $I = \langle \text{SynSym}_t^+ \rangle \cap \mathbb{F}_2[\underline{\sigma_v}, \underline{S}]$ pour l'ordre lexicographique $\sigma_1 > \cdots > \sigma_{t-1} > \sigma_t > \underline{S}$ et notons $G_k = G \cap \mathbb{F}_2[\sigma_k, \ldots, \sigma_t, \underline{S}]$ une base de Gröbner de l'idéal d'élimination des $k-1$ premières coordonnées $I \cap \mathbb{F}_2[\sigma_k, \ldots, \sigma_t, \underline{S}]$.*

Alors une erreur e de syndrome $\underline{S}^*$ est de poids $v \leq t$ si et seulement si

$$\forall k \geq v+1,\ \forall g \in G_k,\ g(0_{t-k+1}, \underline{S}^*) = 0 \quad et \quad \exists g \in G_v,\ g(0_{t-v+1}, \underline{S}^*) \neq 0 \tag{2.7}$$

De plus,

$$\langle G_1(\sigma_1, \sigma_2, \ldots, \sigma_v, 0_{t-v}, \underline{S}^*) \rangle = \langle \sigma_1 - \sigma_1^*, \ldots, \sigma_v - \sigma_v^* \rangle$$

et pour toute spécialisation $(\sigma_{v+1}, \ldots, \sigma_t) = 0_{t-v}$ et $\underline{S} = \underline{S}^*$ en le syndrome d'une erreur de poids v, pour tout $j \in [1; v]$, G_j contient un polynôme de degré 1 en σ_j dont l'initial *(i.e. le coefficient de σ_j) ne s'annule pas lorsque l'on spécialise.*

Démonstration Le schéma de la preuve est le même que pour le système Syndrom_v^+ (théorème 2.4.7), nous en redonnons les grandes étapes. L'équation 2.7 provient du fait que, si $\sigma_1^*, \ldots, \sigma_v^*$ sont les coefficients du polynôme localisateur de e, alors $(\sigma_1^*, \ldots, \sigma_v^*, 0, \ldots, 0)$ est solution de $\langle \text{SynSym}_t^+(\underline{S}^*) \rangle$, que $\langle \text{SynSym}_{v-1}^+(\underline{S}^*) \rangle = \langle 1 \rangle$, ainsi que du théorème d'élimination 1.3.3. Considérons maintenant l'idéal G_j^* qui est l'idéal G_j spécialisé en $(\sigma_{v+1}, \ldots, \sigma_t) = 0_{t-v}$, $\underline{S} = \underline{S}^*$, et $\sigma_{j+1} = \sigma_{j+1}^*, \ldots, \sigma_v = \sigma_v^*$. Alors en utilisant le théorème de spécialisation 1.3.6 il existe un polynôme $g_j \in G_j$ tel que $g_j(\sigma_j, \sigma_{j+1}^*, \ldots, \sigma_v^*, 0_{t-v}, S^*)$ engendre G_j^*. Comme G_j^* n'a qu'une solution $\sigma_j = \sigma_j^*$ et est un idéal radical (grâce aux équations de corps), il est engendré par $\sigma_j - \sigma_j^*$. Ainsi g_j est un polynôme de degré 1 en σ_j dont l'initial ne s'annule pas lorsqu'il est spécialisé. □

Remarque.

1. Ce théorème montre que pour toute spécialisation $(\sigma_{v+1}, \ldots, \sigma_t) = 0_{t-v}$, $\underline{S} = \underline{S}^*$, la base de Gröbner contient un polynôme de degré un en σ_j dont l'initial ne s'annule pas, mais ce polynôme peut différer pour chaque spécialisation,
2. L'initial des polynômes de degré un en σ_j est un polynôme en les variables $\underline{S}$ et σ_k, $k > j$.

Le théorème précédent conduit à l'algorithme de décodage suivant :

Algorithme 2 (Décodage formel à partir des syndromes)

Pré-calcul. *1) Calculer la base de Gröbner G de $\langle \text{SynSym}_t^+ \rangle \cap \mathbb{F}_2[\underline{\sigma}, \underline{S}]$ (l'ordre utilisé est un ordre d'élimination des Z_j, et l'ordre lexicographique $\sigma_1 > \ldots > \sigma_t > \underline{S}$ sur le deuxième bloc de variables, en donnant un poids[7] i aux variables σ_i et S_i),*

[7] Voir la section 1.2.4 page 10 pour une discussion sur les stratégies de choix des ordres monomials.

2) éventuellement sélectionner certaines formules de la forme $P\sigma_j + Q$ avec P et Q des polynômes en $\underline{S}, \sigma_{j+1}, \ldots, \sigma_t$ pour lesquelles on peut prouver (théoriquement ou par recherche exhaustive) que pour chaque erreur de poids $v \leq t$, au moins l'un des initiaux P ne va pas s'annuler pour chaque $j \in [1; v]$.

Décodage. *Pour chaque mot à décoder,*

- *calculer le syndrome $\underline{S}^*$,*
- *évaluer G en S^*,*
- *en évaluant les polynômes en $\sigma_t = 0, \sigma_{t-1} = 0, \ldots$ successivement, en déduire le poids v de l'erreur, puis les cœfficients $\sigma_v^*, \ldots, \sigma_1^*$ du polynôme localisateur,*
- *calculer les racines du polynôme localisateur.*

Cette méthode de décodage est illustrée à la fin de cette section sur l'exemple des codes à résidus quadratiques de longueur 23 et 31. L'avantage de l'introduction des fonctions symétriques élémentaires en les localisateurs est que l'idéal $\langle \text{SynSym}_v^+ \rangle$ possède environ $v!$ fois moins de solutions que l'idéal $\langle \text{Syndrom}_v^+ \rangle$, le calcul de la base de Gröbner est donc bien plus rapide (ces deux systèmes sont déjà des bases de Gröbner, on peut donc utiliser un algorithme de changement d'ordre du type [FGLM93] : voir le paragraphe "efficacité des algorithmes" de la section 6.2 pour plus de détails).

Décodage formel à partir du système des équations de Newton. Il est aussi possible d'étudier le système $\text{Newton}_v^+ \subset \mathbb{F}_2[\overline{S}, \underline{\sigma_v}, \underline{S}]$ en considérant cette fois les $S_i,\ i \in Q$ comme des variables. En éliminant les $\{S_k, k \notin Q\}$, on obtient des formules pour les σ_j en fonction des $\{S_i, i \in Q\}$.

Dans [RYT90, RTCY92, RRTC01, CRT94], les auteurs donnent des formules pour le décodage des codes à résidus quadratiques de longueur 31, 41, 47 et 73. Ils expriment les σ_j en fonction des syndromes "à la main" à partir des équations de Newton, et ne donnent pas de technique automatisable.

Si l'on calcule une base de Gröbner de Newton_v^+ pour un ordre par blocs (grevlex, grevlex) (voir page 14), elle contiendra les formules de plus petit degré possible en les σ_j. Ainsi, théoriquement le calcul de la base de Gröbner permet de retrouver automatiquement les formules linéaires pour les σ_j trouvées à la main, mais nous verrons sur deux exemples qu'en pratique il est impossible de faire le calcul de la base de Gröbner, à cause des équations $\sigma_j^{2^m} + \sigma_j$.

2.4.4 Deux exemples détaillés

Exemple 1 : le code de Golay de longueur 23 C'est le code à résidus quadratiques Q_{23} de type $[23, 12, 7]$, de rendement $\frac{12}{23}$, étudié en exemple dans [LVY97, CRHT94a]. Soit $\alpha \in \mathbb{F}_{2^{11}}$ une racine primitive $23^{\text{ème}}$ de l'unité, prenons α racine de $g(x) = x^{11} + x^9 + x^7 + x^6 + x^5 + x + 1$ qui est aussi le générateur de Q_{23}. L'ensemble de définition $Q_{23} = \{r^2 \mod 23\} = \{1, 2, 3, 4, 6, 8, 9, 12, 13, 16, 18\}$

est constitué d'une seule classe cyclotomique. Les calculs ont été effectués avec le logiciel FGb de Jean-Charles Faugère, sur un Pentium 4 à 2.2 GHz. Considérons le système

$$\text{SYNDROM}_3^+ = \left\{ \begin{array}{l} S_1 - Z_1 - Z_2 - Z_3, \\ Z_1^{24} - Z_1, \\ Z_2^{24} - Z_2, \\ Z_3^{24} - Z_3 \end{array} \right\}$$

Une base de Gröbner pour l'ordre lexicographique $Z_1 > Z_2 > Z_3 > S_1$ est obtenue en 9 secondes :

$$G \left\{ \begin{array}{l} Z_1 + Z_2 + Z_3 + S_1 = g_1(Z_1, Z_2, Z_3, S_1), \\ Z_2^{24} + Z_2, \\ Z_2^2(Z_3 + S_1) + Z_2(Z_3^2 + S_1^2) + Z_3^2 S_1 + Z_3 S_1^2 + S_1^{256} + S_1^3 = g_{2,1}(Z_2, Z_3, S_1), \\ (S_1^{24} + S_1)[Z_2^2 + Z_2(Z_3 + S_1) + Z_3^2 + Z_3 S_1 + \boldsymbol{f_1}(S_1)] = g_{2,2}(Z_2, Z_3, S_1), \\ Z_3^{24} + Z_3, \\ (S_1^{24} + S_1)[Z_3^3 + Z_3^2 S_1 + Z_3 \boldsymbol{f_1}(S_1) + \boldsymbol{f_2}(S_1)] = g_3(Z_3, S_1), \\ S_1^{2048} + S_1 = c(S_1) \end{array} \right.$$

de dimension 0, degré 13824, où f_1 (resp. f_2) est un polynôme univarié de degré 1313 (resp. 1314) comportant 13 termes (resp. 15) :
$\boldsymbol{f_1} = S_1^{25}(S_1^{1288} + S_1^{1265} + S_1^{1127} + S_1^{1012} + S_1^{759} + S_1^{506} + S_1^{391} + S_1^{368} + S_1^{299} + S_1^{138} + S_1^{46} + S_1^{23} + 1)$
$\boldsymbol{f_2} = S_1^3(S_1^{1311} + S_1^{1288} + S_1^{1150} + S_1^{1035} + S_1^{782} + S_1^{529} + S_1^{414} + S_1^{391} + S_1^{322} + S_1^{253} + S_1^{161} + S_1^{69} + S_1^{46} + S_1^{23} + 1)$.

En gardant les notations du théorème 2.4.7, nous avons

$$\begin{array}{rcl} G_3 & = & \{Z_3^{24} + Z_3, g_3(Z_3, S_1), c(S_1)\}, \\ G_2 \backslash G_3 & = & \{Z_2^{24} + Z_2, g_{2,1}(Z_2, Z_3, S_1), g_{2,2}(Z_2, Z_3, S_1)\}, \\ G_1 \backslash G_2 & = & \{g_1(Z_1, Z_2, Z_3, S_1)\}. \end{array}$$

Comme la solution vérifie toujours l'équation $Z_i^{24} = Z_i$, d'après le théorème 2.4.7, une erreur de syndrome S_1^* est de poids 3 si et seulement si $g_3(0, S_1^*) \neq 0$: le terme constant en Z_3 ne peut être nul. Ainsi, l'inégalité $(S_1^{24} + S_1) f_2(S_1) \neq 0$ caractérise les syndromes S_1 correspondant à une erreur de poids 3, et dans ce cas le polynôme localisateur de l'erreur est $g_3(Z, S_1)$.

De même, une erreur de syndrome S_1^* est de poids 2 si et seulement si $g_3(0, S_1^*) = 0$ et $(g_{2,1}(0, 0, S_1^*) \neq 0$ ou $g_{2,2}(0, 0, S_1^*) \neq 0)$, soit

$$(S_1^{24} + S_1) f_2(S_1) = 0 \text{ et } \left\{ \begin{array}{ll} & S_1^3(S_1^{253} + 1) \neq 0 \\ \text{ou} & (S_1^{24} + S_1) f_1(S_1) \neq 0 \end{array} \right. \tag{2.8}$$

Comme $\text{pgcd}(S_1^3(S_1^{253} + 1), (S_1^{24} + S_1) f_1(S_1)) = S_1^2(S_1^{24} + S_1)$, il est nécessaire que $S_1 \neq 0$ et $S_1^{24} + S_1 \neq 0$ pour que l'une des inégalités de l'équation (2.8) au moins soit vérifiée. L'égalité à zéro se simplifie alors en $f_2(S_1)/S_1^3 = 0$, et les deux inégalités en

$(S_1^{24}+S_1) \neq 0$. Finalement, les erreurs de poids 2 sont caractérisées par $f_2(S_1)/S_1^3 = 0$, et $S_1^{24} + S_1 \neq 0$ et le polynôme localisateur est donné par $g_{2,1}(Z, 0, S_1)/S_1 = Z^2 + S_1 Z_2 + S_1^{255} + S_1^2$.

Une erreur de syndrome S_1^* est de poids 1 si et seulement si $\{(S_1^{24}+S_1)f_2(S_1) = 0, (S_1^{24}+S_1)f_1(S_1) = 0, S_1^3(S_1^{253}+1) = 0\}$ et $S_1 \neq 0$. Le pgcd des trois polynômes qui doivent s'annuler vaut $S_1^3(S_1^{23}+1)$, donc on obtient comme caractérisation des erreurs de poids 1 le système $\{S_1^{23}+1 = 0$ et $S_1 \neq 0\}$. Le polynôme localisateur est donné par $g_1(Z, 0, 0, S_1) = Z + S_1$. On en déduit le schéma de décodage de la figure 2.2 pour le code de Golay.

Calculer le syndrome S_1^*, puis suivre le schéma suivant :

$S_1^* = 0\,?$ $\xrightarrow[\text{oui}]{}$ pas d'erreur,

↓ non

$S_1^{*23} + 1 = 0\,?$ $\xrightarrow[\text{oui}]{}$ erreur de poids 1, $L(Z) = Z + S_1^*$

↓ non

$f_2(S_1^*)/S_1^{*3} = 0\,?$ $\xrightarrow[\text{oui}]{}$ erreur de poids 2, $L(Z) = Z^2 + S_1^* Z_2 + S_1^{*255} + S_1^{*2}$

↓ non

erreur de poids 3, $L(Z) = g_3(Z, S_1^*)$

FIG. 2.2 – Algorithme de décodage du code de Golay de longueur 23.

Étudions maintenant le système avec fonctions symétriques élémentaires :

$$\text{SYNSYM}_3^+ = \left\{ \begin{array}{l} S_1 - Z_1 - Z_2 - Z_3, \\ \sigma_1 - Z_1 - Z_2 - Z_3, \\ \sigma_2 - Z_1Z_2 - Z_1Z_3 - Z_2Z_3, \\ \sigma_3 - Z_1Z_2Z_3, \\ Z_1^{24} - Z_1, \\ Z_2^{24} - Z_2, \\ Z_3^{24} - Z_3 \end{array} \right\}$$

La base Lex du système $\langle \text{SYNSYM}_3^+ \rangle \cap \mathbb{F}_2[\underline{\sigma_3}, S_1]$ pour l'ordre $\sigma_3 > \sigma_2 > \sigma_1 > S_1$ se calcule en 2 secondes :

$$G = \left\{ \begin{array}{l} \sigma_1 + S_1, \\ \sigma_3 + \sigma_2 S_1 + {S_1}^{256} + {S_1}^3, \\ {\sigma_2}^{24} + \sigma_2 + \boldsymbol{f_3}(S_1), \\ ({S_1}^{24} + S_1)(\sigma_2 + \boldsymbol{f_1}(S_1)), \\ {S_1}^{2048} + S_1 \end{array} \right.$$

de dimension 0, degré 2600, où $\boldsymbol{f_1}(S_1)$ est la même fonction que précédemment, et $\boldsymbol{f_3}(S_1) = (S_1^{24} + S_1)({S_1}^{1312} + {S_1}^{1289} + {S_1}^{1105} + {S_1}^{1036} + {S_1}^{576} + {S_1}^{553} + {S_1}^{392} + {S_1}^{277} + {S_1}^{208} + {S_1}^{162} + {S_1}^{139} + {S_1}^{116} + {S_1}^{93} + {S_1}^{24} + S_1)$. Nous retrouvons par le calcul l'algorithme de la figure 2.2.

En ce qui concerne le système des équations de Newton, nous n'avons pas réussi à calculer directement une base de Gröbner de $\langle \textsc{Newton}_3^+ \rangle \cap \mathbb{F}_2[\underline{\sigma_3}, S_1]$. Par contre, en mettant de côté dans un premier temps les équations de corps, nous obtenons en quelques secondes une base de Gröbner de l'idéal des équations de Newton sans équations de corps pour l'ordre Lex $\sigma_3 > \sigma_2 > \sigma_1 > S_1$ (les fonctions puissances inconnues et des syndromes de la même classe cyclotomique que S_1 sont des fonctions linéaires en S_1), et on vérifie[8] qu'en ajoutant l'équation de corps $\sigma_2^{2048} + \sigma_2$ on obtient bien une base de Gröbner de $\langle \textsc{Newton}_3^+ \rangle \cap \mathbb{F}_2[\underline{\sigma_3}, S_1]$ pour l'ordre lexicographique $\sigma_3 > \sigma_2 > \sigma_1 > S_1$:

$$\left\{ \begin{array}{l} \sigma_1 + S_1, \\ \sigma_3 + \sigma_2 S_1 + {S_1}^{256} + {S_1}^3, \\ \sigma_2^{2048} + \sigma_2, \\ ({S_1}^{24} + S_1)(\sigma_2 + f_1(S_1)), \\ {S_1}^{2048} + S_1 \end{array} \right.$$

Le système est de dimension 0 et de degré 51200. Remarquons que $\langle \textsc{Newton}_3^+ \rangle \cap \mathbb{F}_2[\underline{\sigma_3}, S_1]$ est différent de $\langle \textsc{SynSym}_3^+ \rangle \cap \mathbb{F}_2[\underline{\sigma_3}, S_1]$.

Exemple 2 : le code à résidus quadratiques de longueur 31 Le corps des syndromes est $\mathbb{F}_{2^5}$, et l'ensemble de définition est formé de 3 classes cyclotomiques, celles de S_1, S_5 et S_7. Le système

$$\textsc{SynSym}_3^+ = \left\{ \begin{array}{l} S_1 - Z_1 - Z_2 - Z_3, \\ S_5 - Z_1^5 - Z_2^5 - Z_3^5, \\ S_7 - Z_1^7 - Z_2^7 - Z_3^7, \\ \sigma_1 - Z_1 - Z_2 - Z_3, \\ \sigma_2 - Z_1Z_2 - Z_1Z_3 - Z_2Z_3, \\ \sigma_3 - Z_1Z_2Z_3, \\ Z_1^{32} - Z_1, \\ Z_2^{32} - Z_2, \\ Z_3^{32} - Z_3 \end{array} \right\}$$

pour un ordre d'élimination des Z_i et l'ordre lexicographique $\sigma_1 > \sigma_2 > \sigma_3 > S_7 > S_5 > S_1$ donne en moins de 7 secondes la base de Gröbner réduite de 9 polynômes de la figure 2.3. L'idéal est de dimension 0 et de degré 5984. Il contient bien des

[8] par exemple en utilisant la caractérisation d'une base de Gröbner par les S-polynômes, voir proposition 1.2.11 page 9 : il n'y a qu'une seule paire critique à considérer, celle de $g_1 = \sigma_2^{2048} + \sigma_2$ et $g_2 = ({S_1}^{24} + S_1)(\sigma_2 + f_1(S_1))$, soit

$$\begin{array}{lll} S(g_1, g_2) & = S_1\sigma_2^{2048} + (S_1^{24} + S_1)f_1(S_1)\sigma_2^{2047} + S_1^{24}\sigma_2 & \underset{g_1}{\rightarrow} (S_1^{24} + S_1)f_1(S_1)\sigma_2^{2047} + (S_1^{24} + S_1)\sigma_2 \\ & \underset{g_2}{\rightarrow} (S_1^{24} + S_1)f_1(S_1)^2\sigma_2^{2046} + (S_1^{24} + S_1)\sigma_2 & \underset{g_2}{\rightarrow} \dots \\ & \underset{g_2}{\rightarrow} (S_1^{24} + S_1)f_1(S_1)^{2046}\sigma_2^2 + (S_1^{24} + S_1)\sigma_2 & \underset{g_2}{\rightarrow} (S_1^{24} + S_1)(f_1(S_1)^{2047} + 1)\sigma_2 \\ & \underset{g_2}{\rightarrow} (S_1^{24} + S_1)(f_1(S_1)^{2048} + f_1(S_1)) & \underset{S_1^{2048} + S_1}{\rightarrow} 0 \end{array}$$

$$
\begin{cases}
\boldsymbol{\sigma_3}+\sigma_2 S_1 + S_7^4S_5^{30}S_1^{11} + S_7^4S_5^{29}S_1^{16} + S_7^4S_5^{28}S_1^{21} + S_7^4S_5^{27}S_1^{26} + S_7^4S_5^{26}S_1^{31} + S_7^4S_5^{26} + S_7^4S_5^{25}S_1^5 + S_7^4S_5^{21}S_1^{25} + \\
S_7^4S_5^{20}S_1^{30}+S_7^4S_5^{14}S_1^{29}+S_7^4S_5^{10}S_1^{18}+S_7^4S_1^6+S_7^3S_5^{27}S_1^2+S_7^3S_5^{26}S_1^7+S_7^3S_5^{24}S_1^{17}+S_7^3S_5^{23}S_1^{22}+S_7^3S_5^{20}S_1^6+S_7^3S_5^{19}S_1^{11}+ \\
S_7^3S_5^{18}S_1^{16}+S_7^3S_5^{17}S_1^{21}+S_7^3S_5^{16}S_1^{26}+S_7^3S_5^{15}S_1^{31}+S_7^3S_5^{14}S_1^5+S_7^3S_5^{10}S_1^{25}+S_7^3S_5^7S_1^9+S_7^3S_5^3S_1^{29}+S_7^3S_5^2S_1^3+S_7^2S_5^{30}S_1^{25}+ \\
S_7^2S_5^{29}S_1^{30} + S_7^2S_5^{24}S_1^{24} + S_7^2S_5^{23}S_1^{29} + S_7^2S_5^{22}S_1^3 + S_7^2S_5^{21}S_1^8 + S_7^2S_5^{18}S_1^{23} + S_7^2S_5^{15}S_1^7 + S_7^2S_5^{11}S_1^{27} + S_7^2S_5^{10}S_1 + \\
S_7^2S_5^9S_1^6 + S_7^2S_5^8S_1^{11} + S_7^2S_5^7S_1^{16} + S_7^2S_5^6S_1^{21} + S_7^2S_5^4S_1^{31} + S_7^2S_5^4 + S_7^2S_5^3S_1^5 + S_7S_5^{30}S_1 + S_7S_5^{28}S_1^{11} + S_7S_5^{25}S_1^{26} + \\
S_7S_5^{24}S_1^{31} + S_7S_5^{20}S_1^{20} + S_7S_5^{14}S_1^{19} + S_7S_5^{13}S_1^{24} + S_7S_5^{11}S_1^3 + S_7S_5^5S_1^2 + S_7S_5^3S_1^{12} + S_7S_5^2S_1^{17} + S_7S_1^{27} + S_5^{31}S_1^3 + \\
S_5^{30}S_1^8+S_5^{29}S_1^{13}+S_5^{28}S_1^{18}+S_5^{26}S_1^{28}+S_5^{25}S_1^2+S_5^{22}S_1^{17}+S_5^{21}S_1^{22}+S_5^{20}S_1^{27}+S_5^{18}S_1^6+S_5^{15}S_1^{21}+S_5^8S_1^{25}+S_5^6S_1^4+S_5^5S_1^9, \\
\boldsymbol{\sigma_2^{32}} + \sigma_2, \\
\boldsymbol{\sigma_2(S_7 + S_1^7)}+S_7^4S_5^{26}S_1^6+S_7^4S_5^{25}S_1^{11}+S_7^4S_5^{15}S_1^{30}+S_7^4S_5^{13}S_1^9+S_7^4S_5^{12}S_1^{14}+S_7^4S_5^9S_1^{29}+S_7^4S_5^8S_1^3+S_7^4S_5^6S_1^{13}+ \\
S_7^4S_5^5S_1^{18} + S_7^4S_5^4S_1^{23} + S_7^3S_5^{30}S_1^{24} + S_7^3S_5^{29}S_1^{29} + S_7^3S_5^{26}S_1^{13} + S_7^3S_5^{25}S_1^{18} + S_7^3S_5^{24}S_1^{23} + S_7^3S_5^{23}S_1^{28} + S_7^3S_5^{22}S_1^2 + \\
S_7^3S_5^{21}S_1^7+S_7^3S_5^{20}S_1^{12}+S_7^3S_5^{19}S_1^{17}+S_7^3S_5^{18}S_1^{22}+S_7^3S_5^{16}S_1+S_7^3S_5^{14}S_1^{11}+S_7^3S_5^{12}S_1^{21}+S_7^3S_5^5S_1^{25}+S_7^3S_5^2S_1^9+S_7^3S_1^{19}+ \\
S_7^2S_5^{30}S_1^{31}+S_7^2S_5^{30}+S_7^2S_5^{29}S_1^5+S_7^2S_5^{26}S_1^{20}+S_7^2S_5^{25}S_1^{25}+S_7^2S_5^{23}S_1^4+S_7^2S_5^{22}S_1^9+S_7^2S_5^{21}S_1^{14}+S_7^2S_5^{18}S_1^{29}+S_7^2S_5^{15}S_1^{13}+ \\
S_7^2S_5^{11}S_1^2+S_7^2S_5^6S_1^{27}+S_7^2S_5^4S_1^6+S_7^2S_5^2S_1^{16}+S_7S_5^{30}S_1^7+S_7S_5^{28}S_1^{17}+S_7S_5^{24}S_1^6+S_7S_5^{21}S_1^{21}+S_7S_5^{19}S_1^{31}+S_7S_5^{18}S_1^5+ \\
S_7S_5^{17}S_1^{10} + S_7S_5^{15}S_1^{20} + S_7S_5^{12}S_1^4 + S_7S_5^9S_1^{19} + S_7S_5^8S_1^{24} + S_7S_5^7S_1^{29} + S_7S_5^6S_1^3 + S_7S_5^2S_1^{23} + S_7S_1^2 + S_5^{29}S_1^{19} + \\
S_5^{27}S_1^{29}+S_5^{25}S_1^8+S_5^{24}S_1^{13}+S_5^{22}S_1^{23}+S_5^{21}S_1^{28}+S_5^{20}S_1^2+S_5^{17}S_1^{17}+S_5^{16}S_1^{22}+S_5^{15}S_1^{27}+S_5^{12}S_1^{11}+S_5^{11}S_1^{16}+S_5^9S_1^{26}+ \\
S_5^8S_1^{31} + S_5^5S_1^{15} + S_5^3S_1^{25} + S_5^2S_1^{30} + S_1^9, \\
\boldsymbol{\sigma_2(S_5 + S_1^5)}+S_7^4S_5^{30}S_1^{15} + S_7^4S_5^{28}S_1^{25} + S_7^4S_5^{26}S_1^4 + S_7^4S_5^{25}S_1^9 + S_7^4S_5^{24}S_1^{14} + S_7^4S_5^{22}S_1^{24} + S_7^4S_5^{19}S_1^8 + \\
S_7^4S_5^{17}S_1^{18} + S_7^4S_5^{16}S_1^{23} + S_7^4S_5^{14}S_1^2 + S_7^4S_5^{13}S_1^7 + S_7^4S_5^{12}S_1^{12} + S_7^4S_5^{11}S_1^{17} + S_7^4S_5^{10}S_1^{22} + S_7^4S_5^9S_1^{27} + S_7^4S_5^8S_1 + \\
S_7^4S_5^7S_1^6+S_7^4S_5^4S_1^{21}+S_7^4S_5^3S_1^{26}+S_7^4S_5^2S_1^{31}+S_7^4S_1^{10}+S_7^3S_5^{27}S_1^6+S_7^3S_5^{26}S_1^{11}+S_7^3S_5^{21}S_1^5+S_7^3S_5^{17}S_1^{25}+S_7^3S_5^{16}S_1^{30}+ \\
S_7^3S_5^{15}S_1^4+S_7^3S_5^{13}S_1^{14}+S_7^3S_5^{11}S_1^{24}+S_7^3S_5^9S_1^3+S_7^3S_5^7S_1^{13}+S_7^3S_5^6S_1^{18}+S_7^3S_5^3S_1^2+S_7^3S_1^{17}+S_7^2S_5^{30}S_1^{29}+S_7^2S_5^{29}S_1^3+ \\
S_7^2S_5^{27}S_1^{13} + S_7^2S_5^{24}S_1^{28} + S_7^2S_5^{23}S_1^2 + S_7^2S_5^{22}S_1^7 + S_7^2S_5^{20}S_1^{17} + S_7^2S_5^{18}S_1^{27} + S_7^2S_5^{15}S_1^{11} + S_7^2S_5^{14}S_1^{16} + S_7^2S_5^{13}S_1^{21} + \\
S_7^2S_5^7S_1^{20}+S_7^2S_5^6S_1^{25}+S_7^2S_5^4S_1^4+S_7^2S_5^2S_1^{14}+S_7^2S_5^{24}+S_7S_5^{25}S_1^{30}+S_7S_5^{24}S_1^4+S_7S_5^{23}S_1^9+S_7S_5^{21}S_1^{19}+S_7S_5^{20}S_1^{24}+ \\
S_7S_5^{18}S_1^3 + S_7S_5^{16}S_1^{13} + S_7S_5^{15}S_1^{18} + S_7S_5^{13}S_1^{28} + S_7S_5^{11}S_1^7 + S_7S_5^{10}S_1^{12} + S_7S_5^9S_1^{17} + S_7S_5^7S_1^{27} + S_7S_5^3S_1^{16} + \\
S_7S_5^2S_1^{21} + S_7S_5S_1^{26} + S_7S_5^{31} + S_7 + S_5^{29}S_1^{17} + S_5^{26}S_1 + S_5^{25}S_1^6 + S_5^{23}S_1^{16} + S_5^{21}S_1^{26} + S_5^{20}S_1^{31} + S_5^{17}S_1^{15} + S_5^{16}S_1^{20} + \\
S_5^{14}S_1^{30} + S_5^{11}S_1^{14} + S_5^6S_1^8 + S_5^5S_1^{13} + S_5^4S_1^{18} + S_5^3S_1^{23} + S_5^2S_1^{28} + S_5S_1^2, \\
\boldsymbol{\sigma_1} + S_1, \\
\boldsymbol{S_7^5}+S_7^4S_5S_1^2 + S_7^3S_5^9 + S_7^3S_5^2S_1^4 + S_7^3S_5S_1^9 + S_7^3S_1^{14} + S_7^2S_5^{16}S_1^3 + S_7^2S_5^4S_1 + S_7^2S_5^2S_1^{11} + S_7^2S_1^{21} + S_7S_5^{17}S_1^5 + \\
S_7S_5^{16}S_1^{10}+S_7S_5^{10}S_1^9+S_7S_5^9S_1^{14}+S_7S_5^8S_1^{19}+S_7S_5^5S_1^3+S_7S_5^3S_1^{13}+S_7S_5^2S_1^{18}+S_7S_1^{28}+S_5^{25}S_1^3+S_5^{24}S_1^8+S_5^{19}S_1^2+ \\
S_5^{18}S_1^7 + S_5^{16}S_1^{17} + S_5^{13}S_1 + S_5^{12}S_1^6 + S_5^7 + S_5^6S_1^5 + S_5^4S_1^{15} + S_5^2S_1^{25} + S_5S_1^{30} + S_1^4, \\
(\boldsymbol{S_7} + S_1^7)((S_5 + S_1^5)^{31} + 1), \\
\boldsymbol{S_5^{32}} + S_5, \\
\boldsymbol{S_1^{32}} + S_1
\end{cases}
$$

FIG. 2.3 – Base Lex de $\langle \text{SYNSYM}_3^+ \rangle \cap \mathbb{F}_2[\sigma, S_7, S_5, S_1]$ pour le code RQ 31.

équations linéaires en les σ_i, et permet donc de résoudre le problème du décodage.

La base de Gröbner du système NEWTON_t^+ se calcule comme précédemment, en mettant de côté dans un premier temps les équations de corps, puis en rajoutant l'équation $\sigma_2^{32} + \sigma_2$. Nous obtenons en moins de 5 secondes exactement le même système que pour l'idéal $\langle \text{SYNSYM}_3^+ \rangle \cap \mathbb{F}_2[\underline{\sigma_3}, \underline{S}]$. Ainsi, pour le code RQ 31 nous avons $\langle \text{NEWTON}_3^+ \rangle \cap \mathbb{F}_2[\underline{\sigma_3}, \underline{S}] = \langle \text{SYNSYM}_3^+ \rangle \cap \mathbb{F}_2[\underline{\sigma_3}, \underline{S}]$ (cela provient probablement du fait que pour ce code, $2^m = n + 1$ et donc les équations $\sigma_j^{n+1} + \sigma_j$ et $\sigma_j^{2^m} + \sigma_j$ sont égales).

2.5 Conclusion

Nous verrons au chapitre 6 section 6.2 une classification des systèmes présentés ici, ainsi qu'une analyse de la complexité du calcul de leur base de Gröbner, montrant qu'il est improbable de réussir à calculer ces bases de Gröbner même pour des codes de taille raisonnable. Nous donnerons d'autres systèmes, de dimension positive, pour lesquels la base de Gröbner est bien plus facile à calculer.

Deuxième partie

Contributions

Chapitre 3

Suites semi-régulières. Complexité de F5.

Dans ce chapitre nous définissons la notion de suites semi-régulières, qui étend celle de suites régulières au cas surdéterminé.

Nous analysons précisément l'algorithme F5-matriciel pour des suites semi-régulières. Nous en déduisons des propriétés des suites semi-régulières (série de Hilbert, indice de régularité, etc.). Nous donnons également une formule *bornant le nombre d'opérations élémentaires de cet algorithme pour des suites régulières en position de Noether, pour l'ordre grevlex.*

Dans tout ce chapitre, sauf Section 3.5, tous les idéaux considérés seront homogènes et de dimension zéro (i.e. engendrés par des polynômes homogènes et possédant un nombre fini de solutions). Nous notons I un idéal de dimension zéro, engendré par m polynômes homogènes $f_1, \ldots, f_m$ de degrés $d_1, \ldots, d_m$ en n variables (ce qui implique $m \geq n$), à coefficients dans un corps $\mathbb{K}$. Rappelons que I_d est l'ensemble des polynômes de I de degré d.

3.1 Introduction, motivations

La motivation essentielle de notre travail était de donner une définition rigoureuse d'une suite surdéterminée "tirée au hasard", et de pouvoir analyser son comportement vis-à-vis d'un calcul de base de Gröbner. Cela provenait entre autres du besoin de comprendre le comportement de suites à coefficients dans le corps fini $\mathbb{F}_2$, "tirées au hasard", et en particulier celles dont on cherche les solutions dans $\mathbb{F}_2$ (qui sont nécessairement surdéterminées, car elles contiennent les équations de corps $x_1^2 + x_1, \ldots, x_n^2 + x_n$), besoin essentiel en cryptographie.

Suites régulières Nous avons vu Section 1.7 la définition de suite régulière, qui répond bien à cette motivation lorsque le nombre de polynômes m est inférieur

au nombre de variables n. Les suites régulières homogènes sont des suites très "prédictibles" : on connaît leur série de Hilbert, la dimension de la variété algébrique associée est nécessairement $n - m$ (il n'existe donc pas de suites régulières pour $m > n$), etc. Les suites régulières sont également caractérisées par l'existence de diviseurs de zéro : une suite régulière vérifie que pour tout $i \in [1; m]$, le $i$ème polynôme f_i n'est pas diviseur de zéro dans $S_n/\langle f_1, \ldots, f_{i-1}\rangle$. Or, dès que $m > n$, f_m (et f_i pour $i > n$) devient inévitablement diviseur de zéro. Ainsi, si on calcule une base de Gröbner de l'idéal en utilisant une stratégie Normale (par degrés croissants), il est évident que lorsqu'on arrive au degré où tous les monômes sont divisibles par le terme de tête d'un élément de la base de Gröbner, il va y avoir des réductions à zéro (et ce degré existe si l'idéal est de dimension zéro) qui traduisent le fait que f_i soit un diviseur de zéro.

Pour étendre la notion de suites régulières à des suites surdéterminées, il est donc nécessaire de tolérer certains diviseurs de zéro dans $S_n/\langle f_1, \ldots, f_{i-1}\rangle$. La définition que nous donnons Section 3.2 de suites semi-régulières impose simplement que f_i soit un diviseur de zéro dans $(S_n)_d$ pour un d suffisamment grand, i.e. que ces diviseurs de zéro apparaissent le plus tard possible lors d'un calcul de base de Gröbner pour une stratégie Normale. Rappelons que, pour un idéal de dimension zéro, le plus petit degré d tel que $\dim(I_d) = \dim((S_n)_d)$, vus comme espaces vectoriels, est exactement celui où la fonction de Hilbert devient nulle, on l'appelle l'indice de régularité (ou régularité) de la fonction de Hilbert et on le note $H(I)$. Nous définissons les suites semi-régulières comme les suites pour lesquelles f_i n'est pas diviseur de zéro dans $(S_n/\langle f_1, \ldots, f_{i-1}\rangle)_d$ pour tout degré d strictement plus petit que $H(I)$. C'est le mieux que l'on puisse espérer, au-delà de ce degré il y aura nécessairement des réductions à zéro lorsque $m > n$. Nous relions Section 3.2.3 notre définition à d'autres étendant la notion de suite régulières [Par00, PR03, Frö85].

Généricité Les suites régulières représentent bien le comportement d'une suite dont les coefficients ont été tirés au hasard. Plus précisément, on peut montrer que, lorsque le corps des coefficients est infini, "presque toute" suite est une suite régulière, où plus rigoureusement que "être une suite régulière" est une propriété générique au sens de la définition suivante :

Définition 3.1.1 *Considérons l'ensemble $E(n, m, d_1, \ldots, d_m)$ des suites $f_1, \ldots, f_m$ de m polynômes de $S_n = \mathbb{K}[x_1, \ldots, x_n]$ en n variables, de degrés $d_1, \ldots, d_m$. Lorsque le corps $\mathbb{K}$ est infini, on dit qu'une propriété des suites est* générique *si l'ensemble des suites vérifiant cette propriété est un ouvert non vide de Zariski, i.e. si elle est vérifiée par toutes les suites de E, sauf un ensemble algébrique*[1] *de codimension au moins un.*

Montrer que la propriété de semi-régularité est générique est une conjecture de Fröberg [Frö85] de 1985, qui n'a pu être démontré jusqu'ici que dans les quelques cas

[1] Un ensemble algébrique est l'ensemble des zéros d'un système de polynômes

particuliers récapitulés Section 1.6. Il est en général très facile de montrer qu'une propriété est vérifiée par toutes les suites sauf un ensemble algébrique, la partie difficile étant de montrer que cet ensemble est de co-dimension au moins un, c'est-à-dire qu'il existe au moins une suite vérifiant la propriété. Nous n'avons pas pu démontrer de nouveaux cas de cette conjecture, mais nous montrons que l'algorithme F5-matriciel, décrit Section 1.5, permet de décider si une suite explicite est semi-régulière ou non.

Analyse de l'algorithme F5-matriciel Nous allons montrer dans ce chapitre, en utilisant l'algorithme F5-matriciel, que les suites semi-régulières prolongent bien la notion de suites régulières, et vérifient essentiellement les mêmes propriétés que les suites régulières. En particulier, les suites semi-régulières homogènes de m équations en n variables de degrés $d_1, \ldots, d_m$ sont exactement celles ayant pour séries de Hilbert[2] :

$$\Big[\prod_{i=1}^{m}(1-z^{d_i})\Big/(1-z)^n\Big]. \tag{3.1}$$

Ce sont exactement les suites appelées "suffisamment génériques" par Fröberg et Hollman [FH94]. Les suites semi-régulières sur $\mathbb{F}_2$ de m équations en n variables de degrés $d_1, \ldots, d_m$ sont exactement celles ayant pour séries de Hilbert :

$$\Big[(1+z)^n\Big/\prod_{i=1}^{m}(1+z^{d_i})\Big] \text{ sur } \mathbb{F}_2 \tag{3.2}$$

A partir de la série de Hilbert, on peut calculer explicitement l'indice de régularité de la suite, qui majore le degré maximal d'un élément d'une base de Gröbner pour un quelconque ordre gradué par le degré. Notons que l'algorithme F5-matriciel est utilisé pour prouver les propriétés des suites semi-régulières, mais que les résultats sont indépendants de tout algorithme de calcul de base de Gröbner, ils ne dépendent que de l'idéal considéré.

Pour les suites régulières en position de Noether, nous donnons Section 3.4 une analyse fine du comportement de l'algorithme F5-matriciel pour l'ordre grevlex. Nous obtenons une borne précise sur le nombre de polynômes obtenus dans la base de Gröbner calculée par l'algorithme F5-matriciel. Nous en déduisons une borne *générale* sur le nombre d'éléments d'une base de Gröbner réduite, et nous obtenons une nouvelle preuve de la borne de Macaulay. Nous donnons également une borne sur le nombre total d'opérations élémentaires (multiplications ou additions) effectuées par l'algorithme F5-matriciel.

Suites affines Dans la plupart des applications, les systèmes étudiés ne sont pas homogènes. En particulier, pour les applications en cryptographie, les équations de corps $x_i^2 + x_i$ ne le sont pas. Nous définissons Section 3.5 les suites semi-régulières

[2] où $\left[\sum_{i\geq 0} a_i z^i\right] = \sum_{i\geq 0} b_i z^i$ avec $b_i = a_i$ si $a_j > 0\ \forall 0 \leq j \leq i$ et $b_i = 0$ sinon.

affines comme les suites dont la partie homogène de plus haut degré est semi-régulière. Cette définition est assez contraignante, mais elle permet de conserver les propriétés des suites semi-régulières homogènes.

Suites semi-régulières sur $\mathbb{F}_2$ Dans le cas de polynômes à coefficients et solutions dans $\mathbb{F}_2$, cette restriction sur le degré des diviseurs de zéro n'est plus suffisante : en effet, tout polynôme d'un idéal comportant les équations de corps $x_1^2 + x_1, \ldots, x_n^2 + x_n$ vérifie $f^2 = f$. Il est alors nécessaire de modifier un peu la définition de suites semi-régulières en tenant compte de l'action du morphisme de Frobenius $f \to f^2$ pour l'adapter à ce cas particulier. A notre connaissance il n'existe pas d'autre définition prolongeant celle de suites régulières dans ce cas.

Un autre problème des suites à coefficients dans $\mathbb{F}_2$ est la notion de généricité : l'ensemble des suites est en effet fini, il faut alors parler de la probabilité qu'une suite soit semi-régulière.

Pour des systèmes à coefficients dans $\mathbb{F}_2$, l'ensemble E des suites possibles est de dimension zéro, et donc la notion de propriété générique de la définition 3.1.1 n'a plus de sens. Nous conjecturons tout de même qu'une suite "tirée au hasard" sera semi-régulière sur $\mathbb{F}_2$, dans le sens ou la proportion de suites semi-régulières tend vers 1 quand n tend vers l'infini.

Organisation du chapitre Dans une première partie (Section 3.2), nous définissons des suites semi-régulières (dans le cas général et sur $\mathbb{F}_2$), nous explicitons le lien entre nos définitions et les autres existantes, et nous énonçons leur propriétés. La section 3.3 est consacrée au calcul de la série de Hilbert des suites semi-régulières à l'aide de l'algorithme F5-matriciel. Nous faisons Section 1.7 une étude fine du comportement de l'algorithme F5-matriciel pour des suites régulières en position de Noether. Nous indiquons enfin Section 3.5 comment définir les suites semi-régulières affines pour conserver toutes les propriétés des suites homogènes.

3.2 Suites semi-régulières

Nous rappelons que toutes les suites considérées dans ce chapitre engendrent des idéaux de dimension zéro. En particulier, pour un idéal I, l'indice de régularité $H(I)$ de la fonction de Hilbert est le plus petit d tel que $\dim(I_d) = \dim(\mathbb{K}[x_1, \ldots, x_n]_d) = \binom{n+d-1}{d}$.

3.2.1 Définitions

Dans cette section, nous définissons les suites semi-régulières dans le cas général, puis dans le cas particulier de suites à coefficients dans le corps fini $\mathbb{F}_2$ avec les équations de corps.

Suites semi-régulières homogènes Nous donnons la définition de suite semi-régulière homogène sur le modèle de celle des suites régulières (définition 1.7.1 page 23) :

Définition 3.2.1 *Une suite homogène* $f_1, \ldots, f_m \subset S_n$ *est* semi-régulière *si les conditions suivantes sont vérifiées :*

- $I = \langle f_1, \ldots, f_m \rangle \neq S_n$,
- *Pour* $i \in [1; m]$, *si* $g_i f_i = 0$ *dans* $S_n/\langle f_1, \ldots, f_{i-1} \rangle$ *et* $\deg(g_i f_i) < H(I)$ *alors* $g_i = 0$ *dans* $S_n/\langle f_1, \ldots, f_{i-1} \rangle$.

Nous verrons Section 3.2.2 que cette définition des suites semi-régulières permet de conserver de nombreuses propriétés des suites régulières.

Cas des idéaux de dimension positive La définition précédente ne s'applique qu'à des systèmes de dimension zéro. Pour unifier cette définition et celle de suites régulières, nous proposons la définition suivante :

Définition 3.2.2 *Soit* $f_1, \ldots, f_m \subset S_n$ *une suite homogène, et* $I = \langle f_1, \ldots, f_m \rangle$. *Notons* D_{reg} *le plus petit degré tel* [3] *que* $\dim(I_d) = \dim((S_n)_d)$. *Alors la suite* $f_1, \ldots, f_m$ *est* semi-régulière *si les conditions suivantes sont vérifiées :*

- $I = \langle f_1, \ldots, f_m \rangle \neq S_n$,
- *Pour* $i \in [1; m]$, *si* $g_i f_i = 0$ *dans* $S_n/\langle f_1, \ldots, f_{i-1} \rangle$ *et* $\deg(g_i f_i) < D_{reg}$ *alors* $g_i = 0$ *dans* $S_n/\langle f_1, \ldots, f_{i-1} \rangle$.

Suites semi-régulières sur $\mathbb{F}_2$ Soit $\{f_1, \ldots, f_m\} \subset \mathbb{F}_2[x_1, \ldots, x_n]$ une suite de m équations en n variables pour lesquelles on cherche les solutions dans $\mathbb{F}_2$. On considère donc ce système auquel on ajoute les équations de corps $\{x_1^2+x_1, \ldots, x_n^2+x_n\}$. Ainsi le système à résoudre contient $m+n$ équations en n variables sur $\mathbb{F}_2$. Notons $I = \langle x_1^2+x_1, \ldots, x_n^2+x_n, f_1, \ldots, f_m \rangle$ et $R_n = \mathbb{F}_2[x_1, \ldots, x_n]/(x_1^2+x_1, \ldots, x_n^2+x_n)$. Remarquons qu'il est équivalent de travailler sur le système $\{x_1^2+x_1, \ldots, x_n^2+x_n, f_1, \ldots, f_m\}$ dans $\mathbb{F}_2[x_1, \ldots, x_n]$ ou sur le système $\{f_1, \ldots, f_m\}$ dans R_n. Dans un souci de simplification, nous travaillerons toujours dans R_n.

La présence des équations de corps pose deux problèmes :

1. les équations de corps sont des polynômes affines,
2. l'application $f \to f^2$ de $\mathbb{F}_2[x_1, \ldots, x_n]/(x_1^2+x_1, \ldots, x_n^2+x_n)$ dans lui-même, appelé morphisme de Frobenius, est en fait l'identité. Ainsi, tout $f \in I$ vérifie $f^2 = f$, et aucune suite n'est semi-régulière selon la définition précédente.

Pour définir les suites semi-régulières sur $\mathbb{F}_2$, il faut donc modifier la définition 3.2.1 en se ramenant à des polynômes homogènes et en tenant compte de l'action du morphisme de Frobenius.

Considérons dans un premier temps des polynômes homogènes, le cas des polynômes affines sera traité Section 3.5. En particulier, la partie homogène de plus

[3] Remarquons que D_{reg} vaut $H(I)$ lorsque l'idéal est de dimension zéro, et ∞ sinon.

haut degré des équations de corps étant x_i^2, notons $R_n^h = \mathbb{F}_2[x_1, \ldots, x_n]/(x_1^2, \ldots, x_n^2)$ l'ensemble des polynômes ayant tous leurs monômes sans facteurs carrés, et considérons le système $\{f_1, \ldots, f_m\}$ dans R_n^h.

Tout polynôme homogène de R_n^h de degré d vérifie la relation $f^2 = 0$. L'indice de régularité $H(I)$ vérifie donc la proposition suivante :

Proposition 3.2.3 *Soit $f_1, \ldots, f_m$ une suite de m polynômes homogènes de R_n^h, et $I = \langle f_1, \ldots, f_m \rangle$. L'indice de régularité de I, qui est le plus petit degré d tel que $\dim(I_d) = \dim((R_n^h)_d) = \dim((R_n)_d) = \binom{n}{d}$, vérifie $H(I) \leq n + 1$.*

Démonstration Grâce aux équations $x_i^2 = 0$, tout monôme de degré $n+1$ vaut zéro dans R_n^h. □

Nous pouvons alors donner la définition de suite semi-régulière sur $\mathbb{F}_2$ dans le cas homogène :

Définition 3.2.4 *Une suite homogène $f_1, \ldots, f_m \subset R_n^h$ est* semi-régulière sur $\mathbb{F}_2$ *si :*
- *$I = \langle f_1, \ldots, f_m \rangle \neq R_n^h$,*
- *pour $i \in [1; m]$, si $g_i f_i = 0$ dans $R_n^h/\langle f_1, \ldots, f_{i-1} \rangle$ et $\deg(g_i f_i) < H(I)$ alors $g_i = 0$ dans $R_n^h/\langle f_1, \ldots, f_{i-1}, f_i \rangle$.*

Remarquons que, pour tenir compte du morphisme de Frobenius, on autorise dans $R_n^h/\langle f_1, \ldots, f_{i-1} \rangle$ non seulement les diviseurs de zéro de grand degré, mais aussi ceux appartenant à $R_n^h/\langle f_1, \ldots, f_{i-1}, f_i \rangle$, en incluant le $i$ème polynôme.

3.2.2 Propriétés des suites semi-régulières

Nous énonçons ici des propriétés des suites semi-régulières, les preuves seront faites Section 3.3 en utilisant l'algorithme F5-matriciel et des séries génératrices.

Proposition 3.2.5 *Soit $f_1, \ldots, f_m$ une suite semi-régulière homogène (dans S_n ou R_n^h), f_i étant de degré d_i. Alors les propriétés suivantes sont vérifiées :*

1. *la suite $f_1, \ldots, f_m \subset S_n$ est semi-régulière dans S_n, si et seulement si sa série de Hilbert est*
$$\Big[\prod_{i=1}^{m}(1 - z^{d_i})/(1 - z)^n\Big], \tag{3.1}$$
la série $S_{m,n}(z) = \sum_{d \geqslant 0} h_{d,m}(n) z^d = \prod_{i=1}^{m}(1 - z^{d_i})/(1 - z)^n$ étant appelée la série génératrice de la suite $f_1, \ldots, f_m$,
2. *la suite $f_1, \ldots, f_m \subset R_n^h$ est semi-régulière sur $\mathbb{F}_2$, si et seulement si sa série de Hilbert est*
$$\Big[(1 + z)^n/\prod_{i=1}^{m}(1 + z^{d_i})\Big], \tag{3.2}$$
la série $S_{m,n}(z) = \sum_{d \geqslant 0} h_{d,m}(n) z^d = (1 + z)^n/\prod_{i=1}^{m}(1 + z^{d_i})$ étant appelée la série génératrice de la suite $f_1, \ldots, f_m$,

3. *toute permutation* $f_{\sigma(1)}, \ldots, f_{\sigma(m)}$ *est une suite semi-régulière,*
4. *lorsque la variété associée est de dimension zéro (i.e.* $m \geq n$ *dans* S_n *et toujours dans* R_n^h*),* $H(I)$ *est caractérisé par*

$$\forall d < H(I), \quad h_{d,m}(n) > 0 \quad \text{et} \quad h_{H(I),m}(n) \leq 0.$$

5. *la suite* $f_1, \ldots, f_m$ *est semi-régulière* $\not\Rightarrow$ *pour tout* $i \in [1; m]$*, la suite* $f_1, \ldots, f_i$ *est semi-régulière.*

Démonstration Les propriétés (1) et (2) seront prouvées Section 3.3, corollaires 3.3.4 page 66 et 3.3.8 page 68. La propriété (3) est un corollaire immédiat de la première. La propriété (4) provient de la définition de $H(I)$. Enfin, la dernière provient du fait que $H(I)$ dépend de la suite $f_1, \ldots, f_m$ tout entière, et que $H(\langle f_1, \ldots, f_i \rangle) \geq H(I)$ en général. Par exemple, la suite $\{f_1 = x_1^2, f_2 = x_1x_2 + x_2^2, f_3 = x_2x_3, f_4 = x_3^2\}$ est semi-régulière ($H(I) = 3$), mais la sous-suite $\{f_1, f_2, f_3\}$ ne l'est pas (l'idéal engendré est de dimension positive, et $x_1 f_3$ se réduit à zéro). □

Remarquons que la propriété (4) donne un moyen de calcul efficace de $H(I)$. Nous donnons Section 3.3.3 page 68 un exemple détaillé (cas de $n+1$ équations quadratiques en n variables).

3.2.3 Lien avec d'autres définitions

Dans le cas général, il existe d'autres définitions, qui étendent la notion de suites régulières au cas surdéterminé. Nous donnons les liens entre ces définitions, celle de suites régulières et notre définition de suites semi-régulières. Pour le cas particulier de suites à coefficients et solutions dans $\mathbb{F}_2$, il n'existe pas à notre connaissance d'autres définitions de suites semi-régulières.

Dans [FH94], les auteurs appellent suites "suffisamment génériques" les suites ayant pour série de Hilbert la série (3.1). D'après la proposition 3.2.5, ce sont exactement les suites semi-régulières. Ces suites sont également appelées suites *willing* dans [MS03b].

Dans [Par00, PR03], les auteurs définissent une autre notion de semi-régularité :

Définition 3.2.6 (Suites semi-régulières^{PR}) *Soit* $I \subset S_n$ *un idéal homogène. Une forme non nulle* $f_i \in S_n$ *de degré* d_i *est dite semi-régulière^{PR} (au sens de Pardue-Richert) sur* S_n/I *si l'application de multiplication* $(S_n/I)_{a-d_i} \xrightarrow{f_i} (S_n/I)_a$ *est une application linéaire de rang plein pour tout* a*. Une suite* $f_1, \ldots, f_m$ *dans* S_n *de degrés* $d_1, \ldots, d_m$ *est dite semi-régulière^{PR} si* f_i *est semi-régulière^{PR} sur* $S_n/\langle f_1, \ldots, f_{i-1} \rangle$ *pour tout* $i \in [1; m]$*.*

Les suites semi-régulières selon notre définition 3.2.1 sont plus générales que les suites semi-régulières^{PR} (ces dernières vérifient que toute sous-suite $f_1, \ldots, f_i$ est aussi semi-régulière^{PR}, mais cette propriété n'est pas vérifiée pour notre définition) :

Proposition 3.2.7 *Toute suite semi-régulière^{PR} est semi-régulière. La suite* $\{f_1 = x_1^2, f_2 = x_1x_2, f_3 = x_2^2\}$ *est semi-régulière mais pas semi-régulière^{PR}.*

Démonstration La première affirmation provient du fait que la série de Hilbert d'une suite semi-régulière^{PR} est la même que cette d'une suite semi-régulière (cf. [PR03]). Pour la réciproque, la suite $\{f_1 = x_1^2, f_2 = x_1x_2, f_3 = x_2^2\}$ est semi-régulière car $H(I) = 2$, mais n'est pas semi-régulière^{PR} : l'application $(S/(f_1))_1 \xrightarrow{f_2} (S/(f_1))_3$ n'est ni injective (x_1 s'envoie sur 0) si surjective (x_2^3 n'a pas de préimage). □

Remarque. La proposition 1 de [Par00] est donc fausse : il n'y a pas équivalence entre les suites semi-régulières^{PR} et celles ayant pour série de Hilbert (3.1).

Corollaire 3.2.8 *Étant donné une suite de* $m \leq n$ *polynômes de* S_n*, les propriétés suivantes sont équivalentes :*

- *la suite est régulière,*
- *la suite est semi-régulière^{PR},*
- *la suite est semi-régulière,*
- *sa série de Hilbert est* $\prod_{i=1}^{m}(1 - z^{d_i}) \Big/ (1 - z)^n$

3.2.4 Algorithme F5-matriciel et suites semi-régulières

Nous renvoyons le lecteur à la section 1.5 pour une présentation de l'algorithme F5-matriciel, transcription matricielle de l'algorithme F5 de Faugère [Fau02]. Cet algorithme applique deux critères, le critère général et le critère de Frobenius, qui vont assurer l'absence de réduction à zéro (jusqu'à un certain degré) pour les suites semi-régulières.

Le théorème suivant est prouvé dans [Fau02] :

Théorème 3.2.9 *Soit* $f_1, \ldots, f_m$ *une suite homogène, et* $<$ *un ordre monomial admissible gradué par le degré, alors* $f_1, \ldots, f_m$ *est une suite régulière si et seulement si il n'y a pas de réduction à zéro au cours de l'algorithme F5-matriciel pour n'importe quel degré.*

Les théorèmes suivants sont l'analogue du précédent pour des suites semi-régulières générales ou sur $\mathbb{F}_2$:

Théorème 3.2.10 *Soit* $f_1, \ldots, f_m$ *une suite homogène, telle que* $\langle f_1, \ldots, f_m \rangle$ *soit de dimension zéro, et* $<$ *un ordre monomial admissible gradué par le degré. Alors,*

- *Si la suite est semi-régulière alors il n'y a pas de réduction à zéro dans l'algorithme F5-matriciel jusqu'au degré* $d = H(I) - 1$*,*
- *Réciproquement, s'il n'y a aucune réduction à zéro dans l'algorithme F5-matriciel jusqu'au degré* $D - 1$*, et si la matrice en degré* D *est de rang plein et est la première matrice à avoir plus de lignes que de colonnes, alors la suite est semi-régulière et son indice de régularité est* $H(I) = D$*.*

Démonstration Soit $g_i \in S_n$ et notons $m_1 > m_2 > \ldots > m_k$ tous les monômes de g_i triés par ordre décroissant, nous avons alors $g_i = a_1 m_1 + a_2 m_2 + \ldots + a_k m_k$ avec $a_i \in \mathbb{K}$. Alors $g_i f_i = 0$ dans $S_n/\langle f_1, \ldots, f_{i-1}\rangle$ équivaut au fait que la ligne étiquetée (m_1, f_i) est combinaison linéaire des lignes situées au dessus d'elle dans la matrice de Macaulay $\mathcal{M}_{d,i}^{\text{acaulay}}$, avec $d = \deg(g_i f_i)$.

Ainsi, si la suite est semi-régulière alors il n'y a pas de réduction à zéro au cours de l'algorithme jusqu'au degré $H(I) - 1$ (les lignes qui se réduisent à zéro dans la matrice $\mathcal{M}_{d,i}^{\text{acaulay}}$ sont celles étiquetées (m, f_i) avec m terme de tête d'un élément de $S_n/\langle f_1, \ldots, f_{i-1}\rangle$, et ce sont exactement les lignes supprimées par le critère général).

Réciproquement, s'il n'y a pas de réduction à zéro jusqu'au degré $D-1$, alors $g_i f_i = 0$ dans $S_n/\langle f_1, \ldots, f_{i-1}\rangle \Longrightarrow g_i = 0$ dans $S_n/\langle f_1, \ldots, f_{i-1}\rangle$ dès que $\deg(g_i f_i) \leq D-1$. Or, comme le degré D est le premier pour lequel tous les monômes sont terme de tête d'un élément de l'idéal, on a $H(I) = D$ et la suite est semi-régulière. □

Dans le cas de suites semi-régulières sur $\mathbb{F}_2$, le théorème est similaire :

Théorème 3.2.11 *Si $f_1, \ldots, f_m$ est une suite semi-régulière sur $\mathbb{F}_2$, alors il n'y a pas de réduction à zéro dans l'algorithme F5-matriciel jusqu'à $d = H(I) - 1$. Réciproquement, s'il n'y a aucune réduction à zéro dans l'algorithme F5-matriciel jusqu'au degré $D-1$, et si la matrice en degré D est de rang plein et a plus de lignes que de colonnes, alors la suite est semi-régulière et son indice de régularité est $H(I) = D$.*

Démonstration La preuve est essentiellement la même que dans le cas général, la seule différence étant que les lignes de la matrice de Macaulay supprimées par les deux critères sont celles étiquetées (m, f_i) avec m terme de tête d'un élément de $S_n/\langle f_1, \ldots, f_i\rangle$. □

Ainsi, l'algorithme F5-matriciel est un algorithme de décision pour la semi-régularité. De plus, si la suite est semi-régulière, on peut prendre $d_{\max} = H(I)$ et l'algorithme F5-matriciel va rendre une base de Gröbner de $\langle f_1, \ldots, f_m\rangle$. Il n'y a aucune réduction à zéro, excepté peut-être dans le dernier degré, donc on peut calculer exactement le nombre de lignes de chaque matrice, ce que nous faisons maintenant à l'aide de séries génératrices.

3.2.5 Existence de suites semi-régulières

L'une des questions les plus importantes qui subsistent est : "combien y a-t-il de suites semi-régulières ?", et même "existe-t-il des suites semi-régulières" ?. La caractérisation des suites semi-régulières comme les suites ayant pour série de Hilbert (3.1) (proposition 3.2.5) montre que cette question rejoint la conjecture 1.6.3 page 22 de Fröberg. Comme nous l'avons rappelé, cette conjecture n'a pu être prouvée que dans un petit nombre de cas particuliers. Elle implique qu'une suite de polynômes aléatoires (dont les coefficients sont tirés aléatoirement) est semi-régulière.

Proposition 3.2.12 *Soit $P(n,d)$ l'espace vectoriel des polynômes de S_n de degré d. Nous définissons $E = E(n,m,d_1,\ldots,d_m) = P(n,d_1)\times P(n,d_2)\times\ldots\times P(n,d_m)$ comme l'espace vectoriel des suites de m polynômes homogènes $f_1,\ldots,f_m$ en n variables de degrés $d_1,\ldots,d_m$. Alors l'ensemble des points de E correspondant à des suites semi-régulières est un ouvert de l'espace E pour la topologie de Zariski.*

Démonstration Notons $U = U(n,m,d_1,\ldots,d_m)$ l'ensemble des suites semi-régulières. Il est équivalent de prouver que l'ensemble $E\backslash U$ des points de E qui ne sont pas des suites semi-régulières est une variété algébrique de E. Nous utilisons pour cela l'algorithme F5-matriciel. Tout élément de $E\backslash U$ vérifie que pour un $d < H(I)$, le rang de la matrice de Macaulay en degré d, $\mathcal{M}_{d,m}^{\text{acaulay}}$, est inférieur au rang de cette matrice pour une suite régulière. Or, ce rang est connu : c'est $M_d(n) - h_{d,m}(n)$, où $M_d(n)$ est le nombre de monômes de degré d, et $h_{d,m}(n)$ est le coefficient de degré d de la série génératrice associée aux suites semi-régulières. Les suites qui ne sont pas semi-régulières sont donc celles dont les coefficients annulent tous les mineurs de la matrice générique $\mathcal{M}_{d,n}^{\text{acaulay}}$ d'ordre $M_d(n)-h_{d,m}(n)$. La réciproque est également vraie, les systèmes de $E\backslash U$ sont en bijection avec ceux dont les coefficients annulent l'un de ces déterminants. Comme les déterminants sont des polynômes en les coefficients de $f_1,\ldots,f_m$, $E\backslash U$ est un ensemble fermé dans E pour la topologie de Zariski. □

La preuve de cette proposition est assez simple, la difficulté étant de montrer que cet ouvert U n'est pas l'ensemble vide. Pour cela, lorsque le corps $\mathbb{K}$ est infini, il suffit d'exhiber un exemple concret de suite $f_1,\ldots,f_n$ dans $E(n,m,d_1,\ldots,d_m)$ qui soit semi-régulière, alors $E\backslash U$ est de co-dimension positive. Le cas de $n+1$ polynômes a déjà été traité dans [Frö85]. Dans cette section nous redonnons une preuve élémentaire (nous montrons qu'il n'y a pas de réduction à zéro dans l'algorithme F5-matriciel avant le degré $H(I)$) de cette conjecture dans le cas de $m = n+1$ équations quadratiques de S_n.

Proposition 3.2.13 *La suite*

$$\left\{x_1^2,\ldots,x_n^2,\sum_{1\le i<j\le n}x_ix_j\right\}\subset S_n$$

de $n+1$ équations quadratiques en n variables est semi-régulière.

La suite $\{\sum_{1\le i<j\le n}x_ix_j\}\subset R_n^h$ de une équation quadratique sur $\mathbb{F}_2$ est semi-régulière sur $\mathbb{F}_2$.

Démonstration Nous donnons les grandes lignes de la preuve : nous allons montrer que toutes les matrices apparaissant au cours de l'algorithme F5-matriciel sont de rang plein. Notons f_n le dernier polynôme. Pour simplifier les calculs, en tenant compte des équations $x_1^2,\ldots,x_n^2$, nous ne considérons, pour chaque matrice, que les lignes et les colonnes correspondant à des monômes de degré au plus un en chaque x_i. Pour les lignes, cela revient à appliquer le critère général, pour les colonnes cela

revient à travailler dans R_n^h. Nous continuons à noter $\mathcal{M}_{d,m}$ la matrice construite par l'algorithme F5-matriciel au degré d pour m polynômes. Si l'on sépare les monômes qui dépendent de x_n de ceux qui ne dépendent que de $x_1, \ldots, x_{n-1}$, on obtient la formule de récurrence :

$$\mathcal{M}_{d,n} = \begin{array}{c} \\ x_n\,(S_{n-1})_{d-3}\,f_n \\ (S_{n-1})_{d-2}\,f_n \end{array} \begin{array}{c} \begin{array}{cc} (S_{n-1})_d & x_n\,(S_{n-1})_{d-1} \end{array} \\ \begin{pmatrix} 0 & \mathcal{M}_{d-1,n-1} \\ \mathcal{M}_{d,n-1} & \mathcal{M}'_{d-1,n-1} \end{pmatrix} \end{array}$$

et

$$\mathcal{M}'_{d,n} = \begin{array}{c} \\ x_n\,(S_{n-1})_{d-2}\,f_n \\ (S_{n-1})_{d-1}\,f_n \end{array} \begin{array}{c} \begin{array}{cc} (S_{n-1})_d & x_n\,(S_{n-1})_{d-1} \end{array} \\ \begin{pmatrix} 0 & \mathcal{M}'_{d-1,n-1} \\ \mathcal{M}'_{d,n-1} & J \end{pmatrix} \end{array}$$

où J est la matrice anti-diagonale. On a également $\mathcal{M}_{2,n} = (1 \ldots 1) = {}^{\mathrm{T}}\mathcal{M}_{n,n}$ (avec $\binom{n}{2}$ colonnes) et $\mathcal{M}'_{1,n} = (1 \ldots 1) = {}^{\mathrm{T}}\mathcal{M}'_{n,n}$ (avec n colonnes).

Nous allons prouver la propriété par récurrence sur n. Pour $n = 2$ la propriété est clairement vraie. Supposons que toutes les matrices sont de rang plein pour $n' \leq n-1$. Nous voulons le prouver pour $n' = n$.

Il est facile de vérifier que

$$\forall 2 \leq d \leq n \qquad {}^{\mathrm{T}}\mathcal{M}_{d,n} = \mathcal{M}_{n-d+2,n} \text{ et } \forall 1 \leq d \leq n \qquad {}^{\mathrm{T}}\mathcal{M}'_{d,n} = \mathcal{M}'_{n-d+1,n}$$

Pour une suite semi-régulière, on a $H(I) = \frac{n}{2}+1$. Tant que $d < H(I)$ on a $h_{d,m}(n) > 0$ et pour $d \geq H(I)$ alors $h_{d,m}(n) \leq 0$. Tant que $d \leq \frac{n-1}{2} + 1$, en utilisant la formule de récurrence on montre que $\mathcal{M}_{d,n}$ est de rang plein. Le seul cas pour lequel on ne peut pas utiliser la formule de récurrence est lorsque n est pair et que $d = \frac{n}{2} + 1$: ici $\mathcal{M}_{d-1,n-1}$ a plus de colonnes que de lignes et $\mathcal{M}_{d,n-1}$ a plus de lignes que de colonnes. Cependant, dans ce cas $\mathcal{M}_{d,n}$ est une matrice carrée de la forme :

$$\mathcal{M}_{d,n} = \begin{pmatrix} 0 & A \\ {}^{\mathrm{T}}A & C \end{pmatrix}$$

avec C une matrice symétrique inversible, et A une matrice de rang plein ayant plus de colonnes que de lignes. Un résultat algébrique simple donne que dans ce cas, $\mathcal{M}_{d,n}$ est aussi inversible (il suffit d'écrire $C = D\,{}^{\mathrm{T}}D$ avec D une matrice inversible, alors $\mathcal{M}_{d,n}$ est équivalente à la matrice $\begin{pmatrix} -A_1\,{}^{\mathrm{T}}A_1 & 0 \\ 0 & I \end{pmatrix}$ où $A_1 = A\,{}^{\mathrm{T}}D^{-1}$ est de rang plein, de sorte que $A_1\,{}^{\mathrm{T}}A_1$ est inversible.

Dans le cas d'un unique polynôme quadratique sur $\mathbb{F}_2$, la preuve est similaire : nous montrons que les matrices $\mathcal{M}'_{d,n}$ sont de rang $\binom{n-1}{d-1}$ et que la série génératrice associée au rang de ces matrices $\mathcal{M}_{d,n}$ est $z^2\frac{(1+z)^n}{1+z^2}$. La preuve utilise des propriétés algébriques des matrices de la même manière. □

Cette proposition se généralise au cas de $n+1$ polynômes de degrés quelconques :

Proposition 3.2.14 *La suite*

$$\left\{ x_1^{d_1}, \ldots, x_n^{d_n}, \sum_{m \text{ monôme de degré } d_{n+1}} m \right\}$$

de $n+1$ équations de degrés $d_1, \ldots, d_{n+1}$ en n variables est semi-régulière sur $\mathbb{K} = \mathbb{Q}$.

Une des particularités de l'indice de régularité $H(I)$ qui fait que la preuve précédente marche dans ce cas est que $H(I)$ peut être calculé explicitement, et a une forme simple en fonction de n et des degrés $d_1, \ldots, d_m$ des polynômes. Dans tous les cas autres que $m \leq n+1$, $H(I)$ ne possède pas de forme explicite simple. Cela nous incite à penser que le problème de trouver un exemple explicite de suites semi-régulière pour $m \geq n+2$ en utilisant l'approche de l'algorithme F5 est un problème difficile.

La suite que nous exhibons a l'avantage de rester semi-régulière sur $\mathbb{F}_2$, cependant sur $\mathbb{F}_2$ le nombre total de suites de degrés $d_1, \ldots, d_m$ à coefficients dans $\mathbb{F}_2$ est fini, donc la définition 3.1.1 de généricité n'a pas de sens, et il ne suffit plus de donner un exemple pour montrer que "presque toute" suite est semi-régulière. Nous conjecturons que la proportion de suites semi-régulières sur $\mathbb{F}_2$ tend vers 1 lorsque le nombre de variables tend vers l'infini.

Nous avons fait de nombreuses expériences numériques avec des suites aléatoires, et nous avons toujours obtenu des suites semi-régulières (dans le cas général, et sur $\mathbb{F}_2$ lorsque n est suffisamment grand).

3.3 Séries génératrices

Dans cette section, nous calculons explicitement la formule de récurrence vérifiée par la fonction de Hilbert d'un système de polynômes homogènes tant qu'il n'y a pas de réductions à zéro. Nous calculons la série génératrice associée, et la relions à la série de Hilbert de suites semi-régulières. Cette série génératrice sera également très utile pour l'analyse asymptotique du chapitre suivant. Nous montrons comment calculer $H(I)$ à partir de ces quantités pour des suites semi-régulières.

Soit $f_1, \ldots, f_m$ une suite de polynômes de S_n de degrés $\underline{d}_m = (d_1, \ldots, d_m)$. Notons $I = \langle f_1, \ldots, f_m \rangle$ et $HF_{d,m,\underline{d}_m}(n)$ la fonction de Hilbert de I en degré d. Nous allons calculer exactement le nombre de lignes et de colonnes de la matrice $\mathcal{M}_{d,m}$ construite par l'algorithme F5-matriciel en degré d pour une suite semi-régulière, et en déduire la fonction de Hilbert associée.

3.3.1 Série de Hilbert d'une suite semi-régulière

Rappelons que le nombre de colonnes de la matrice $\mathcal{M}_{d,m}$ est simplement le nombre de monômes en degré d :

Lemme 3.3.1 *Le nombre $M_d(n)$ de monômes en $x_1, \ldots, x_n$ de degré d dans S_n est donné par sa série génératrice :*

$$\sum_{d\geq 0} M_d(n) z^d = \frac{1}{(1-z)^n} = \sum_{d\geq 0} \binom{n+d-1}{d} z^d$$

Tant que les lignes de $\mathcal{M}_{d,m}$ sont indépendantes, la fonction de Hilbert de I est donnée par le nombre de colonnes moins le nombre de lignes de $\mathcal{M}_{d,m}$.

Lemme 3.3.2 *Tant qu'il n'y a pas de réduction à zéro au cours de l'algorithme F5-matriciel, alors la fonction de Hilbert $HF_{d,m,\underline{d}_m}(n)$ vérifie la formule de récurrence suivante :*

$$HF_{d,m,\underline{d}_m}(n) = HF_{d,m-1,\underline{d}_{m-1}}(n) - HF_{d-d_m,m-1,\underline{d}_{m-1}}(n)$$

avec comme conditions initiales $HF_{d,m,\underline{d}_m}(n) = M_d(n)$ si $m = 0$ ou $d < \min_{k\leq m}\{d_k\}$.

Démonstration Comme toutes les matrices sont de rang plein, pour construire la matrice $\mathcal{M}_{d,m}$ à partir de la matrice $\mathcal{M}_{d,m-1}$, il suffit d'ajouter toutes les lignes d'étiquette (m_l, f_m) et de supprimer autant de lignes qu'il y en a dans $\mathcal{M}_{d-d_m,m-1}$ (critère général). Si l'on note $U_{d,m}(n)$ le nombre de lignes de $\mathcal{M}_{d,m}$, on obtient :

$$\underbrace{U_{d,m}(n) - U_{d,m-1}(n)}_{\#\text{ lignes de la forme } m_l f_m \text{ en degré } d} = \underbrace{M_{d-d_m}(n)}_{\#\text{ monômes de degré } d-d_m} - \underbrace{U_{d-d_m,m-1}(n)}_{\text{Critère général}}$$

avec comme conditions initiales $U_{d,m}(n) = 0$ si $m = 0$ ou $d < \min_{k\leq m}\{d_k\}$. Comme $HF_{d,m,\underline{d}_m}(n) = M_d(n) - U_{d,m}(n)$ alors $HF_{d,m,\underline{d}_m}(n) - HF_{d,m-1,\underline{d}_{m-1}}(n) = U_{d,m-1}(n) - U_{d,m}(n)$ ce qui termine la preuve. □

La série génératrice associée à la formule de récurrence du lemme 3.3.2 a une forme très simple, et tant qu'il n'y a pas de réduction à zéro alors $HF_{d,m,\underline{d}_m}(n)$= # colonnes - # lignes est donné par le coefficient de degré d de cette série, et est donc très facile à calculer.

Proposition 3.3.3 *Soit $h_{d,m}(n)$ une suite vérifiant la formule de récurrence du lemme 3.3.2, i.e.*

$$h_{d,m}(n) = h_{d,m-1}(n) - h_{d-d_m,m-1}(n)$$

avec pour conditions initiales $h_{d,m}(n) = M_d(n)$ si $m = 0$ ou $d < \min_{k\leq m}\{d_k\}$. Alors la série génératrice de $h_{d,m}(n)$ est

$$S_{m,n}(z) = \sum_{d\geqslant 0} h_{d,m}(n) z^d = \prod_{k=1}^{m} (1 - z^{d_k}) \Big/ (1-z)^n \tag{3.3}$$

Démonstration En utilisant la formule de récurrence on obtient :

$$\begin{aligned}\sum_{d\geqslant 0} h_{d,m}(n) z^d &= \sum_{d\geqslant 0} h_{d,m-1}(n) z^d - z^{d_m} \sum_{d\geqslant 0} h_{d,m-1}(n) z^d \\ &= (1 - z^{d_m}) \sum_{d\geqslant 0} h_{d,m-1}(n) z^d = \prod_{k=2}^{m} (1 - z^{d_k}) \sum_{d\geqslant 0} h_{d,1}(n) z^d\end{aligned}$$

On a $h_{d,1}(n) = M_d(n) - M_{d-d_1}(n)$ ce qui donne

$$\sum_{d\geqslant 0} h_{d,1}(n)z^d = (1-z^{d_1})\sum_{d\geqslant 0} M_d(n)z^d = \frac{1-z^{d_1}}{(1-z)^n}.$$

. □

Corollaire 3.3.4 *La série de Hilbert d'une suite semi-régulière de m polynômes homogènes de degrés respectifs $d_1, \ldots, d_m$ est*

$$\Big[\prod_{i=1}^{m}(1-z^{d_i})/(1-z)^n\Big] \tag{3.1}$$

Réciproquement, toute suite homogène de m polynômes de degrés $d_1, \ldots, d_m$ ayant pour série de Hilbert (3.1) est une suite semi-régulière.

Démonstration Tant que $h_{d,m}(n) > 0$, la fonction de Hilbert vérifie la formule de récurrence du lemme 3.3.2, et les coefficients de la série de Hilbert et de la série (3.1) sont égaux. Considérons le premier d pour lequel le coefficient de la série génératrice est négatif. Comme la suite est semi-régulière, on a $d = H(I)$ et tous les monômes sont atteints (sinon il y aurait une réduction à zéro avant le degré $H(I)$). Ainsi la fonction de Hilbert vaut zéro pour $d' \geq d$, ce qui est aussi le cas pour les coefficients de la série (3.1).

Réciproquement, supposons que la suite $f_1, \ldots, f_m$ a pour série de Hilbert la série (3.1). Le $d^{\text{ème}}$ coefficient $HF_{d,m,\underline{d}_m}(n)$ de la série de Hilbert correspond au nombre de colonnes de la matrice $\mathcal{M}_{d,m}$ moins son rang. Si l'on note $\sum_{d\geq 0} a_d z^d$ la série (3.3), alors a_d correspond au nombre de colonnes de $\mathcal{M}_{d,m}$ moins le nombre de lignes, et tant que $a_d > 0$ on a $HF_{d,m,\underline{d}_m}(n) = a_d$ de sorte que la matrice $\mathcal{M}_{d,m}$ est de rang plein, il n'y a pas de réduction à zéro. Si $a_d \leq 0$, alors $HF_{d,m,\underline{d}_m}(n) = 0$ de sorte que tous les monômes de degré d sont atteints et la suite est semi-régulière. □

3.3.2 Série de Hilbert d'une suite semi-régulière sur $\mathbb{F}_2$

Sur $\mathbb{F}_2$, le critère général et le critère de Frobenius de l'algorithme F5-matriciel sont indépendants, ce qui permet comme dans le cas général de compter exactement le rang de chaque matrice $\mathcal{M}_{d,m}$, et de déterminer la série de Hilbert et l'indice de régularité de suites semi-régulières homogènes sur $\mathbb{F}_2$.

Lemme 3.3.5 *Le nombre $M_d(n)$ de monômes en $x_1, \ldots, x_n$ de degré d dans R_n^h est donné par la série génératrice :*

$$\sum_{d\geq 0} M_d(n)z^d = (1+z)^n = \sum_{d\geq 0}\binom{n}{d}z^d$$

Lemme 3.3.6 *Étant donnée une suite $f_1, \ldots, f_m$, tant qu'il n'y a pas de réduction à zéro, la fonction de Hilbert $HF_{d,m,\underline{d}_m}(n)$ vérifie la formule de récurrence :*

$$HF_{d,m,\underline{d}_m}(n) = HF_{d,m-1,\underline{d}_{m-1}}(n) - HF_{d-d_m,m,\underline{d}_m}(n)$$

avec pour conditions initiales $HF_{d,m,\underline{d}_m}(n) = M_d(n)$ si $m = 0$ ou $d < \min_{k\leq m}\{d_k\}$.

Démonstration Comme toutes les matrices sont de rang plein, pour construire la matrice $\mathcal{M}_{d,m}$ à partir de la matrice $\mathcal{M}_{d,m-1}$, il suffit d'ajouter toutes les lignes d'étiquette (m_l, f_m) et de retirer un nombre de lignes égal au nombre de lignes de $\mathcal{M}_{d-d_m,m}$, ce qui correspond à l'application des critères généraux et de Frobenius. Si l'on note $U_{d,m}(n)$ le nombre de lignes de $\mathcal{M}_{d,m}$, alors on a :

$$\underbrace{U_{d,m}(n) - U_{d,m-1}(n)}_{\#\text{ lignes de la forme } m_l f_m \text{ en degré } d} = \underbrace{M_{d-d_m}(n)}_{\#\text{ monômes de degré } d-d_m} - \underbrace{U_{d-d_m,m}(n)}_{\text{(Critère général + Frobenius)}}$$

avec pour conditions initiales $U_{d,m}(n) = 0$ si $m = 0$ ou $d < \min_{k\leq m}\{d_k\}$. Comme $HF_{d,m,\underline{d}_m}(n) = M_d(n) - U_{d,m}(n)$ alors $HF_{d,m,\underline{d}_m}(n) - HF_{d,m-1,\underline{d}_{m-1}}(n) = U_{d,m-1}(n) - U_{d,m}(n)$. □

La série génératrice associée à la formule de récurrence du lemme 3.3.6 est simple à calculer :

Proposition 3.3.7 *Soit $h_{d,m}(n)$ une suite vérifiant la formule de récurrence du lemme 3.3.6, soit*

$$h_{d,m}(n) = h_{d,m-1}(n) - h_{d-d_m,m}(n)$$

avec pour conditions initiales $h_{d,m}(n) = M_d(n)$ si $m = 0$ ou $d < \min_{k\leq m}\{d_k\}$. Alors la série génératrice de $h_{d,m}(n)$ est

$$S_{m,n}(z) = \sum_{d\geqslant 0} h_{d,m}(n) z^d = (1+z)^n \Big/ \prod_{k=1}^{m} \left(1 + z^{d_k}\right) \tag{3.4}$$

Démonstration A partir de la formule de récurrence on obtient :

$$\sum_{d\geqslant 0} h_{d,m}(n) z^d = \sum_{d\geqslant 0} h_{d,m-1}(n) z^d - z^{d_m} \sum_{d\geqslant 0} h_{d,m}(n) z^d$$

et ainsi

$$\sum_{d\geqslant 0} h_{d,m}(n) z^d = \frac{1}{1+z^{d_m}} \sum_{d\geqslant 0} h_{d,m-1}(n) z^d = \frac{1}{\prod_{k=2}^{m}\left(1+z^{d_k}\right)} \sum_{d\geqslant 0} h_{d,1}(n) z^d$$

Nous avons $h_{d,1}(n) = M_d(n) - h_{d-d_1,1}(n)$ ce qui donne

$$\sum_{d\geqslant 0} h_{d,1}(n) z^d = \frac{1}{1+z^{d_1}} \sum_{d\geqslant 0} M_d(n) z^d = \frac{(1+z)^n}{1+z^{d_1}}.$$

□

Corollaire 3.3.8 *La série de Hilbert d'une suite semi-régulière de m polynômes homogènes de degrés respectifs $d_1, \ldots, d_m$ sur $\mathbb{F}_2$ est*

$$\left[(1+z)^n / \prod_{i=1}^{m} (1+z^{d_i})\right] \tag{3.2}$$

Réciproquement, toute suite de m polynômes homogènes de degrés $d_1, \ldots, d_m$ ayant pour série de Hilbert (3.2) est une suite semi-régulière.

3.3.3 Calcul explicite de $H(I)$ pour les suites semi-régulières

Pour un système de dimension zéro, le lemme suivant permet de calculer l'indice de régularité $H(I)$ à partir de la série génératrice (sur un corps quelconque) :

Lemme 3.3.9 *Soit $f_1, \ldots, f_m$ une suite semi-régulière ayant un nombre fini de solutions, notons $S_{m,n}(z) = \sum_{d \geqslant 0} h_{d,m}(n) z^d$ sa série génératrice. Alors $\forall d \leq H(I)$, $h_{d,m}(n) = HF_{d,m,\underline{d}_m}(n)$ et $H(I)$ est caractérisé par*

$$(\forall d < H(I), \quad h_{d,m}(n) > 0) \quad \text{et} \quad h_{H(I),m}(n) \leq 0$$

Démonstration Comme la suite est semi-régulière, pour $d < H(I)$ il n'y a aucune réduction à zéro, de sorte qu'il y a plus de colonnes que de lignes et $HF_{d,m,\underline{d}_m}(n) = h_{d,m}(n) > 0$ (il ne peut pas y avoir égalité à zéro par définition de $H(I)$). Pour $d = H(I)$, tous les monômes de degré $H(I)$ sont atteints, le rang de la matrice est exactement égal au nombre de colonnes, et $h_{H(I),m}(n) = \#\text{ colonnes } - \#\text{ lignes } \leq \#\text{ colonnes } - \text{ rang } = 0$. □

Il est donc très facile de calculer $H(I)$ à partir des séries génératrices : il suffit d'évaluer les premiers coefficients de la série, jusqu'à trouver le premier négatif.

Considérons par exemple le cas de $n+1$ équations quadratiques. La série génératrice est

$$\begin{aligned} S_{n+1,n}(z) &= (1+z)^n(1-z^2) \\ &= 1 + nz + \frac{1}{2}(n+1)(n-2)z^2 + \frac{1}{6}n(n+1)(n-4)z^3 \\ &\quad + \frac{1}{24}(n-1)n(n+1)(n-6)z^4 \\ &\quad + \frac{1}{120}(n-2)(n-1)n(n+1)(n-8)z^5 + O(z^6) \end{aligned}$$

Le coefficient en z est n, qui est toujours positif, donc on a $H(I) \geq 1$. Le coefficient en z^2 est ≤ 0 pour $n \leq 2$, et > 0 pour $n > 2$, ce qui donne $H(I) = 2$ pour $n \leq 2$ et $H(I) \geq 3$ pour $n > 2$. Le coefficient en z^3 a une racine positive 4, donc il est négatif pour $2 < n \leq 4$ et $H(I) = 3$, et positif pour $n \geq 5$, et $H(I) \geq 4$. À nouveau, la plus grande racine du coefficient en z^4 est 6, donc pour $5 \leq n \leq 6$ on

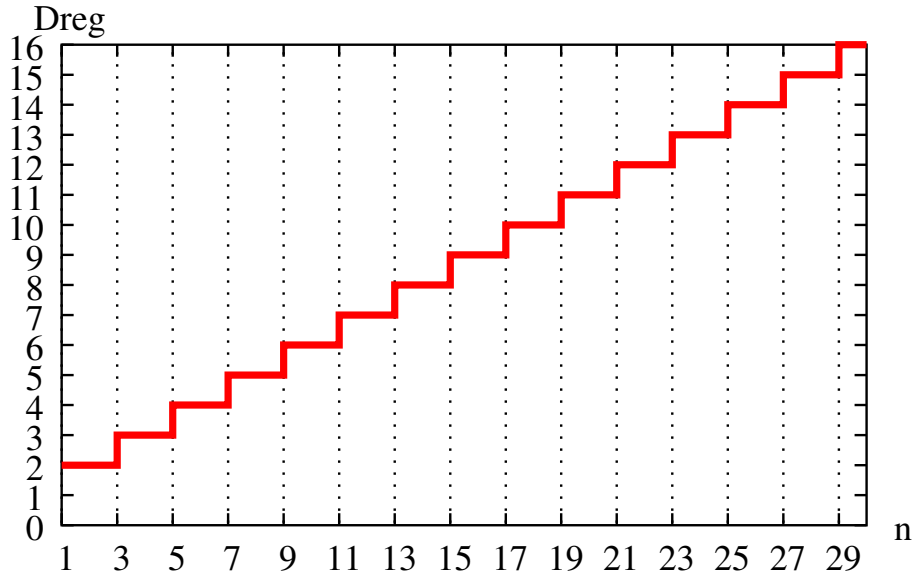


FIG. 3.1 – $H(I)$ pour $m = n + 1$ polynômes quadratiques

a $H(I) = 4$ et pour $n \geq 7$ on a $H(I) \geq 5$. Il suffit donc de calculer successivement la plus grande racine du coefficient de z^d dans $S_{n,n}(z)$ pour tracer l'escalier de la figure 3.1

La question que l'on se pose alors est la suivante : quelle est la pente moyenne de cet escalier ? la réponse est $1/2$, et dans ce cas particulier on peut le calculer explicitement à partir de la série $S_{n,n}(z)$: on a $[z^d]S_{n,n}(z) = \binom{n+2}{d}\frac{n+2-2d}{n+2}$, sa plus grande racine en tant que polynôme en n est $2(d-1)$. Dans le cas général, l'expression du coefficient de degré d de $S_{n,m}(z)$ en fonction de n est trop complexe pour que l'on puisse calculer explicitement sa plus grande racine. Nous donnons dans le chapitre 4 une estimation asymptotique de cette pente, en calculant un développement asymptotique de $H(I)$ lorsque $n \to \infty$.

3.4 Étude fine de la complexité de F5 pour des suites régulières

Dans cette section, nous étudions plus précisément le comportement de l'algorithme F5 sur des suites régulière en position de Noether, pour l'ordre grevlex. Considérons une suite $f_1, \ldots, f_m \subset S_n$ avec $m \leq n$. Fixons comme ordre monomial l'ordre grevlex avec $x_1 > \ldots > x_n$ et notons $I = \langle f_1, \ldots, f_m \rangle$ l'idéal engendré par la suite de polynômes. Rappelons la notation $S_i = \mathbb{K}[x_1, \ldots, x_i]$.

Dans toute cette section nous nous placerons sous les hypothèses suivantes :

Hypothèse 1. La suite $f_1, \ldots, f_m$ est régulière.

Hypothèse 2. Pour tout $1 \leq i \leq m$, les variables $x_1, \ldots, x_i$ mettent la sous-suite

$f_1, \ldots, f_i$ en position de Noether.

La définition classique d'une suite en position de Noether est rappelée par exemple dans [Eis95]. Nous utiliserons plutôt ici la caractérisation donnée page 27 (définition 1.8.3) dans le cas de l'ordre grevlex. L'hypothèse 2 peut alors s'écrire :
Hypothèse 2 (version équivalente). Pour tout $1 \leq i \leq m$ il existe $n_i > 0$ tel que $x_i^{n_i} \in \mathrm{LT}(\langle f_1, \ldots, f_i \rangle)$ et il n'existe pas $t_i > 0$ tel que $x_i^{t_i} \in \mathrm{LT}(\langle f_1, \ldots, f_{i-1} \rangle)$.

Nous conjecturons que l'hypothèse 2 équivaut à supposer que la suite $f_1, \ldots, f_m$ est en position de Noether, à permutation près des polynômes.

Nous allons montrer le théorème suivant :

Théorème 3.4.1 *Sous les hypothèses 1 et 2, le nombre de polynômes de degré d de la base de Gröbner de $\langle f_1, \ldots, f_i \rangle$ est majoré par $g_{d,i}(n)$, qui est donné par la série génératrice suivante :*

$$\sum_{d=0}^{\infty} g_{d,i}(n) z^d = \frac{z^{d_i}}{(1-z)^{i-1}} \prod_{k=1}^{i-1} (1 - z^{d_k}) \tag{3.5}$$

Remarquons que cette série est en fait un polynôme, de degré $\delta = \sum_{j=1}^{i} (d_j - 1) + 1$. On en déduit donc que la base de Gröbner ne contient que des polynômes de degré $\leq \delta$, ce qui permet de retrouver la borne de Macaulay [Laz83].

Ce théorème sera prouvé Section 3.4.1 en suivant très précisément le comportement de l'algorithme F5-matriciel. Nous en déduirons Section 3.4.2 une borne très précise pour le nombre total d'opérations élémentaires effectuées par l'algorithme F5-matriciel :

Théorème 3.4.2 *Sous les hypothèses 1 et 2, le nombre total d'opérations élémentaires effectuées par l'algorithme F5-matriciel pour calculer la base de Gröbner de $\langle f_1, \ldots, f_m \rangle$ est majoré par N_{op}, avec*

$$N_{op} = \sum_{i=1}^{m-1} \sum_{d=0}^{\infty} g_{d+d_{i+1},i+1}(n) \binom{i+d+d_{i+1}}{d+d_{i+1}} \binom{n+d+d_{i+1}-1}{d+d_{i+1}}$$

En supposant que tous les polynômes sont quadratiques, la formule se simplifie en :

$$N_{op} = \sum_{d=0}^{m-1} \binom{2d+2}{d} \binom{n+d+1}{d+2} \binom{m+d+2}{2d+3} - \binom{n+1}{2}.$$

De telles formules pour l'algorithme de Buchberger n'existent que dans le cas de polynômes en 2 variables [Buc83] ou 3 variables [Win84]. En analysant asymptotiquement la valeur de N_{op}, nous pouvons comparer le gain de l'algorithme F5-matriciel (qui construit incrémentalement des matrices de type Macaulay, et calcule au fur et à mesure leur forme Echelon) par rapport, par exemple, au calcul direct d'une forme Echelon sur la matrice de Macaulay en degré suffisamment élevé :

Théorème 3.4.3 *Sous les hypothèses 1 et 2, le nombre total d'opérations élémentaires effectuées par l'algorithme F5-matriciel pour calculer la base de Gröbner d'un système $f_1, \ldots, f_n$ de n équations quadratiques en n variables est majoré par :*

$$N_{op} = 2^{d_1 n + o(n)} = 2^{4.3n + o(n)}$$

$$\begin{aligned} \text{avec } d_1 &= -3\lambda_0 \log_2(\lambda_0) + 2(\lambda_0+1)\log_2(\lambda_0+1) - (1-\lambda_0)\log_2(1-\lambda_0) \\ \text{et } \lambda_0 &= \frac{1}{6}\left((44+3\sqrt{177})^{1/3} + (44-3\sqrt{177})^{1/3} - 1\right) \approx 0.829 \end{aligned}$$

Comparativement, le coût de la mise sous forme Échelon de la matrice de Macaulay en degré $n+1$ est dominé par

$$N_{op}^{Macaulay} = 2^{6n+o(n)}$$

Il n'existe pas à notre connaissance d'algorithme rapide permettant de calculer une forme Echelon pour des matrices creuses. Un tel algorithme, s'il existait, aurait au mieux une complexité de $2^{4n+o(n)}$. La borne de complexité *prouvée* de l'algorithme F5-matriciel en $2^{4.3n+o(n)}$ est donc vraiment compétitive.

Cette analyse n'est donnée ici que dans le cas de polynômes quadratiques comportant autant d'équations que d'inconnues, il sera intéressant de la poursuivre dans d'autres cas (équations de degré $D > 2$, systèmes polynomiaux creux, comparaison avec la méthode des sous-résultants creux, etc.).

Les deux sections suivantes contiennent les preuves des théorèmes précédents.

3.4.1 L'algorithme F5-matriciel pas à pas

Le but de cette section est de prouver le théorème 3.4.1. Nous allons voir que, pour des suites vérifiant les hypothèses 1 et 2, il est possible de suivre pas à pas le déroulement de l'algorithme F5-matriciel.

Nous commençons la preuve par trois lemmes :

Lemme 3.4.4 *Sous les hypothèses 1 et 2, pour tout $1 \leq i \leq m$, la suite $f_1, \ldots, f_i$, $x_{i+1}, \ldots, x_n$ est régulière.*

Démonstration On a $\langle f_1, \ldots, f_i \rangle \cap \mathbb{K}[x_{i+1}, \ldots, x_n] = \{0\}$ par définition de la dimension et le fait que la suite soit en position de Noether. Donc si $f_1, \ldots, f_i, x_{i+1}, \ldots, x_n$ n'est pas une suite régulière, il existe $g_1, \ldots, g_i$ tels que $\sum_{j=1}^{i} f_i g_i$ soit un élément non nul de $\mathbb{K}[x_{i+1}, \ldots, x_n]$ ce qui contredit la première équation. □

De l'hypothèse 2, nous pouvons déduire en particulier que $\mathrm{LT}(f_1) = x_1^{d_1}$. Notons G_i la base de Gröbner de $f_1, \ldots, f_i$. Plus généralement, ces deux hypothèses impliquent le lemme suivant :

Lemme 3.4.5 *Sous les hypothèses 1 et 2, tout polynôme f d'étiquette (m_i, f_i) avec $m_i \in S_i$ apparaissant dans l'algorithme F5 au cours du calcul de G_i vérifie $LT(f) \in S_i$.*

Démonstration Notons H_i la base de Gröbner de $x_{i+1}, \ldots, x_n, f_1, \ldots, f_i$. Si l'on construit en parallèle les matrices de l'algorithme F5 pour G_i et H_i, alors les lignes de H_i concernant le polynôme f_i sont exactement les lignes de G_i concernant f_i d'étiquette dans S_i. Si un polynôme f apparaissait dans le calcul de G_i avec une étiquette (m_i, f_i), $m_i \in S_i$ et $x_k|\mathrm{LT}(f)$ pour un $k > i$ alors on aurait une réduction à zéro dans la matrice correspondant au calcul de H_i, ce qui est absurde car d'après le lemme 3.4.4 la suite $x_{i+1}, \ldots, x_n, f_1, \ldots, f_i$ est régulière. □

Toujours sous les hypothèses 1 et 2, on peut démontrer le lemme suivant, qui précise les polynômes appartenant à la base de Gröbner de $f_1, \ldots, f_i$:

Lemma 3.4.6 *Sous les hypothèses 1 et 2, tout polynôme g d'étiquette (m_i, f_i) qui est réduit lors du calcul de G_i vérifie $m_i \in S_{i-1}$.*

Démonstration Nous prouvons ce lemme par récurrence sur i le nombre de polynômes et d le degré. Le lemme est vrai pour $i = 1$ car il n'y a aucune réduction. Soit $i > 1$ un entier, alors le lemme est vrai pour $d = d_i$ le degré de f_i car l'étiquette de f_i est $(1, f_i)$ (le lemme est aussi vrai pour $d < d_i$ car il n'y a aucun polynôme de ce degré d'étiquette (m_i, f_i)). Soit maintenant f un polynôme intervenant dans le calcul de G_i en degré $d > d_i$, supposons le lemme vrai pour tous les polynômes d'étiquette plus petite que celle de f. Tout nouveau polynôme f de degré d s'écrit, avant réduction, $f = \lambda f_0$, avec degré$(\lambda) \geqslant 1$, f_0 d'étiquette (μ, f_i) réduit par rapport à la base G_{i-1} et tous les νf_0 réduits par rapport à G_{i-1} pour $\nu|\lambda$ strictement. On a par hypothèse de récurrence $\mu \in S_{i-1}$, $\mathrm{LT}(f_0) \in S_i$ et f est d'étiquette $(\lambda\mu, f_i)$.

Supposons que f est réductible, soit g un élément de G_k, avec $k \leq i$, d'étiquette (β, f_k), qui réduise f. Par hypothèse de récurrence on a $\beta \in S_{k-1}$ et d'après le lemme 3.4.5 on a $\mathrm{LT}(g) \in S_k$. Comme g ne réduit aucun νf_0 pour $\nu|\lambda$, alors $\mathrm{LT}(g)$ divise $\mathrm{LT}(f) = \lambda\mathrm{LT}(f_0)$ mais $\mathrm{LT}(g)$ ne divise aucun $\nu\mathrm{LT}(f_0)$ pour ν diviseur strict de λ, donc $\lambda|\mathrm{LT}(g)$ et $\lambda \in S_k$.

1. si $k < i$, on a $\lambda \in S_{i-1}$ et l'étiquette de f est $(\lambda\mu, f_i)$ avec $\lambda\mu \in S_{i-1}$.
2. si $k = i$, soit x_l la variable principale de λ (i.e. le plus grand l tel que x_l apparaît dans λ).
   - si $l \leq i - 1$ alors $\lambda \in S_{i-1}$ et on a bien $\lambda\mu \in S_{i-1}$.
   - si $l = i$, alors on peut écrire $\lambda\mathrm{LT}(f_0) = c\mathrm{LT}(g)$ avec $c \in S_{i-1}$ (si $x_i|c$ alors $\mathrm{LT}(g)|\frac{\lambda}{x_i}\mathrm{LT}(f_0)$ absurde). Alors, l'étiquette de cg est $(c\beta, f_i)$ qui est supérieure à l'étiquette de f car $c\beta \in S_{i-1}$ et $\lambda\mu$ contient du x_i, ce qui est absurde car cg réduit f.

□

Nous allons déterminer précisément le nombre de lignes qui vont être réduites au cours de l'algorithme F5, ce qui donnera exactement le nombre de polynômes dans la base de Gröbner (non réduite) donnée en sortie de l'algorithme F5. Nous utilisons les notations suivantes :

Notations. Nous noterons :

- $u_{d,i}(n)$ le nombre de lignes en degré d d'étiquette (t, f_i),
- $s_{d,i,k}(n)$ le nombre de lignes en degré d d'étiquette (t, f_i) avec $t \in \mathbb{K}[x_1, \ldots, x_k]$,
- $ht_{d,i,k}(n)$ le nombre de lignes en degré d d'étiquette (t, f_i) dont le terme de tête après réduction est dans $\mathbb{K}[x_1, \ldots, x_k]$,
- $g_{d,i}(n)$ le nombre de lignes en degré d d'étiquette (t, f_i) qui peuvent se réduire lors du calcul de G_i.

Notons également $U_{d,i}(n) = \sum_{j=1}^{i} U_{d,j}(n)$, $S_{d,i,k}(n) = \sum_{j=1}^{i} s_{d,j,k}(n)$ et $HT_{d,i,k}(n) = \sum_{j=1}^{i} ht_{d,j,k}(n)$.

Lemme 3.4.7 *Si $f_1, \ldots, f_m$ sont de degrés $d_1, \ldots, d_m$, alors*

$$s_{d,i,k}(n) = M_{d-d_i}(k) - HT_{d-d_i,i-1,k}(n) \tag{3.6}$$

Démonstration Le nombre de lignes possibles ayant pour étiquette (t, f_i) avec $t \in \mathbb{K}[x_1, \ldots, x_k]$ est le nombre de monômes de degré $d - d_i$ en k variables, moins les lignes rejetées par les critères, et il y en a autant que de lignes en degré $d - d_i$ et en les $i - 1$ premiers polynômes dont le terme de tête est dans $\mathbb{K}[x_1, \ldots, x_k]$. □

Le lemme suivant est un corollaire du lemme 3.4.5 :

Lemme 3.4.8 *Pour $i \leqslant k \leqslant n$ on a $u_{d,i}(k) = s_{d,i,k}(n) = ht_{d,i,k}(n)$ et $g_{d,i}(n) = s_{d,i,i-1}(n)$.*

Démonstration La première égalité provient du lemme 3.4.4, et la seconde du lemme 3.4.5. La dernière découle du lemme 3.4.6. □

Lemme 3.4.9 *Le nombre de polynômes dans la base de Gröbner G_i vérifie la formule de récurrence :*

$$g_{d,i}(n) = M_{d-d_i}(i-1) - U_{d-d_i,i-1}(i-1) \tag{3.7}$$

Proposition 3.4.10 *La série génératrice de $g_{d,i}(n)$ est :*

$$\sum_{d=0}^{\infty} g_{d,i}(n) z^d = \frac{z^{d_i}}{(1-z)^{i-1}} \prod_{k=1}^{i-1} (1 - z^{d_k}) \tag{3.8}$$

Démonstration Nous avons vu que les séries génératrices de $U_{d-d_i,i-1}(i-1)$ et $M_{d-d_i}(i-1)$ sont :

$$\begin{aligned} \sum_{d=0}^{\infty} U_{d-d_i,i-1}(i-1) z^d &= z^{d_i} \left(\frac{1}{(1-z)^{i-1}} - \frac{\prod_{k=1}^{i-1}(1-z^{d_k})}{(1-z)^{i-1}} \right) \\ \sum_{d=0}^{\infty} M_{d-d_i}(i-1) z^d &= \frac{z^{d_i}}{(1-z)^{i-1}} \end{aligned}$$

□

3.4.2 Nombre d'opérations élémentaires

Connaissant exactement les lignes de la matrices qui vont être réduites, il est possible de calculer le nombre d'opérations élémentaires effectuées par l'algorithme F5 pour réduire la matrice. Nous utiliserons ici, pour réduire la matrice en degré d, un algorithme de Gauss sans pivot de lignes ou de colonnes, et pour lequel une ligne ne peut être réduite que par les lignes qui la précèdent dans la matrice.

Pour chaque polynôme f_i considéré, pour tous les degrés d de 0 au degré maximal (ici $i-1$) on réduit $g_{d,i}(n)$ lignes. Pour une ligne correspondant à un polynôme g, d'après la section précédente son terme de tête après réduction est dans $\mathbb{K}[x_1,\ldots,x_i]$, donc on réduit par au plus le nombre de lignes correspondant à des monômes de $\mathbb{K}[x_1,\ldots,x_i]$, il y en a $\binom{i+d-1}{d}$. Pour chaque ligne f réduisant g, on effectue la combinaison linéaire de ces deux lignes $L_g \leftarrow L_g - cL_f$, il faut donc multiplier chaque terme de f par c, et il y a au plus $\binom{n+d-1}{d}$ termes dans f. On obtient donc la formule suivante :

$$\begin{aligned}
N_{\text{op}} &= \sum_{i=2}^{m} \sum_{d=d_i}^{\sum_{j=1}^{i}(d_j-1)+1} g_{d,i}(n)\binom{i+d-1}{d}\binom{n+d-1}{d} \\
&= \sum_{i=1}^{m-1} \sum_{d=0}^{\sum_{j=1}^{i}(d_j-1)} g_{d+d_{i+1},i+1}(n)\binom{i+d+d_{i+1}}{d+d_{i+1}}\binom{n+d+d_{i+1}-1}{d+d_{i+1}} \\
&= \sum_{i=1}^{m-1} \sum_{d=0}^{\infty} g_{d+d_{i+1},i+1}(n)\binom{i+d+d_{i+1}}{d+d_{i+1}}\binom{n+d+d_{i+1}-1}{d+d_{i+1}}
\end{aligned}$$

Lorsque tous les degrés $d_i = 2$, on a $g_{d,i}(n) = \binom{i-1}{d-2}$ et on obtient :

$$N_{\text{op}} = \sum_{i=1}^{m-1} \sum_{d=0}^{i} \binom{i}{d}\binom{i+d+2}{d+2}\binom{n+d+1}{d+2}$$

En intervertissant les deux signes somme, on obtient

$$\begin{aligned}
N_{\text{op}} &= \sum_{d=0}^{m-1} \sum_{i=\max(d,1)}^{m-1} \binom{i}{d}\binom{i+d+2}{d+2}\binom{n+d+1}{d+2} \\
&= \sum_{d=0}^{m-1} \binom{n+d+1}{d+2} \left(\sum_{i=\max(0,1-d)}^{m-1-d} \binom{i+d}{d}\binom{i+2d+2}{d+2} \right)
\end{aligned}$$

En remarquant que $\binom{i+d}{d}\binom{i+2d+2}{d+2} = \frac{(i+d)!(i+2d+2)!}{d!i!(i+d)!(d+2)!} = \frac{(i+2d+2)!}{i!(2d+2)!}\frac{(2d+2)!}{d!(d+2)!} = \binom{i+2d+2}{2d+2}\binom{2d+2}{d}$

on obtient

$$\begin{aligned} N_{\text{op}} &= \sum_{d=0}^{m-1} \binom{2d+2}{d} \binom{n+d+1}{d+2} \left(\sum_{i=\max(0,1-d)}^{m-d-1} \binom{i+2d+2}{2d+2} \right) \\ &= \sum_{d=0}^{m-1} \binom{2d+2}{d} \binom{n+d+1}{d+2} \binom{m+d+2}{2d+3} - \binom{n+1}{2} \end{aligned}$$

Pour estimer asymptotiquement N_{op}, nous estimons le plus grand terme u_d de la somme. Pour $m = n$ équations, nous obtenons la valeur suivante :

Lemme 3.4.11 *Le terme dominant de la somme précédente est obtenu pour $d \sim \lambda_0 n$ avec*

$$\lambda_0 = \frac{1}{6}\left((44 + 3\sqrt{177})^{1/3} + (44 - 3\sqrt{177})^{1/3} - 1\right) \approx 0.829$$

On a alors

$$N_{op} = 2^{d_1 n + o(n)}$$

$$\begin{aligned} \text{avec } d_1 &= -3\lambda_0 \log_2(\lambda_0) + 2(\lambda_0+1)\log_2(\lambda_0+1) - (1-\lambda_0)\log_2(1-\lambda_0) \\ &\approx 4.295 \end{aligned}$$

Comparativement, le coût de la mise sous forme Échelon de la matrice de Macaulay en degré $n+1$ est au pire

$$n\binom{2n-1}{n-1}\binom{2n+1}{n+1}^2 \sim \frac{2}{\sqrt{n}\pi^{\frac{3}{2}}}\, 2^{6n}$$

Démonstration L'étude de la fonction $f(d) = \frac{u_{d+1}}{u_d} - 1$ montre que cette fonction est décroissante en d, et qu'elle s'annule pour $d = d_0$ avec $\frac{1}{2}n < d_0 < n-2$. Posons alors $\lambda_0 = \frac{d_0}{n}$ et cherchons une valeur approchée de $\lambda_0 = O(1)$ lorsque n tend vers l'infini. Le numérateur de $f(d)$ est un polynôme de degré 4 en n, dont le terme dominant est $2\lambda(-2\,\lambda^3 - \lambda^2 + \lambda + 1)n^4$. Pour approcher de la racine de $f(d)$, il faut que ce terme dominant s'annule, et donc que λ_0 soit sa racine strictement positive.

Pour estimer u_{d_0}, on utilise la formule $\log(\binom{an+b}{cn+d}) = (a\log(a) - c\log(c) - (a-c)\log(a-c))n - \frac{1}{2}\log(n) + O(1)$ qui donne : $\log(u_{d_0}) \sim d_1 n$.

Calculons le coût de la résolution de la matrice de Macaulay : cette matrice (en degré $n+1$) possède $\binom{2n+1}{n+1}$ colonnes et $n\binom{2n-1}{n-1}$ lignes. La complexité de la mise sous forme Échelon de cette matrice est au pire

$$n\binom{2n-1}{n-1}\binom{2n+1}{n+1}^2 \sim \frac{2}{\sqrt{n}\pi^{\frac{3}{2}}}\, 2^{6n}$$

□

3.5 Suites semi-régulières affines

Nous avons vu que les suites semi-régulières homogènes sont exactement les suites pour lesquelles il n'y a aucune réduction à zéro dans l'algorithme F5-matriciel.

On définit habituellement les suites régulières affines de la même façon que dans le cas homogène. Cependant, pour définir les suites semi-régulières affines tout en conservant les propriétés des suites homogènes, nous avons besoin d'une hypothèse plus forte, qui est qu'il n'y ait aucune *chute de degré* avant le degré $H(I)$. Notons qu'une chute de degré pour un polynôme affine correspond à une réduction à zéro de sa partie homogène de plus haut degré, ce qui conduit à la définition suivante :

Définition 3.5.1 *Soit $f_1, \ldots, f_m \subset S_n$ une suite de m polynômes affines, et $I = \langle f_1, \ldots, f_m \rangle$. Notons f_i^h la partie homogène de f_i de plus haut degré, $1 \leq i \leq m$. La suite $f_1, \ldots, f_m$ est dite semi-régulière si la suite $f_1^h, \ldots, f_m^h$ l'est. L'indice de régularité de la suite $f_1^h, \ldots, f_m^h$ s'appelle le* degré de régularité *de la suite $f_1, \ldots, f_m$, c'est le degré de la première chute de degré lors d'un calcul de base de Gröbner. Nous le noterons D_{reg}.*

Notons que D_{reg} dépend des générateurs $f_1, \ldots, f_m$ de l'idéal.

Notre analyse de complexité ne s'appliquera qu'à la définition ci-dessus de suites semi-régulières. Par exemple, nous ne considérons pas que la suite $\{x + 1, xy\}$ soit régulière. Remarquons que génériquement, un système affine surdéterminé n'a aucune solution, l'idéal associé vaut $\langle 1 \rangle$, et donc l'indice de régularité de la fonction de Hilbert est 0, par contre le degré de régularité est beaucoup plus élevé.

Avec notre définition, l'ensemble des suites semi-régulières affines reste tout de même générique (c'est un ouvert non vide de Zariski, "presque toute" suite est semi-régulière). De plus, parmi les suites affines ayant une solution commune (i.e. pour lesquels l'idéal n'est pas réduit à $\{1\}$), l'ensemble des suites semi-régulières est encore un ouvert de Zariski non vide.

L'algorithme F5-matriciel n'est décrit que dans le cas de polynômes homogènes, mais les critères restent valables tant qu'il n'y a pas de chutes de degré (les colonnes de $\mathcal{M}_{d,m}$ étant alors indexées par tous les monômes de degré $\leq d$). Pour une suite affine semi-régulière, l'algorithme F5-matriciel va donc marcher pour tout $d \leq D_{\text{reg}}$: les parties homogènes de plus haut degré des polynômes de la suite formant une suite semi-régulière, il n'y a aucune réduction à zéro, ce qui équivaut à aucune chute de degré pour les polynômes affines. L'algorithme F5-matriciel ne renvoie plus une base de Gröbner jusqu'au degré d, mais D_{reg} est toujours une borne sur le degré des éléments de la base de Gröbner. On peut exécuter F5-matriciel avec le degré D_{reg} puis terminer les calculs en utilisant par exemple l'algorithme F4 pour obtenir la base de Gröbner finale. La partie la plus coûteuse en temps de calcul est en général la première.

Comme précédemment, une suite affine est dite semi-régulière sur $\mathbb{F}_2$ si sa partie homogène de plus haut degré l'est :

Définition 3.5.2 *Soit $f_1, \ldots, f_m \subset R_n$ une suite affine de polynômes, et $I = \langle f_1, \ldots, f_m \rangle \subset R_n$. Pour tout $1 \leq i \leq m$ notons f_i^h la partie homogène de plus haut degré de f_i. La suite $f_1, \ldots, f_m$ est dite semi-régulière sur $\mathbb{F}_2$ si la suite $f_1^h, \ldots, f_m^h$ l'est (avec $f_i^h \in R_n^h$). Le degré de régularité de la suite $f_1, \ldots, f_m$ est l'indice de régularité de la suite $f_1^h, \ldots, f_m^h$.*

Chapitre 4

Analyse asymptotique de la régularité de suites semi-régulières

Dans ce chapitre nous présentons une analyse asymptotique du degré de régularité d'un idéal de dimension zéro défini par une suite semi-régulière. Nous appliquons des méthodes de points cols et de points cols coalescents pour calculer un développement asymptotique de ce degré de régularité lorsque le nombre de variables n tend vers l'infini. Nous donnons de nombreux calculs explicites de ces développements, dont les premiers termes fournissent déjà une très bonne approximation du degré de régularité pour de petites valeurs de n.

4.1 Introduction

Dans la section 3.3 nous avons calculé les séries génératrices associées à des suites semi-régulières de $m > n$ équations en n variables, de degrés $d_1, \ldots, d_m$:

$$S_{m,n}(z) = \sum_{d \geqslant 0} h_{d,m}(n) z^d = \prod_{i=1}^{m} (1 - z^{d_i}) \Big/ (1-z)^n \tag{4.1}$$

dans le cas général et

$$S_{m,n}(z) = \sum_{d \geqslant 0} h_{d,m}(n) z^d = (1+z)^n \Big/ \prod_{i=1}^{m} (1 + z^{d_i}) \tag{4.2}$$

sur $\mathbb{F}_2$. Nous avons vu Section 3.3.3 que le degré de régularité D_{reg} de ces suites est égal au plus petit d tel que $h_{d,m}(n) \leq 0$.

Le but de ce chapitre est de calculer le développement asymptotique de D_{reg} lorsque $n \to \infty$. Pour cela, exprimons les coefficients $h_{d,m}(n)$ de la série génératrice sous la forme d'une intégrale en utilisant les formules de Cauchy :

$$h_{d,m}(n) = \frac{1}{2\imath\pi} \oint \frac{S_{m,n}(z)}{z^{d+1}} dz =: \mathcal{I}_n(d) \tag{4.3}$$

Le chemin d'intégration est un lacet simple, entourant $z = 0$ mais aucun des autres pôles de la série génératrice.

La formule (4.3) définit la fonction $\mathcal{I}_n(d)$ pour tout d entier strictement positif. Nous allons calculer un développement asymptotique de $\mathcal{I}_n(d)$, lorsque $n \to \infty$, en considérant d comme un paramètre. Cela nous permet de définir asymptotiquement la fonction $\mathcal{I}_n(d)$ pour d réel (le domaine de validité dépendant de n). La régularité D_{reg} vérifie $\mathcal{I}_n(d) > 0$ pour tout entier $d < D_{\text{reg}}$ et $\mathcal{I}_n(D_{\text{reg}}) \leq 0$. Nous montrerons que cette propriété reste vraie pour d réel lorsque $n \to \infty$, c'est-à-dire que pour n suffisamment grand, D_{reg} est le premier entier supérieur ou égal au plus petit zéro de $\mathcal{I}_n(d)$.

Dans tout ce chapitre, nous notons $\delta(n)$ le plus petit zéro de $\mathcal{I}_n(d)$. Comme $\delta(n)$ est racine de $\mathcal{I}_n(d)$, la valeur de d annulant le premier terme du développement asymptotique de $\mathcal{I}_n(d)$ donnera le premier terme du développement asymptotique de $\delta(n)$ lorsque $n \to \infty$. Pour obtenir chaque terme supplémentaire, il faut déterminer le coefficient dominant de $\mathcal{I}_n(d)$ au voisinage de $\delta(n)$ et l'annuler.

Le développement asymptotique de $\mathcal{I}_n(d)$ est calculé en utilisant la méthode du col ou des points cols coalescents. L'idée de ces méthodes est de faire passer le chemin d'intégration par les *points cols* de la fonction à intégrer. On montre alors que la contribution des parties du chemin qui ne sont pas voisines des cols est asymptotiquement négligeable, et qu'au voisinage de ces cols la fonction à intégrer peut être approchée par une fonction gaussienne (pour la méthode des cols) ou une fonction d'Airy (pour la méthode des points cols coalescents). Nous décrivons ces méthodes Section 4.2, et renvoyons le lecteur à des ouvrages comme [dB81, Won89] pour plus de détails.

Nous avons choisi dans cette thèse de détailler ces méthodes sur deux exemples particuliers. Nous calculons Section 4.3 le développement asymptotique de la régularité d'une suite semi-régulière de $n + k$ équations en n variables en utilisant la méthode des cols. Cette méthode ne s'applique plus dans le cas de αn équations, il faut alors appliquer la méthode des points cols coalescents, ce que nous détaillons Section 4.4. Nous obtenons les théorèmes suivants :

Théorème 4.1.1 *Le degré de régularité d'une suite semi-régulière de $n + k$ équations de degrés $d_1, \ldots, d_{n+k}$ en n variables se comporte asymptotiquement comme*

$$D_{reg} = \sum_{i=1}^{n+k} \frac{d_i - 1}{2} - \alpha_k \sqrt{\sum_{i=1}^{n+k} \frac{d_i^2 - 1}{6}} + O(1) \tag{4.4}$$

lorsque $n \to \infty$, où α_k est la plus grande racine du $k^{\text{ème}}$ polynôme de Hermite.

Dans la section 4.3, nous donnons tous les détails de la preuve de ce théorème dans le cas où tous les d_i sont égaux à 2. Nous en indiquons les grandes étapes dans le cas général. La proposition suivante est un corollaire de ce théorème dans le cas où tous les polynômes sont de même degré :

Proposition 4.1.2 *Le degré de régularité d'une suite semi-régulière de $n+k$ équations de degré D en n variables se comporte asymptotiquement comme*

$$D_{reg} = n\frac{D-1}{2} - \alpha_k\sqrt{n\frac{D^2-1}{6}} + o(\sqrt{n}) \text{ lorsque } n \to \infty. \tag{4.5}$$

Nous détaillons Section 4.4 les grandes étapes du calcul du développement asymptotique du degré de régularité pour αn équations :

Théorème 4.1.3 *Le degré de régularité d'une suite semi-régulière de αn équations de degrés $d_1, \ldots, d_{\alpha n}$ en n variables, $\alpha > 1$ constant, se comporte asymptotiquement comme*

$$D_{reg} = \phi(z_0)n - a_1\left(-\frac{1}{2}\phi''(z_0)z_0^2\right)^{\frac{1}{3}} n^{\frac{1}{3}} + o(n^{\frac{1}{3}}) \tag{4.6}$$

lorsque $n \to \infty$, où

$$\phi(z) = \frac{z}{1-z} - \frac{1}{n}\sum_{i=1}^{\alpha n}\frac{d_i z^{d_i}}{1-z^{d_i}},$$

z_0 est la racine de $\phi'(z)$ qui minimise $\phi(z_0) > 0$, et a_1 est la plus grande racine de la fonction Ai *d'Airy.*

La proposition suivante est un corollaire de ce théorème dans le cas où tous les polynômes sont quadratiques :

Proposition 4.1.4 *Le degré de régularité d'une suite semi-régulière de αn équations quadratiques, $\alpha > 1$ constant, se comporte asymptotiquement comme*

$$D_{reg} = (\alpha - 1/2 - \sqrt{\alpha(\alpha-1)})n + \frac{-a_1}{2(\alpha(\alpha-1))^{\frac{1}{6}}} n^{\frac{1}{3}} - \left(2 - \frac{2\alpha-1}{4(\alpha(\alpha-1))^{\frac{1}{2}}}\right) + O(\frac{1}{n^{1/3}})$$

Les deux méthodes utilisées permettent d'obtenir tous les termes du développement asymptotique du degré de régularité.

Nous donnons également le développement asymptotique du degré de régularité de suites semi-régulières sur $\mathbb{F}_2$, en utilisant les mêmes méthodes. Ainsi, les quatre premiers termes du développement asymptotique de D_{reg} pour une suite semi-régulière sur $\mathbb{F}_2$ de n équations quadratiques en n variables sont :

Proposition 4.1.5 *Le degré de régularité d'une suite semi-régulière sur $\mathbb{F}_2$ de n équations quadratiques en n variables se comporte asymptotiquement comme*

$$\begin{aligned} D_{reg} = {} & \lambda_0 n - a_1\left(\frac{3}{2}\right)^{\frac{1}{2}}\left(\frac{13}{3\sqrt{3}} - \frac{5}{2}\right)^{\frac{1}{6}} n^{\frac{1}{3}} - 1 - \left(\frac{265}{648} - \frac{85}{36}\lambda_0 - \frac{16}{27}\lambda_0^2 - \frac{38}{81}\lambda_0^3\right)^{\frac{1}{3}} \\ & + \left(\left(\tfrac{31}{108} + \tfrac{71}{216}\lambda_0 + \tfrac{1}{18}\lambda_0^3 + \tfrac{13}{108}\lambda_0^2\right)^{\frac{1}{3}} - \left(\tfrac{152231}{2187000} + \tfrac{1159}{16200}\lambda_0 + \tfrac{2386}{91125}\lambda_0^2 + \tfrac{3062}{273375}\lambda_0^3\right)^{\frac{1}{3}}\right)\frac{a_1^2}{n^{\frac{1}{3}}} \\ & + o(\frac{1}{n^{\frac{1}{3}}}) \end{aligned}$$

où $\lambda_0 = -1/2 + 3/2\sqrt{2/\sqrt{3}-1} \simeq 11.114$ *est la plus petite racine de* $2\lambda^4 + 4\lambda^3 + 12\lambda^2 + 10\lambda - 1$*, ce qui donne numériquement :*

$$D_{reg} = 0.0900\, n + 1.00\, n^{\frac{1}{3}} - 1.58 + \frac{1.41}{n^{\frac{1}{3}}} + o(\frac{1}{n^{\frac{1}{3}}})$$

Ces développements asymptotiques donnent également une très bonne approximation de la régularité pour de petites valeurs de n. La figure 4.1 compare les valeurs exactes de D_{reg}, calculées à partir de la série génératrice, et les formules asymptotiques avec deux ou trois termes, pour des suites semi-régulières sur $\mathbb{F}_2$ de n équations à n inconnues. Par contre, le quatrième terme, en $1.41n^{-\frac{1}{3}}$, ne sera pas petit pour de petites valeurs de n.

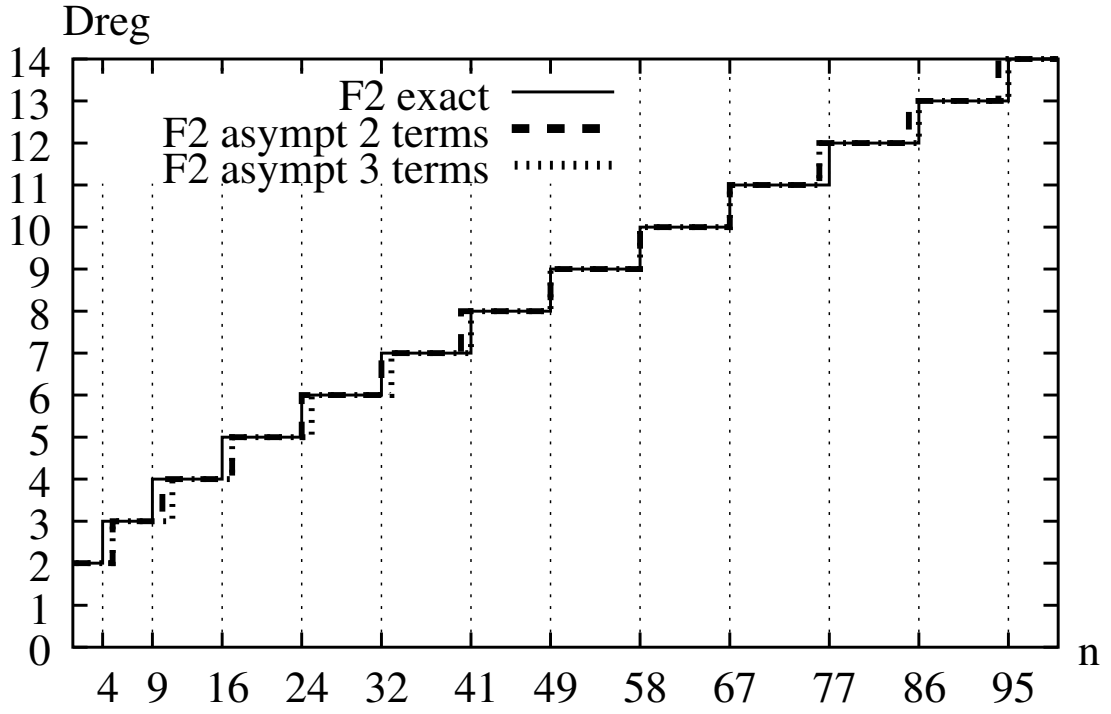


FIG. 4.1 – Comparaison de la régularité et de son développement asymptotique pour $m = n$ équations quadratiques sur $\mathbb{F}_2$.

Organisation du chapitre Nous décrivons les méthodes d'analyse asymptotique employées dans ce chapitre Section 4.2. Nous utilisons la méthode du col Section 4.3 pour calculer le développement asymptotique de la régularité dans le cas de $n + k$ équations en n variables. Nous donnons une preuve intégrale du théorème 4.1.1 lorsque tous les polynômes sont quadratiques, et esquissons la preuve dans le cas général. La méthode des points cols coalescents est utilisée Section 4.4 pour calculer

le développement asymptotique de la régularité de suites semi-régulières de αn polynômes en n variables. Nous esquissons les preuves du théorème 4.1.3. Enfin, dans la section 4.4.3 nous donnons d'autres exemples de développements asymptotiques de la régularité pour des suites semi-régulières sur $\mathbb{F}_2$, obtenus en utilisant l'une des deux méthodes présentées.

Les résultats de ce chapitre ont été obtenus en collaboration avec Bruno Salvy. Ils ont fait l'objet d'une présentation à la conférence ICPSS en Novembre 2004 [BFS04]. Pour le cas particulier d'équations à coefficients et solutions dans $\mathbb{F}_2$ un article est disponible sous forme de rapport de recherche INRIA [BFS03].

4.2 Description des méthodes employées

Nous décrivons brièvement deux méthodes qui permettent, sous certaines conditions, de calculer un développement asymptotique d'une intégrale de la forme

$$I = \oint e^{tf(z)} g(z) dz,$$

lorsque $t \to \infty$.

Les deux méthodes considèrent les points cols de la fonction à intégrer, qui sont les points z_0 tels que $f'(z_0) = 0$, et montrent qu'asymptotiquement, l'essentiel de l'intégrale est concentrée au voisinage de ces points cols.

Nous donnons une description générale de la méthode du col section 4.2.1. Dans le cas où la fonction $f(z)$ est paramétrée, pour toute valeur fixe du paramètre λ la méthode du col va fournir un développement asymptotique de l'intégrale. Le problème se pose lorsque, pour une valeur λ_0 du paramètre, deux cols coalescent. Dans ce cas, les approximations obtenues par la méthode du col pour $\lambda \neq \lambda_0$ ne tendent pas vers celle pour $\lambda = \lambda_0$: la méthode des cols ne donne pas d'approximation uniforme au voisinage de $\lambda = \lambda_0$. Nous décrivons Section 4.2.2 la méthode des cols coalescents de Chester-Friedmann-Ursell, qui permet de donner un développement asymptotique uniforme au voisinage de λ_0.

4.2.1 Méthode du col

Un point col est dit d'ordre k si $f^{(i)}(z_0) = 0$ pour $1 \leq i \leq k$ et $f^{(k+1)}(z_0) \neq 0$. Un point col d'ordre 1 est appelé point col simple. La méthode du col s'applique quel que soit l'ordre des points cols, mais nous la décrivons dans le cas ou tous les cols sont simples (qui sera le cas rencontré dans nos applications).

D'après le théorème de Cauchy, on peut déformer le lacet d'intégration sans changer la valeur de l'intégrale. La méthode du col fait passer le lacet d'intégration par certains points cols simples, et montre que l'intégrale est essentiellement donnée par la somme des contributions de ces points cols, la contribution des autres parties

étant asymptotiquement négligeable :

$$I \sim \sum_{z_0 \text{ point col simple}} e^{tf(z_0)} \sqrt{\frac{2\pi}{-f''(z_0)t}} g(z_0) \tag{4.7}$$

Nous voyons sur cette formule que seuls les points cols pour lesquels $|e^{f(z_0)}|$ est maximal vont contribuer asymptotiquement à l'intégrale.

Décrivons les trois étapes qui permettent d'obtenir la formule (4.7). Nous montrons successivement que :

1. Seules les parties du lacet au voisinage des cols contribuent asymptotiquement à l'intégrale,
2. Au voisinage de ces points cols, la fonction sous l'intégrale peut être approchée par une fonction gaussienne, intégrée sur un voisinage de 0,
3. Cette fonction gaussienne peut être intégrée sur $\mathbb{R}$ tout entier sans changer la valeur asymptotique de l'intégrale.

Plus précisément :

1. Pour z voisin d'un point col z_0, on a $f(z) = f(z_0) + \frac{1}{2}f''(z_0)(z - z_0)^2 + \ldots$. Choisissons la direction du lacet de sorte que $\frac{1}{2}f''(z_0)(z - z_0)^2$ soit réel négatif au voisinage de z_0. En faisant le changement de variable $u^2 = -\frac{t}{2}f''(z_0)(z - z_0)^2$, la variable u sera réelle au voisinage de $u = 0$ (disons dans l'intervalle $(-\epsilon, \epsilon)$). Le changement de variable s'écrit $u = \sqrt{-\frac{t}{2}f''(z_0)}(z - z_0)$, la racine étant choisie telle que du soit positif lorsque dz a la direction positive de circulation du lacet, et l'on se ramène à l'intégrale :

$$I = e^{tf(z_0)} \int_{-\epsilon}^{\epsilon} g(z(u)) e^{-u^2 + O(u^3)} \sqrt{\frac{2}{-f''(z_0)t}} du + R'$$

où R' correspond à l'intégrale I sur le reste du lacet (en dehors du voisinage de z_0).

2. Nous obtenons alors une intégrale réelle pour laquelle nous appliquons la méthode de Laplace, en approchant la fonction sous l'intégrale :

$$I = e^{tf(z_0)} \sqrt{\frac{2}{-f''(z_0)t}} \int_{-\epsilon}^{\epsilon} g(z_0) e^{-u^2} du + R' + R''$$

où R'' correspond à l'erreur commise en approchant la fonction $g(z(u))e^{-u^2+O(u^3)}$ par $g(z_0)e^{-u^2}$ sur l'intervalle $(-\epsilon, \epsilon)$,

3. Nous écrivons enfin

$$I = e^{tf(z_0)} \sqrt{\frac{2}{-f''(z_0)t}} g(z_0) \int_{-\infty}^{\infty} e^{-u^2} du + R' + R'' + R'''$$

où R''' est l'erreur commise en intégrant la gaussienne sur tout l'intervalle $\mathbb{R}$,

Il reste à montrer que R', R'' et R''' sont asymptotiquement négligeables devant $e^{tf(z_0)}\sqrt{\frac{2}{-f''(z_0)t}}g(z_0)\int_{-\infty}^{\infty}e^{-u^2}du$; on obtient alors la formule (4.7).

Pour obtenir les termes suivants du développement asymptotique de I, il suffit d'approcher plus précisément $g(z(u))e^{-u^2+O(u^3)} = (\sum_{k\geq 0}a_k u^k)e^{-u^2}$. On se ramène alors à des sommes d'intégrales de la forme

$$\int_{-\infty}^{\infty}e^{-tu^2}u^{2k+1}du = 0 \text{ pour } k\geq 0 \tag{4.8}$$

$$\int_{-\infty}^{\infty}e^{-tu^2}u^{2k}du = \frac{1}{t^{k+1/2}}\Gamma(k+\frac{1}{2}) \text{ pour } k\geq 0$$

$$= \frac{1}{t^{k+1/2}}\frac{(2k)!}{k!2^{2k}}\sqrt{\pi}. \tag{4.9}$$

4.2.2 Méthode des points cols coalescents

Supposons maintenant que la fonction $f(z)$ dépende d'un paramètre λ, tel que pour $\lambda\neq\lambda_0$ on ait deux points cols z_0^+ et z_0^-, et pour $\lambda=\lambda_0$ un unique point col double z_0. En utilisant la méthode du col, on obtient un développement asymptotique en $(tf''(z_0^{\pm}))^{-1/2}$ lorsque $\lambda\neq\lambda_0$ et en $(tf'''(z_0))^{-1/3}$ lorsque $\lambda=\lambda_0$. Ces deux équivalents sont radicalement différents : le premier devient singulier lorsque $\lambda\to\lambda_0$, et t est d'ordre $-1/2$ dans le premier et $-1/3$ dans le second.

La méthode des cols coalescents [CFU57] donne un équivalent asymptotique de I uniforme, valable pour λ dans un voisinage de λ_0. L'idée est d'introduire un changement de variable cubique de la forme $f(z)=\frac{u^3}{3}-\zeta u+\eta$, pour se ramener à l'intégrale d'une fonction d'Airy (voir annexe A.4 page 146),

$$\mathrm{Ai}(x)=\frac{1}{2i\pi}\int_{C_1}e^{\frac{v^3}{3}-xv}dv$$

où C_1 est un chemin d'origine un point à l'infini dans le secteur $-\frac{\pi}{2}\leq\arg(v)\leq-\frac{\pi}{6}$ et de fin un point dans le secteur conjugué. Les constantes ζ et η dépendent évidemment de λ. Pour que le changement de variable soit régulier, les points cols z_0^+ et z_0^- doivent correspondre aux points cols $u_0^{\pm}=\pm\zeta^{1/2}$ de la nouvelle fonction en u, et coalescer en $u_0=0$ lorsque $\lambda=\lambda_0$. Cela impose de choisir

$$\zeta^{\frac{3}{2}}=\frac{3}{4}(f(z_0^-)-f(z_0^+)) \quad \text{et} \quad \eta=\frac{1}{2}(f(z_0^-)+f(z_0^+))$$

En choisissant bien le chemin d'intégration au voisinage des points cols, et en supposant pour simplifier que $g(z)=1$, nous obtenons

$$I=e^{t\eta}\int_{C_1\cap D}e^{t(\frac{u^3}{3}-\zeta u)}\frac{dz(u)}{du}du+R'$$

où D est un voisinage de $u_0^{\pm}$ et R' est l'intégrale sur le reste du chemin. Écrivons

$$\frac{dz(u)}{du} = \sum_{i\geq 0}(u^2-\zeta)^i(b_i+c_iu).$$

En intégrant terme à terme et en remplaçant le chemin $C_1 \cap D$ par C_1 entier, nous obtenons

$$I = e^{t\eta}\int_{C_1}\sum_{l\geq 0} e^{t(\frac{u^3}{3}-\zeta u)}(u^2-\zeta)^l(b_l+c_lu)du + R' + R'''$$

$$I = e^{t\eta}2i\pi\left[\frac{\mathrm{Ai}(t^{\frac{2}{3}}\zeta)}{t^{\frac{1}{3}}}\sum_{0\leq m\leq M}\frac{B_m}{t^m} + \frac{\mathrm{Ai}'(t^{\frac{2}{3}}\zeta)}{t^{\frac{2}{3}}}\sum_{0\leq m\leq M}\frac{C_m}{t^m}\right] + R' + R''' + R''_M \quad (4.10)$$

où les coefficients B_m et C_m peuvent s'exprimer en fonction des coefficients b_m et c_m (posons $G_0(u) = \frac{dz}{du}$, alors pour tout $m \geq 0$, on a $G_m(u) = B_m + C_m u + (u^2-\zeta)H_m(u)$ et $G_{m+1}(u) = \frac{dH_m}{du}(u)$).

Comme dans la méthode du col, pour prouver la formule (4.10) il faut montrer que R', R''_M pour tout $M > 0$ et R''' sont asymptotiquement négligeables. Les auteurs de [CFU57] prouvent, sous des hypothèses simples de régularité des fonctions intégrées (par exemple elles doivent être analytiques en z et λ), que R''' est négligeable et que

$$\exists C_M, D_M > 0 : \ |R''_M| \leq \frac{C_M}{t^{M+1/3}}|\mathrm{Ai}(t^{\frac{2}{3}}\zeta)| + \frac{D_M}{t^{M+2/3}}|\mathrm{Ai}'(t^{\frac{2}{3}}\zeta)|.$$

4.3 Le cas de $n+k$ équations

Nous étudions dans cette section la régularité d'une suite semi-régulière de $n+k$ équations de degrés $d_1, \ldots, d_{n+k}$ en n variables. La formule (4.3) devient :

$$\mathcal{I}_n(d) = \frac{1}{2i\pi}\oint \underbrace{\prod_{i=1}^{n+k}\frac{1-z^{d_i}}{1-z}\frac{1}{z^{d+1}}}_{=F(z)}\underbrace{(1-z)^k}_{=g(z)}dz$$

Nous montrons Section 4.3.1 comment obtenir $\delta(n)$ lorsque l'on connaît le développement asymptotique de $\mathcal{I}_n(d)$, qui sera calculé Section 4.3.2. Nous prouvons également que le degré de régularité est bien $D_{\mathrm{reg}} = \lceil \delta(n) \rceil$.

4.3.1 Asymptotique du degré de régularité

Nous utilisons ici le lemme suivant. Il sera démontré Section 4.3.2 dans le cas où tous les $d_i = 2$. Dans le cas général, la preuve semble beaucoup plus technique et ne sera qu'esquissée :

Lemme 4.3.1 *Le premier terme du développement asymptotique de $\mathcal{I}_n(d)$ est*

$$\mathcal{I}_n(d) \sim \sqrt{\frac{1}{2\pi f''(z_0)}} g(z_0) F(z_0) \tag{4.11}$$

où $f(z) = \sum_{i=1}^{n+k} \log\left(\frac{1-z^{d_i}}{1-z}\right) - (d+1)\log(z)$, et z_0 est la racine réelle positive de $f'(z)$.

Si, de plus, z_0 tend vers 1 lorsque n tend vers l'infini, soit $z_0 = 1 - \Delta z$ avec $\Delta z \to 0$, un nouvel équivalent asymptotique de $\mathcal{I}_n(d)$ est donné par :

$$\mathcal{I}_n(d) \sim \frac{F(z_0)}{\sqrt{\pi}\left(\sqrt{-2f''(z_0)}\right)^{k+1}} H_k\left(\frac{\Delta z}{2}\sqrt{\sum_{i=1}^{n+k}\frac{d_i^2-1}{6}}(1+o(1))\right) \tag{4.12}$$

où H_k est le $k^{ème}$ polynôme de Hermite.

On a $H_k(x) = \frac{2^k}{\sqrt{\pi}}\int_{-\infty}^{\infty}(x+iu)^k e^{-u^2}du = \sum_{j=0}^{\lfloor \frac{k}{2} \rfloor}(-1)^j \frac{2^{k-2j}k!}{j!(k-2j)!}x^{k-2j}$. Les premières valeurs de α_k sont données Figure 4.2.

Valeur de la plus grande racine α_k du $k^{\text{ème}}$ polynôme de Hermite :					
k	1	2	3	4	5
α_k	0	$\frac{1}{\sqrt{2}} \simeq 0.707$	$\sqrt{\frac{3}{2}} \simeq 1.22$	$\left(\frac{3}{2}+\sqrt{\frac{3}{2}}\right)^{\frac{1}{2}} \simeq 1.65$	$\alpha_5 = \left(\frac{5}{2}+\sqrt{\frac{5}{2}}\right)^{\frac{1}{2}} \simeq 2.02$

FIG. 4.2 – Racines des premiers polynômes de Hermite

Pour approcher une racine de $\mathcal{I}_n(d)$, nous annulons le premier terme du développement asymptotique de l'intégrale. Comme z_0 est réel > 0, alors $F(z_0)$ n'est jamais nul, il faut donc que $g(z_0) \sim 0$ i.e. $z_0 \sim 1$. De l'équation

$$f'(z_0) = \sum_{i=1}^{n+k} \frac{1 + 2z_0 + \ldots + (d_i-1)z_0^{d_i-2}}{1 + z_0 + \ldots + z_0^{d_i-1}} - \frac{d+1}{z_0} = 0, \tag{4.13}$$

on déduit $d+1 \sim \sum_{i=1}^{n+k}\frac{d_i-1}{2}$, ce qui donne le premier terme de l'équation (4.4).

Lorsque z_0 est voisin de 1, soit $z_0 = 1-\Delta z$ avec $\Delta z \to 0$, pour annuler le terme de droite de l'équation (4.12) il faut que $\Delta z = 2\alpha_k/\sqrt{\sum_{i=1}^{n+k}\frac{d_i^2-1}{6}}$ où α_k est une racine de H_k. En utilisant l'équation (4.13), on obtient $d+1 = \sum_{i=1}^{n+k}\frac{d_i-1}{2} - \frac{d_i^2-1}{12}\Delta z + o(\Delta z)$, ce qui donne

$$d+1 = \sum_{i=1}^{n+k}\frac{d_i-1}{2} - \alpha_k\sqrt{\sum_{i=1}^{n+k}\frac{d_i^2-1}{6}}(1+o(1))$$

Pour obtenir la plus petite racine de $\mathcal{I}_n(d)$, il faut prendre pour α_k la plus grande racine de $H_k(z)$.

Il nous reste à montrer que, asymptotiquement, le degré de régularité est bien la partie entière supérieure de la plus petite racine de $\mathcal{I}_n(d)$. Si cette propriété était fausse, il n'y aurait aucun entier entre les deux plus petites racines de $\mathcal{I}_n(d)$. Or, l'écart entre les deux plus petites racines de $\mathcal{I}_n(d)$ est donné asymptotiquement par

$$\beta_k \sqrt{\sum_{i=1}^{n+k} \frac{d_i^2 - 1}{6}}$$

où $\beta_k > 0$ est la différence entre les deux plus grandes racines de $H_k(z)$, et il tend vers l'infini lorsque $n \to \infty$, ce qui montre que le degré de régularité est bien donné asymptotiquement par la plus petite racine de $\mathcal{I}_n(d)$.

4.3.2 Asymptotique de $\mathcal{I}_n(d)$

Cette section est consacrée à la preuve du lemme 4.3.1.

Dans le cas particulier d'équations quadratiques, nous donnons une preuve complète du lemme. La fonction F vaut alors $F(z) = \frac{(1+z)^{n+k}}{z^{d+1}} = e^{f(z)}$. Les cols vérifient $f'(z_0) = \frac{n+k}{1+z_0} - \frac{d+1}{z_0} = \frac{z_0(n+k-d-1)-(d+1)}{z_0(1-z_0)} = 0$ (équation 4.13). Le seul point col est $z_0 = \frac{1}{\frac{n+k}{d+1}-1}$, et il est réel.

Nous donnons dans le dernier paragraphe le schéma de la preuve pour le cas général. En particulier, nous admettons l'existence d'un point col z_0 strictement positif qui "capture" l'essentiel de l'intégrale, et nous appliquons la méthode du col au voisinage de z_0, en admettant la validité de chacune des étapes.

Restriction du domaine pour d Une première analyse montre que l'on peut se restreindre au cas où $1 < \epsilon_1 \leq \frac{n+k}{d+1} \leq \epsilon_2 < \infty$. En effet, pour n équations le degré de régularité vaut $d = n + 1$ et il diminue lorsque l'on ajoute des équations, donc on peut supposer $d \leq n + 1 \leq n + k$. De plus, pour d, n, k entiers, on a

$$\begin{aligned} \mathcal{I}_n(d) &= \sum_{l=0}^{k} (-1)^l \binom{k}{l} \frac{1}{2i\pi} \oint \frac{(1+z)^{n+k}}{z^{d+1-l}} dz \\ &= \sum_{l=0}^{\min(k,d)} u_l \quad \text{avec} \quad u_l = (-1)^l \binom{k}{l} \binom{n+k}{d-l} \end{aligned}$$

Or, pour $d = o(n)$, la somme est dominée par le premier terme (car $\frac{u_{l+1}}{u_l} \to 0$), qui tend vers l'infini lorsque $n \to 0$ (car $\log(u_0) \sim d \log(n)$), donc $\mathcal{I}_n(d)$ ne possède pas de racine en $o(n)$. Si $d \sim n$, alors le terme dominant de la somme est le plus grand ($\frac{u_l}{u_{l+1}} \to 0$), et $\log(u_k) \sim (n-d+2k)\log(n) \to +\infty$ donc $\mathcal{I}_n(d)$ ne peut pas s'annuler pour $d \sim n$. Nous nous restreignons donc au cas où $1 < \epsilon_1 \leq \frac{n+k}{d+1} \leq \epsilon_2 < \infty$ avec ϵ_1 et ϵ_2 des constantes.

Fixation du chemin d'intégration Au voisinage du col, on a $f(z) = f(z_0) + \frac{f''(z_0)}{2}(z-z_0)^2 + o(z-z_0)^2$. Or, $f''(z_0) = \frac{(d+1)^2(\frac{n+k}{d+1}-1)^3}{n+k} > 0$. Il faut donc choisir le contour de sorte que $z-z_0$ soit imaginaire pur au voisinage de z_0, alors $\frac{f''(z_0)}{2}(z-z_0)^2$ est réel négatif.

Le lacet d'intégration sera constitué :

– d'un segment vertical L, ayant pour milieu z_0. Notons z_1 et z_2 ses extrémités, z_1 étant de partie imaginaire négative, sa longueur sera fixée ultérieurement,
– d'un arc de cercle C, de centre 0, joignant z_1 et z_2, et coupant l'axe réel négatif.

L'intégrale s'écrit $\mathcal{I}_n(d) = \frac{1}{2i\pi}\int_L F(z)g(z)dz + \frac{1}{2i\pi}\int_C F(z)g(z)dz = I_L + I_C$, et nous montrerons section 4.3.3 que, sous certaines hypothèses sur la longueur du segment L,

$$\left|\frac{I_C}{F(z_0)}\right| = O(\frac{1}{n^M}) \text{ pour tout } M > 0 \tag{4.14}$$

Intéressons nous à la portion d'intégrale au voisinage du col.

La contribution du voisinage du col Posons $z - z_0 = i\zeta$, on a alors

$$\frac{F(z)}{F(z_0)} = \exp\left((n+k)\log\left(1+\frac{i\zeta}{1+z_0}\right) - (d+1)\log\left(1+\frac{i\zeta}{z_0}\right)\right)$$

en utilisant le logarithme usuel (défini sur $\mathbb{C}\backslash\mathbb{R}^-$ et valant $\log(1) = 0$ pour $\zeta = 0$). Pour $|\zeta| < \max(|1+z_0|, |z_0|)$, nous avons le développement en séries entières :

$$\frac{F(z)}{F(z_0)} = \exp\left((n+k)\sum_{j=1}^{\infty}\frac{i^j(-1)^{j+1}\zeta^j}{j(1+z_0)^j} - (d+1)\sum_{j=1}^{\infty}\frac{i^j(-1)^{j+1}\zeta^j}{jz_0^j}\right)$$

Or, l'équation du col donne $d+1 = (n+k)\frac{z_0}{1+z_0}$, et donc

$$\begin{aligned}\frac{F(z)}{F(z_0)} &= \exp\left[(n+k)\left(\sum_{j=2}^{\infty}\frac{i^j(-1)^{j+1}\zeta^j}{j(1+z_0)}\left(\frac{1}{(1+z_0)^{j-1}} - \frac{1}{z_0^{j-1}}\right)\right)\right] \\ &= \exp\left[(n+k)\left(-\frac{\zeta^2}{2(1+z_0)^2 z_0} + \sum_{j=3}^{\infty}\frac{i^j(-1)^{j+1}\zeta^j}{j(1+z_0)}\left(\frac{1}{(1+z_0)^{j-1}} - \frac{1}{z_0^{j-1}}\right)\right)\right]\end{aligned}$$

Posons $u = \frac{-\zeta}{(1+z_0)\sqrt{2z_0}} = \frac{i}{(1+z_0)\sqrt{2z_0}}(z-z_0)$, notons $\theta_0 z_0$ la demi-longueur du segment L, alors sur ce segment u varie de $-N$ à N avec $N = \frac{\theta_0}{1+z_0}\sqrt{\frac{z_0}{2}}$, et on obtient

$$I_L = \frac{(1+z_0)\sqrt{2z_0}}{2\pi}F(z_0)\int_{-N}^{N} g(z(u))\exp\left[(n+k)\left(-u^2 + u^3\sum_{j=3}^{\infty}a_j u^{j-3}\right)\right]du$$

avec $a_j = -\frac{i^j 2^{j/2} z_0^{j/2}}{j}\left(\left(\frac{1+z_0}{z_0}\right)^{j-1} - 1\right)$ qui vérifie $|a_j| \leq \frac{2^{j/2}}{j}\frac{\epsilon_2^{j-1}}{(\epsilon_1-1)^{j/2}}$.

Méthode de Laplace Nous nous sommes ramenés à une intégrale réelle, pour laquelle nous allons pouvoir utiliser la méthode de Laplace [dB81, page 67-68]. Nous avons $g(z(u)) = (1-z(u))^k = (1-z_0 - i(1+z_0)\sqrt{2z_0}u)^k = \sum_{j=0}^{k} g_j u^j$ avec $g_j = (1-z_0)^{k-j} i^j z_0^{j/2} 2^{j/2} (1+z_0)^j \binom{k}{j}$, qui est également borné uniformément en d : $|g_j| \leq \binom{k}{j} \frac{2^{j/2}}{(\epsilon_1 - 1)^{k+j/2}} \epsilon_1^j$.

Développons en série entière la fonction sous l'intégrale :

$$\begin{aligned} P((n+k)u^3, u) &= (\sum_{j=0}^{k} g_j u^j) \exp\left[(n+k)u^3 \sum_{j=3}^{\infty} a_j u^{j-3}\right] \\ &= \sum_{j=0}^{\infty} \sum_{l=0}^{\infty} c_{j,l} ((n+k)u^3)^j u^l, \end{aligned}$$

les coefficients $c_{j,l}$ étant bornés uniformément en n et u. Nous ne détaillerons pas les calculs, donnés dans [dB81], qui montrent que, si $\theta = \frac{1}{n^\alpha}$ avec $\alpha > \frac{1}{3}$, alors[1] $\frac{I_L}{F(z_0)}$ peut être approché uniformément par l'intégrale sur $\mathbb{R}$ des sommes partielles de P. Nous obtenons, pour tout $A > 0$, $M > 0$, et lorsque $n \to \infty$:

$$\frac{I_L}{F(z_0)} \sim \sum_{j,l \geq 0, j+l \text{ pair}, j+l \leq A} c_{j,l} \frac{\sqrt{2z_0}(1+z_0)}{2\pi (n+k)^{\frac{j+l+1}{2}}} \Gamma\left(\frac{3j+l+1}{2}\right) + O\left(\frac{1}{n^{\frac{A}{2}+1}}\right) + O\left(\frac{1}{n^M}\right) \tag{4.15}$$

Terme dominant On obtient le terme dominant de la formule (4.15) pour $j = l = 0$, et comme $c_{0,0} = g(z_0)$ on obtient :

$$\mathcal{I}_n(d) \sim \frac{\sqrt{z_0}(1+z_0)}{\sqrt{2\pi}} F(z_0) g(z_0) \frac{1}{(n+k)^{1/2}} \tag{4.16}$$

Terme dominant au voisinage de $z_0 = 1$ Notons $z_0 = 1 - \Delta z$ avec $\Delta z \to 0$ lorsque $n \to \infty$, alors $g_j \sim \binom{k}{j} 2^{3j/2} i^j (\Delta z)^{k-j}$ et a_j tend vers une constante non nulle. On peut alors montrer que $c_{j,l}$ se comporte comme $(\Delta z)^{\min(k,l)}$, et le terme dominant de la formule (4.15) est obtenu pour $j = 0$, ce qui donne

$$\begin{aligned} I_L &\sim \left(\frac{2^{n+k+1/2}}{\pi} + o(1)\right) \int_{-\infty}^{\infty} g(z(u)) e^{-(n+k)u^2} du \\ &= \left(\frac{2^{n+k+1/2}}{\pi} + o(1)\right) \int_{-\infty}^{\infty} (1 - z_0 + i(1+z_0)\sqrt{2z_0}u)^k e^{-(n+k)u^2} du \end{aligned}$$

[1] Alors $nu^3 \leq \frac{n\theta_0^3}{2^{3/2}(1+z_0)^{3/2}} \leq n^{1-3\alpha} = o(1)$ donc $nu^3 \leq 1$ sur l'intervalle $[-N, N]$ pour n suffisamment grand.

On reconnaît l'intégrale $H_k(x) = \frac{2^k}{\sqrt{\pi}} \int_{-\infty}^{\infty} (x+iu)^k e^{-u^2} du$ qui définit le $k^{\text{ème}}$ polynôme de Hermite, et en normalisant on obtient

$$I_L \sim \frac{2^{n+k+1/2}}{\pi} \left(\frac{(1+z_0)\sqrt{2z_0}}{\sqrt{n+k}} \right)^k \frac{\sqrt{\pi}}{2^k \sqrt{n+k}} H_k(x) \text{ avec } x = \frac{(1-z_0)\sqrt{n+k}}{(1+z_0)\sqrt{2z_0}}$$

et finalement

$$\mathcal{I}_n(d) \sim \frac{2^{n+\frac{3}{2}k+\frac{1}{2}}}{\sqrt{\pi}\sqrt{n+k}^{k+1}} H_k\left(\frac{\sqrt{n+k}}{2^{3/2}} \Delta z(1+o(1)) \right) \tag{4.17}$$

Cas général Dans le cas général, on a $F(z) = \prod_{i=1}^{n+k} \frac{1-z^{d_i}}{1-z} \frac{1}{z^{d+1}} = e^{f(z)}$. L'équation du col est

$$f'(z) = \sum_{i=1}^{n+k} \frac{g_i'(z)}{g_i(z)} - \frac{d+1}{z} \text{ avec } g_i(z) = \frac{1-z^{d_i}}{1-z} \tag{4.18}$$

Nous admettons que l'essentiel de l'intégrale est concentrée autour d'un unique point col réel positif, que nous noterons z_0.

Nous appliquons comme précédemment la méthode du col. Le changement de variable $u = i\sqrt{\frac{f''(z_0)}{2}}(z-z_0)$ permet de se ramener à une intégrale réelle, à laquelle on applique la méthode de Laplace.

Le premier terme du développement asymptotique est alors

$$\mathcal{I}_n(d) \sim \sqrt{\frac{2}{f''(z_0)}} g(z_0) F(z_0) \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-u^2} du = \sqrt{\frac{1}{2\pi f''(z_0)}} g(z_0) F(z_0) \tag{4.19}$$

Au voisinage de $z_0 = 1$, en posant $z = 1-\Delta z$, le premier terme du développement asymptotique de I devient :

$$\begin{aligned} \mathcal{I}_n(d) &\sim \frac{1}{2\pi} \sqrt{\frac{2}{f''(z_0)}} F(z_0) \int_{-\infty}^{\infty} g(z(u)) e^{-u^2} du \\ &\sim \left(\sqrt{\frac{2}{f''(z_0)}} \right)^{k+1} \frac{1}{2^{k+1}\sqrt{\pi}} F(z_0) H_k\left((1-z_0)\sqrt{\frac{f''(z_0)}{2}} \right) \end{aligned}$$

Estimons $f''(z_0)$ lorsque $n \to \infty$: on a

$$f''(z) = \frac{d+1}{z^2} + \sum_{i=1}^{n+k} \frac{g_i''(z)}{g_i(z)} - \frac{g_i'(z)^2}{g_i(z)^2}$$

ce qui donne, en utilisant l'équation (4.18) pour éliminer d :

$$f''(z_0) = \sum_{i=1}^{n+k} \frac{g_i''(z_0)}{g_i(z_0)} - \frac{g_i'(z_0)^2}{g_i(z_0)^2} + \frac{g_i'(z_0)}{z_0 g_i(z_0)} = \sum_{i=1}^{n+k} \frac{d_i^2 - 1}{12} + O(\Delta z)$$

et l'on obtient ainsi :

$$\mathcal{I}_n(d) \sim \frac{F(z_0)}{\sqrt{\pi}\left(\sqrt{2f''(z_0)}\right)^{k+1}} H_k\left(\frac{\Delta z}{2}\sqrt{\sum_{i=1}^{n+k}\frac{d_i^2-1}{6}}(1+o(1))\right) \tag{4.20}$$

4.3.3 Estimation de l'intégrale en dehors du col

Nous prouvons dans cette section l'équation (4.14) :

$$\left|\frac{I_C}{F(z_0)}\right| = O(\frac{1}{n^M}) \text{ pour tout } M>0 \text{ lorsque } \theta_0 = \frac{1}{n^\alpha} \text{ et } \frac{1}{4}<\alpha<\frac{1}{2}$$

Pour cela, écrivons

$$\frac{I_C}{F(z_0)} = \frac{F(z_2)}{F(z_0)}\int_C g(z)\frac{F(z)}{F(z_2)}dz$$

Comme $z_2 = z_0(1+i\theta_0)$, on a $\left|\frac{F(z_2)}{F(z_0)}\right| = \left|1+i\frac{\theta_0 z_0}{1+z_0}\right|^{n+k}\left|\frac{1}{1+i\theta_0}\right|^{d+1}$, ce qui donne, en utilisant l'équation du col $(n+k)z_0 = (1+z_0)(d+1)$ et l'encadrement $x-x^2/2 \le \log(1+x) \le x$:

$$\begin{aligned}\frac{2}{n+k}\log\left|\frac{F(z_2)}{F(z_0)}\right| &= \log\left(1+\frac{\theta_0^2 z_0^2}{(1+z_0)^2}\right) - \frac{z_0}{1+z_0}\log(1+\theta_0^2)\\ &\le \theta_0^2\left(-\frac{\epsilon_1-1}{\epsilon_1\epsilon_2}+\frac{\theta_0^2}{\epsilon_2}\right)\end{aligned}$$

Posons alors $\theta_0 = \frac{1}{n^\alpha}$, on obtient

$$\left|\frac{F(z_2)}{F(z_0)}\right| \le \exp\left(-\frac{\epsilon_1-1}{2\epsilon_1\epsilon_2}\frac{n+k}{n^{2\alpha}}+\frac{1}{2\epsilon_2}\frac{n+k}{n^{4\alpha}}\right)$$

En choisissant $\frac{1}{4}<\alpha<\frac{1}{2}$, on obtient

$$\left|\frac{F(z_2)}{F(z_0)}\right| \le 2\exp\left(-\frac{(\epsilon_1-1)n^{1-2\alpha}}{2\epsilon_1\epsilon_2}\right) \text{ pour } n \text{ assez grand.}$$

Il reste à estimer l'intégrale sur l'arc de cercle C, qui est un cercle de rayon $\rho = z_0\sqrt{1+\theta_0^2}$:

$$\begin{aligned}\left|\int_C g(z)\frac{F(z)}{F(z_2)}dz\right| &= \left|\left\{\int_{\arcsin(\theta_0)}^{\pi} + \int_{-\pi}^{-\arcsin(\theta_0)}\right\}\frac{F(\rho e^{i\theta})}{F(z_2)}g(\rho e^{i\theta})i\rho e^{i\theta}d\theta\right|\\ &\le 2\int_{\arcsin(\theta_0)}^{\pi}\left|\frac{1+\rho e^{i\theta}}{1+z_2}\right|^{n+k}\rho|1-\rho e^{i\theta}|^k d\theta\\ &\le 2\rho|1+\rho|^k\pi\end{aligned}$$

car $|\rho e^{i\theta}| = |z_2|$ sur le cercle, $|1-\rho e^{i\theta}| \leq |1+\rho|$ et $|1+\rho e^{i\theta}| \leq |1+\rho e^{i\arcsin(\theta_0)}| = |1+z_2|$ pour $\theta \in [\arcsin(\theta_0), \pi]$. Comme, de plus, $|\rho| \leq \frac{\sqrt{1+\theta_0^2}}{\epsilon_1 - 1} \leq \frac{2}{\epsilon_1 - 1}$ pour n assez grand, et $|1+\rho| \leq 1 + \frac{2}{\epsilon_1 - 1} = \frac{\epsilon_1 + 1}{\epsilon_1 - 1}$, on obtient, pour n assez grand :

$$\left|\frac{I_C}{F(z_0)}\right| \leq \frac{8\pi}{\epsilon_1 - 1}\left(\frac{\epsilon_1+1}{\epsilon_1-1}\right)^k \exp\left(-\frac{(\epsilon_1-1)n^{1-2\alpha}}{2\epsilon_1}\right) = O(\frac{1}{n^M}) \text{ pour tout } M > 0$$

pourvu que $\frac{1}{4} < \alpha < \frac{1}{2}$.

4.4 Le cas de αn équations

Considérons maintenant le cas d'une suite semi-régulière de αn équations de degrés $d_1, \ldots, d_{\alpha n}$ en n variables, où $\alpha > 1$ est une constante.. La formule (4.3) devient :

$$\mathcal{I}_n(d) = \oint \underbrace{\frac{\prod_{i=1}^{\alpha n}(1-z^{d_i})}{(1-z)^n}\frac{1}{z^{d+1}}}_{=F(z)=e^{nf(z)}} dz$$

L'équation de col est :

$$zf'(z) = \frac{1}{n}\sum_{i=1}^{\alpha n} -\frac{d_i z^{d_i}}{1-z^{d_i}} + \frac{z}{1-z} - \frac{(d+1)}{n} =: \phi(z) - \frac{(d+1)}{n} \tag{4.21}$$

et les points cols sont les racines de $n\phi(z) = d+1$.

4.4.1 Premier terme : méthode du col

Commençons par étudier le cas d'équations quadratiques. La fonction intégrée est $F(z) = \frac{(1-z^2)^{\alpha n}}{(1-z)^n}\frac{1}{z^{d+1}} = e^{nf(z)}$. L'équation des cols est

$$zf'(z) = \frac{-((2\alpha-1) - \frac{d+1}{n})z^2 + z - \frac{d+1}{n}}{(1-z^2)}$$

Restriction du domaine de d Nous avons vu dans la section précédente que pour $n+k$ équations, $\delta(n) \sim \frac{n}{2}$. On peut donc se restreindre dans cette section à $d \leq \frac{n}{2}$, ce qui implique que $(2\alpha-1)n - (d+1) > 0$.

Il y a donc bien deux points cols, $z_0^{\pm} = \frac{1 \pm \sqrt{\Delta}}{2((2\alpha-1) - \frac{d+1}{n})}$ où $\Delta = 4\left(\frac{d+1}{n}\right)^2 + 4(1-2\alpha)\frac{d+1}{n} + 1$ est le discriminant du numérateur de l'équation de col. Il s'annule pour $\frac{d+1}{n} = \lambda_0^{\pm}$ avec $\lambda_0^{\pm} = \alpha - \frac{1}{2} \pm \sqrt{\alpha(\alpha-1)} > 0$. Lorsque $\frac{d+1}{n} \neq \lambda_0^{\pm}$, les deux cols sont simples et pour $\frac{d+1}{n} = \lambda_0^{\pm}$ il y a un point col double réel strictement positif, noté z_0.

Méthode du col Plaçons nous dans le cas ou les deux cols sont distincts. En appliquant la méthode du col on obtient :

$$\mathcal{I}_n(d) \sim \frac{F(z_0^+)}{\sqrt{2\pi n f''(z_0^+)}} + \frac{F(z_0^-)}{\sqrt{2\pi n f''(z_0^-)}}.$$

Or, si les deux cols sont réels l'un des deux termes domine et ne s'annule jamais (car $F(z)$ ne s'annule que pour $z = -1$, ou $z = 1$ et $\alpha n > n$). De même, si les deux cols sont complexes conjugués, $\mathcal{I}_n(d) \sim \Re\left(\frac{2F(z_0^+)}{\sqrt{2\pi n f''(z_0^+)}}\right)$ qui ne s'annule également jamais. Nous en déduisons donc que, au voisinage de la plus petite racine de $\mathcal{I}_n(d)$, il faut que le point col soit double, soit :

$$\frac{d+1}{n} \sim \lambda_0^- = \alpha - \frac{1}{2} - \sqrt{\alpha(\alpha-1)}$$

Cas général Dans le cas général, nous admettons que tout se passe comme dans le cas d'équations quadratiques. Plus précisément, nous admettons que :

- asymptotiquement, l'intégrale est dominée par les contributions de deux points cols z_0^+ et z_0^-,
- on approche de la racine $\delta(n)$ de l'intégrale $\mathcal{I}_n(d)$ lorsque ces deux points cols coalescent.

L'équation de col (4.21) donne $d + 1 \sim n\phi(z_0)$ où z_0 est la valeur du point col double. On détermine z_0 en dérivant l'équation de col : z_0 est la racine de $\phi'(z)$ pour laquelle $\phi(z_0)$ a la plus petite valeur positive.

Remarquons que $\lambda_0 = \phi(z_0)$ est aussi la plus petite racine strictement positive du discriminant en z du numérateur de $\phi(z) - \lambda$.

4.4.2 Termes suivants : méthode des cols coalescents

Posons $\lambda = \frac{d+1}{n}$. Pour obtenir les termes suivants du développement asymptotique de $\delta(n)$, nous utilisons la méthode des cols coalescents de Chester-Friedmann-Ursell [CFU57] pour calculer un développement asymptotique uniforme de $\mathcal{I}_n(d)$ au voisinage de $\lambda \sim \lambda_0$.

Changement de variable Nous appliquons le changement de variable cubique : $f(z) = \frac{u^3}{3} - \zeta u + \eta$ avec $\zeta^{\frac{3}{2}} = \frac{3}{4}(f(z_0^-) - f(z_0^+))$ et $\eta = \frac{1}{2}(f(z_0^-) + f(z_0^+))$. La méthode des cols coalescents conduit à un développement asymptotique de la forme de l'équation (4.10) :

$$\mathcal{I}_n(d) = e^{n\eta}\left[\frac{\mathrm{Ai}(n^{\frac{2}{3}}\zeta)}{n^{\frac{1}{3}}}\sum_{m\geq 0}\frac{B_m}{n^m} + \frac{\mathrm{Ai}'(n^{\frac{2}{3}}\zeta)}{n^{\frac{2}{3}}}\sum_{m\geq 0}\frac{C_m}{n^m}\right](1 + o(1))$$

les coefficients B_m et C_m s'exprimant en fonction des coefficients b_m et c_m. Le terme dominant est un multiple de $\mathrm{Ai}(n^{\frac{2}{3}}\zeta)$, qui s'annule donc pour $n^{\frac{2}{3}}\zeta \sim a_1$ un zéro de

la fonction d'Airy. Il faut donc déterminer un développement de ζ en fonction de $\lambda - \lambda_0$.

Calcul de ζ en fonction de $\lambda - \lambda_0$. Nous avons $\lambda - \lambda_0 = \phi(z_0^{\pm}) - \phi(z_0)$. Comme $\phi'(z_0) = 0$, un développement limité au voisinage de z_0 donne :

$$\lambda - \lambda_0 = \frac{\phi''(z_0)}{2}(z_0^{\pm} - z_0)^2 + o((z_0^{\pm} - z_0)^2)$$

Notons que, pour $\lambda > \lambda_0$, $z_0^{\pm} - z_0$ est imaginaire pur, et donc $\phi''(z_0) < 0$. En prenant la racine carrée de ce développement et en l'inversant, on obtient :

$$z_0^{\pm} - z_0 = \pm i\sqrt{\frac{2}{-\phi''(z_0)}}(\lambda - \lambda_0)^{\frac{1}{2}} + o((\lambda - \lambda_0)^{\frac{1}{2}}) \tag{4.22}$$

Rappelons que la fonction $f(z) = f(z, \lambda)$ dépend également de λ. Un développement de Taylor de f au voisinage de (z_0, λ_0) donne, en utilisant (4.22) :

$$f(z_0^{\pm}, \lambda) - f(z_0, \lambda) = \mp\frac{i}{z_0}\left(\frac{2}{-\phi''(z_0)}\right)^{\frac{1}{2}}(\lambda - \lambda_0)^{\frac{3}{2}} + o((\lambda - \lambda_0)^{\frac{3}{2}})$$

ce qui donne

$$\zeta^{\frac{3}{2}} = \frac{3}{4}(f(z_0^-) - f(z_0^+)) = \frac{i}{z_0}\left(\frac{2}{-\phi''(z_0)}\right)^{\frac{1}{2}}(\lambda - \lambda_0)^{3/2} + o((\lambda - \lambda_0)^{3/2})$$

d'où, comme $n^{\frac{2}{3}}\zeta \sim a_1$:

$$\lambda = \lambda_0 - \frac{a_1}{n^{2/3}}\left(\frac{-\phi''(z_0)z_0^2}{2}\right)^{1/3} + o(\frac{1}{n^{2/3}})$$

d'où le résultat du théorème 4.1.3 en utilisant que $d + 1 = \lambda n$.

Corollaire 4.4.1 *Pour αn équations quadratiques semi-régulières, on obtient :*

$$\delta(n) = (\alpha - \frac{1}{2} - \sqrt{\alpha(\alpha-1)})n + \frac{-a_1}{2(\alpha(\alpha-1))^{\frac{1}{6}}}n^{\frac{1}{3}} - \left(2 - \frac{2\alpha - 1}{4(\alpha(\alpha-1))^{\frac{1}{2}}}\right) + O(\frac{1}{n^{1/3}}) \tag{4.23}$$

La figure 4.3 donne les valeurs numériques des quatre premiers termes du développement asymptotique de la régularité pour αn équations quadratiques, $\alpha = 1, 2, 3, 4, 5$.

α	développement asymptotique de D_{reg}, en $O(\frac{1}{n^{2/3}})$
2	$D_{\text{reg}} = 0.085786\,n + 1.0415\,n^{1/3} - 1.4697 + 1.7130\,\frac{1}{n^{1/3}}$
3	$D_{\text{reg}} = 0.050510\,n + 0.86725\,n^{1/3} - 1.4897 + 1.9958\,\frac{1}{n^{1/3}}$
4	$D_{\text{reg}} = 0.035898\,n + 0.77263\,n^{1/3} - 1.4948 + 2.2230\,\frac{1}{n^{1/3}}$
5	$D_{\text{reg}} = 0.027864\,n + 0.70957\,n^{1/3} - 1.4969 + 2.4131\,\frac{1}{n^{1/3}}$

FIG. 4.3 – Développement asymptotique de la régularité pour αn équations quadratiques semi-régulières

4.4.3 Suite semi-régulière de αn équations sur $\mathbb{F}_2$

L'analyse asymptotique s'applique dans ce cas exactement comme dans le cas de αn équations formant une suite semi-régulière générale, il suffit de changer la valeur de la fonction ϕ. On obtient :

Théorème 4.4.2 *La régularité d'une suite semi-régulière sur $\mathbb{F}_2$ de αn équations de degrés $d_1, \ldots, d_{\alpha n}$ en n variables, $\alpha > 1$ constant, se comporte asymptotiquement comme*

$$D_{reg} = \phi(z_0)n - a_1 \left(-\frac{1}{2}\phi''(z_0)z_0^2\right)^{\frac{1}{3}} n^{\frac{1}{3}}(1 + o(1)) \tag{4.24}$$

lorsque $n \to \infty$, où

$$\phi(z) = \frac{z}{1+z} - \frac{1}{n}\sum_{i=1}^{\alpha n} \frac{d_i z^{d_i}}{1 + z^{d_i}},$$

z_0 est la racine de $\phi'(z)$ qui minimise $\phi(z_0) > 0$, et a_1 est la plus grande racine de la fonction Ai *d'Airy.*

Corollaire 4.4.3 *Pour αn équations quadratiques semi-régulières sur $\mathbb{F}_2$, on a :*

$$\delta(n) \sim \left(-\alpha + \frac{1}{2} + \frac{1}{2}\sqrt{2\alpha^2 - 10\alpha - 1 + 2(\alpha+2)\sqrt{\alpha(\alpha+2)}}\right) n \tag{4.25}$$

La figure 4.4 donne les valeurs numériques des trois premiers termes du développement asymptotique de la régularité pour n équations semi-régulières sur $\mathbb{F}_2$ de degré $D \in [2; 7]$.

D	développement asymptotique de D_{reg}, en $O(\frac{1}{n^{1/3}})$
2	$D_{\text{reg}} = 0.09n + 1.00n^{\frac{1}{3}} - 1.58$
3	$D_{\text{reg}} = 0.15n + 1.35n^{\frac{1}{3}} - 1.42$
4	$D_{\text{reg}} = 0.20n + 1.60n^{\frac{1}{3}} - 1.27$
5	$D_{\text{reg}} = 0.24n + 1.79n^{\frac{1}{3}} - 1.11$
6	$D_{\text{reg}} = 0.26n + 1.95n^{\frac{1}{3}} - 0.94$
7	$D_{\text{reg}} = 0.28n + 2.09n^{\frac{1}{3}} - 0.78$

FIG. 4.4 – Développement asymptotique de la régularité pour $m = n$ équations de degré D, semi-régulières sur $\mathbb{F}_2$

Chapitre 5

Applications en cryptographie

Dans ce chapitre, nous appliquons les résultats obtenus pour les suites semi-régulières (Chapitres 3 et 4) à l'analyse de problèmes issus de la cryptographie. Nous explicitons les particularités des systèmes à coefficients et solutions dans $\mathbb{F}_2$, et montrons comment la connaissance du comportement des systèmes semi-réguliers sur $\mathbb{F}_2$ peut apporter des informations sur des systèmes provenant de systèmes de chiffrement. Nous illustrons cela sur le système de chiffrement à clef publique HFE et sur des systèmes de chiffrement symétriques.

5.1 Introduction

Depuis les années 80, sont apparus en cryptographie de nombreux systèmes basés sur le problème de la résolution d'un système algébrique à coefficients dans un corps fini [MI88, Pat96b, Moh99, Kob98]. Dans de nombreux cas, ces systèmes sont à coefficients dans $\mathbb{F}_2$, et les seules solutions intéressantes sont les solutions dans $\mathbb{F}_2$ et non dans sa clôture algébrique. Une possibilité pour trouver ces solutions dans le corps de base est d'ajouter au système original des équations de corps $x_1^2 + x_1, \ldots, x_n^2 + x_n$, si les variables sont $x_1, \ldots, x_n$.

Pour trouver les solutions dans une extension $\mathbb{F}_{2^l} \supset \mathbb{F}_2$ sans ajouter de variable supplémentaire, on peut ajouter les équations de corps $x_i^{2^l} + x_i$. Le système se comporte alors vis-à-vis du calcul de base de Gröbner comme un système sans équations de corps jusqu'au degré 2^l. Si l est suffisamment grand, ce degré n'est pas atteint et il est inutile (voire gênant) d'ajouter les équations de corps au système.

Pour certains cryptosystèmes à clef publique, comme [MI88, Pat96b, Moh99], la clef publique est un ensemble de polynômes algébriques

$$\begin{cases} p_1(x_1, \ldots, x_n), \\ \vdots \\ p_m(x_1, \ldots, x_n) \end{cases}$$

à coefficients dans $\mathbb{F}_2$. Pour chiffrer un message $\mathbf{a} = (a_1, \ldots, a_n)$, l'utilisateur doit évaluer les m polynômes de la clef publique $(y_1, \ldots, y_n) = (p_1(\mathbf{a}), \ldots, p_m(\mathbf{a}))$ modulo 2. Pour déchiffrer le message $(y_1, \ldots, y_m)$ sans connaître la clef secrète, une solution est de résoudre le système de m équations $y_1 - p_1(x_1, \ldots, x_n), \ldots, y_m - p_m(x_1, \ldots, x_n)$ en n variables $(x_1, \ldots, x_n)$ avec les équations de corps.

Une autre méthode générale de cryptanalyse, appelée cryptanalyse algébrique, a fait son apparition plus récemment. Un grand nombre d'articles [BDC03, CM03, AK03, MR02, ...] exploitent la structure algébrique de certains cryptosystèmes (dont l'AES) pour construire un système algébrique reliant les bits d'entrée de l'algorithme de chiffrement aux bits de sortie.

Le but de ce chapitre est de montrer l'intérêt de la connaissance de la complexité de résolution par bases de Gröbner pour des systèmes semi-réguliers sur $\mathbb{F}_2$ (que nous abrégerons en "semi-réguliers" dans ce chapitre lorsqu'il n'y a pas d'ambiguïté) pour l'étude de cryptosystèmes algébriques ou pour l'analyse de la complexité d'une cryptanalyse algébrique.

Nous expliquons Section 5.2 pourquoi les bornes de complexité obtenues pour les systèmes semi-réguliers peuvent servir de bornes pour des systèmes particuliers. Nous verrons dans l'exemple du cryptosystème HFE que ces bornes peuvent être très mauvaises, la complexité réelle des systèmes étant bien meilleure que celle des systèmes semi-réguliers de mêmes paramètres. Dans ce cas il est nécessaire de mener une étude plus poussée pour obtenir une bonne estimation de complexité. Nous définissons pour cela la notion de d-régularité d'un système (Section 5.2.3), qui permet d'estimer précisément la complexité de résolution.

Un schéma d'analyse possible de cryptosystèmes algébriques (ou d'une cryptanalyse algébrique) est le suivant :

1. Éventuellement, remise en équation du problème,
2. Comparaison expérimentale des systèmes obtenus avec des systèmes aléatoires (pour des petites tailles),
3. Comparaison théorique avec des systèmes semi-réguliers, recherche de la d-régularité du système,
4. Analyse de complexité globale.

Organisation du chapitre Nous récapitulons Section 5.2 les résultats de complexité pour des systèmes à coefficients et solutions dans $\mathbb{F}_2$. Nous appliquons ces résultats à l'analyse du cryptosystème HFE (Section 5.3) ainsi qu'à l'analyse de cryptanalyses algébriques sur des systèmes symétriques comme l'AES (Section 5.4).

5.2 Complexité de résolution de systèmes dans $\mathbb{F}_2$

Dans tous ce chapitre nous étudions essentiellement des systèmes d'équations à coefficients dans $\mathbb{F}_2$ dont on recherche les solutions dans ce même corps fini.

Nous ajoutons donc les équations de corps $x_1^2 + x_1, \ldots, x_n^2 + x_n$, et nous notons $R_n = \mathbb{F}_2[x_1, \ldots, x_n]/(x_1^2 + x_1, \ldots, x_n^2 + x_n)$ l'anneau des fonctions booléennes en n variables. Soit $f_1, \ldots, f_m \subset R_n$ un système de m équations en n variables de degrés $(d_1, \ldots, d_m)$.

Du fait de la présence des équations de corps, l'idéal $I = \langle f_1, \ldots, f_m \rangle \subset R_n$ vérifie de bonnes propriétés. Il possède un nombre fini de solutions, même après homogénéisation. Il est radical (i.e. toutes ses solutions sont de multiplicité 1), et si G est une base de Gröbner réduite de I alors $G = \{1\}$ si et seulement si le système n'a pas de solution, et $G = \{x_1 - a_1, \ldots, x_n - a_n\}$ si et seulement si le système à une unique solution $(a_1, \ldots, a_n)$. Dans ces deux cas, calculer la base de Gröbner équivaut à résoudre le système.

5.2.1 Complexité dans le cas le pire et génériquement

Le problème de résoudre un système d'équations algébriques dans $\mathbb{F}_2$ est NP-complet [FY80, FY79]. Pour ce qui est du calcul d'une base de Gröbner, nous avons vu chapitre 1 que le cas le pire est simplement exponentiel : le *degré de régularité* (voir Définition 3.5.1, c'est une borne sur le degré maximal d'un polynôme apparaissant dans un calcul de base de Gröbner) est $D_{\text{reg}} \leq n$.

Il arrive qu'un problème, bien que NP-complet, soit génériquement facile : il existe des instances dures du problème (puisqu'il est NP-complet), mais toute instance tirée au hasard se résout en temps polynômial (c'est par exemple le cas du problème 3-SAT). De tels problèmes n'ont pas d'intérêt en pratique du point de vue de la cryptographie.

Le problème de résoudre un système algébrique semble lui être un problème génériquement difficile. Nous conjecturons en effet à la manière de Fröberg (voir Conjecture 1.6.3 page 22) que les suites semi-régulières sont génériques, i.e. que lorsque $n \to \infty$, la proportion de suites semi-régulières en n variables et m équations de degrés $d_1, \ldots, d_m$ tend vers 1 (le nombre d'équations m pouvant dépendre de n). Cela signifie que "presque toute suite" est semi-régulière. Or, nous avons vu Chapitre 4 que, pour des suites semi-régulières, on peut trouver un équivalent très précis du degré de régularité D_{reg} du système. En admettant notre conjecture de généricité des suites semi-régulières, les bornes pour ces suites donnent des estimations de complexité en moyenne, et peuvent être utilisées comme bornes de complexité pour des systèmes issus de problèmes cryptographiques.

Ainsi, pour αn équations quadratiques en n variables (α constant) ce degré maximal équivaut à

$$D_{\text{reg}} \sim \left(-\alpha + \frac{1}{2} + \frac{1}{2}\sqrt{2\alpha^2 - 10\alpha - 1 + 2(\alpha+2)\sqrt{\alpha(\alpha+2)}} \right) n$$

lorsque n tend vers l'infini.

Remarquons qu'il existe un exemple (celui de Mayr-Meyer) de suite (sans équations de corps) ayant une complexité pire que celle des suites semi-régulières, mais

cette question reste ouverte pour les systèmes à coefficients dans $\mathbb{F}_2$ avec équations de corps.

5.2.2 Régularité "linéaire" d'un système

Pour des suites semi-régulières le degré de régularité est le premier degré où apparaît une chute de degré dans le calcul de la base de Gröbner avec l'algorithme F5-matriciel. Or, les systèmes fortement surdéterminés ont en général peu de solutions (au moins une pour des systèmes provenant d'applications, ils ont été construits pour), et la base de Gröbner contient de nombreux polynômes linéaires. Il est intéressant de regarder le premier degré D_{lin} pour lequel apparaissent ces polynômes *linéaires* dans la base de Gröbner.

Lemme 5.2.1 *En supposant qu'il n'y a pas de réduction à zéro dans l'algorithme F5-matriciel avant le degré D_{lin} (mais il peut y avoir des chutes de degré), alors D_{lin} est l'exposant du premier coefficient négatif ou nul de la série :*

$$\frac{(1+z)^n}{1-z}\frac{1}{\prod_{i=1}^{m}(1+z^{d_i})} - \frac{1+nz}{1-z}$$

Démonstration D_{lin} est le premier degré pour lequel, dans l'algorithme F5-matriciel en version affine, le nombre de lignes de degré $\leq d$ est supérieur au nombre de monômes de degré compris entre 2 et d. Le nombre de monômes est donné par $[z^d]\frac{(1+z)^n-1-nz}{1-z}$, et le nombre de lignes par $[z^d]\frac{(1+z)^n}{1-z}(1-\frac{1}{\prod_{i=1}^{m}(1+z^{d_i})})$. □

Nous avons tracé sur la figure 5.1 les degrés de régularité et de linéarité pour des équations quadratiques sur $\mathbb{F}_2$. Nous voyons que, pour de petites valeurs de n, il suffit d'aller au degré juste supérieur au degré de régularité pour voir apparaître des relations linéaires (et souvent elles apparaissent déjà au degré de régularité). Plus généralement, nous avons le lemme suivant :

Lemme 5.2.2 *En supposant qu'il n'y a pas de réduction à zéro dans l'algorithme F5-matriciel avant le degré D_{lin}, alors $D_{lin} \sim D_{reg}$*

Démonstration Nous utilisons les méthodes d'analyse asymptotique décrites Chapitre 4. Le degré de régularité D_{reg} est la plus petite racine de l'intégrale

$$\mathcal{I}_n(d) = \frac{1}{2i\pi}\oint \frac{(1+z)^n}{\prod_{i=1}^{m}(1+z^{d_i})}\frac{1}{z^{d+1}}dz = \frac{1}{2i\pi}\oint e^{nf(z)}dz.$$

On l'obtient en calculant un équivalent asymptotique de $\mathcal{I}_n(d)$. Notons z_0^+ et z_0^- les deux cols qui contribuent pour l'essentiel à l'intégrale asymptotiquement, alors tant que $z_0^+ \neq z_0^-$ nous obtenons

$$\mathcal{I}_n(d) \sim \frac{e^{nf(z_0^+)}}{\sqrt{2\pi n f''(z_0^+)}} + \frac{e^{nf(z_0^+)}}{\sqrt{2\pi n f''(z_0^-)}}.$$

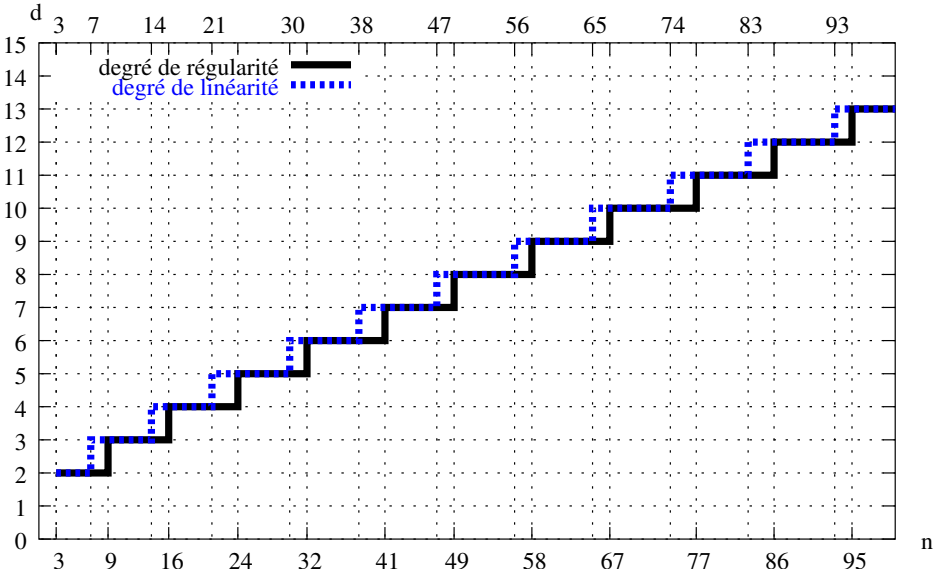


FIG. 5.1 – Comparaison des degrés de régularité et de linéarité, équations quadratiques

Nous montrons que ce terme ne peut pas s'annuler, ce qui implique que l'équivalent asymptotique de D_{reg} est obtenu lorsque les deux cols z_0^+ et z_0^- coalescent.

Le degré de linéarité D_{lin} est la valeur de d telle que l'intégrale définie par

$$\mathcal{I}_n(d) = \frac{1}{2i\pi}\oint \frac{1}{1-z}\frac{(1+z)^n}{\prod_{i=1}^{m}(1+z^{d_i})}\frac{1}{z^{d+1}}dz = \frac{1}{2i\pi}\oint \frac{1}{1-z}e^{nf(z)}dz$$

vérifie $\mathcal{I}_n(d) = n+1$. Les cols contribuant essentiellement à l'intégrale sont les mêmes que précédemment. A nouveau, tant que les deux cols ne coalescent pas, un équivalent asymptotique de $\mathcal{I}_n(d)$ est

$$\mathcal{I}_n(d) \sim \frac{e^{nf(z_0^+)}}{(1-z_0^+)\sqrt{2\pi n f''(z_0^+)}} + \frac{e^{nf(z_0^+)}}{(1-z_0^-)\sqrt{2\pi n f''(z_0^-)}}.$$

Or, ce terme a un comportement exponentiel en n, et $\mathcal{I}_n(d)$ ne peut se comporter comme un polynôme en n que si les deux cols z_0^+ et z_0^- coalescent à nouveau, d'où nous déduisons que D_{reg} et D_{lin} sont asymptotiquement équivalents à la plus petite valeur strictement positive de d pour laquelle $z_0^+ = z_0^-$. □

5.2.3 d-Régularité d'un système

Nous verrons Section 5.3 que certains systèmes, s'ils ne sont pas semi-réguliers, se comportent "presque" comme un système semi-régulier c'est-à-dire que le calcul se termine lorsque les premières chutes de degré apparaissent dans le déroulement de l'algorithme F5-matriciel. Nous pouvons donner la définition suivante :

Définition 5.2.3 *Une suite homogène* $f_1, \ldots, f_m \subset \mathbb{F}_2[x_1, \ldots, x_n]/(x_1^2, \ldots, x_n^2) = R_n^h$ *est* semi-régulière jusqu'à l'ordre d sur $\mathbb{F}_2$ *si :*

- $I = \langle f_1, \ldots, f_m \rangle \neq R_n^h$,
- *pour* $i \in [1; m]$, *si* $g_i f_i = 0$ *dans* $R_n^h/(f_1, \ldots, f_{i-1})$ *et* $\deg(g_i f_i) < d$ *alors* $g_i = 0$ *dans* $R_n^h/(f_1, \ldots, f_{i-1}, f_i)$.

Nous pouvons donner de la même manière la définition de suite semi-régulière générale jusqu'à l'ordre d. Notons qu'une suite semi-régulière est semi-régulière jusqu'à l'ordre D_{reg}.

Étant donnée une suite semi-régulière jusqu'à l'ordre d, le nombre de lignes et de colonnes des matrices de l'algorithme F5-matriciel jusqu'au degré d sont bien déterminées, et ces matrices sont de rang plein (sauf peut-être celle en degré d). Si, au degré d, les chutes de degré sont suffisamment fortes pour que le calcul se termine (par exemple on obtient des équations linéaires), alors la complexité du calcul de la base de Gröbner est essentiellement celle de la mise sous forme Échelon de cette plus grande matrice. Un tel système sera appelé un système d-régulier.

5.3 Analyse du cryptosystème HFE

Le système HFE (Hidden Field Equations) est décrit dans [Pat96b]. Ce cryptosystème généralise et améliore celui de Matsumoto et Imai [MI88] cassé dans [Pat95]. Il peut être utilisé en chiffrement ou en signature, et est très étudié car il fournit des signatures très courtes (le schéma Quartz basé sur HFE propose des signatures de 128 bits).

Longtemps considérée comme sûre, la version basique du système HFE, décrite Section 5.3.1, a été cassée par Jean-Charles Faugère [Fau03] en calculant une base de Gröbner du système $y_1 = p_1(x_1, \ldots, x_n), \ldots, y_n = p_n(x_1, \ldots, x_n)$ avec équations de corps, où $(p_1, \ldots, p_n)$ est la clef publique, $y_1, \ldots, y_n$ le message chiffré et $x_1, \ldots, x_n$ le message clair à retrouver. Le Challenge [Pat96a] portait sur un système HFE-basic de 80 équations en 80 variables. La plus grande matrice à résoudre était de taille 307 126 x 1 667 009, et le temps de calcul global d'environ 2 jours et 4 heures sur une station de travail HP avec processeur alpha EV68 à 1000 Mhz et 4 Go de mémoire (le process utilisant une taille mémoire de 7.65 Go).

Dans une publication commune avec Antoine Joux [FJ03], les auteurs analysent la complexité de cette attaque, en calculant la d-régularité de ces systèmes.

5.3.1 Description de HFE-basic

Nous n'étudions pas ici toutes les variantes possibles des systèmes HFE, dont la sécurité n'est pas encore bien comprise. Nous décrivons la version basique de HFE.

Soit q une puissance d'un nombre premier, et n un entier. Soit

$$f(x) = \sum_{q^{\theta_{i,j}} + q^{\phi_{i,j}} \leq d} \beta_{i,j} x^{q^{\theta_{i,j}} + q^{\phi_{i,j}}} + \sum_{q^{\epsilon_k} \leq d} \alpha_k x^{q^{\epsilon_k}}$$

une fonction quadratique de $\mathbb{F}_{q^n}[x]$ de degré d, avec $\beta_{i,j}, \alpha_k \in \mathbb{F}_{q^n}$, $\theta_{i,j}, \phi_{i,j}, \epsilon_k \in \mathbb{N}$. Comme $\mathbb{F}_{q^n} \simeq (\mathbb{F}_q)^n$, les éléments de $\mathbb{F}_{q^n}$ sont représentables par des n-uplets d'éléments de $\mathbb{F}_q$, et comme

$$\mathbb{F}_{q^n}[x]/(x^{q^n} - x) \simeq (\mathbb{F}_q[x_1, \ldots, x_n]/(x_1^q - x_1, \ldots, x_n^q - x_n))^n$$

alors $f(x)$ est représentable (voir annexe A.3) par un n-uplet de fonctions polynômes en n variables sur $\mathbb{F}_q$,

$$f(x_1, \ldots, x_n) = (f_1(x_1, \ldots, x_n), \ldots, f_n(x_1, \ldots, x_n))$$

La fonction f a été choisie de poids de Hamming 2 (chaque exposant d'un monôme de f écrit en base q est de poids de Hamming au plus 2), or la fonction $x \mapsto x^q$ est linéaire sur $\mathbb{F}_{q^n}$ donc les polynômes f_i sont de degré au plus 2 en $(x_1, \ldots, x_n)$.
Convention. Si $x_1, \ldots, x_n$ sont des éléments de $\mathbb{F}_q$, nous noterons $\mathbf{x} = (x_1, \ldots, x_n)$ l'élément de ${\mathbb{F}_q}^n$ et $x = \sum_{i=1}^n x_i w^{i-1} \in \mathbb{F}_{q^m}$ où w est un générateur de $\mathbb{F}_{q^n}$ sur $\mathbb{F}_q$.

Soit s et t deux bijections affines de $(\mathbb{F}_q)^n \to (\mathbb{F}_q)^n$, à nouveau s et t peuvent être représentées soit comme des fonctions linéaires bijectives de $\mathbb{F}_{q^n}[x]/(x^{q^n} - x)$ (chaque monôme est de la forme x^{q^i}), soit comme des n-uplets de polynômes linéaires en $(x_1, \ldots, x_n)$ sur $\mathbb{F}_q$ linéairement indépendants. Notons que du point de vue de la mise en équations algébriques, les bijections s et t ne changent pas la complexité du système (les degrés de régularité ou de linéarité sont les mêmes). Ces bijections sont nécessaires pour protéger le systèmes vis-à-vis d'autres attaques que les attaques algébriques.

Nous pouvons alors décrire une version basique du cryptosystème HFE :

Clef secrète : les fonctions f, s, t,

Clef publique : une représentation de $\mathbb{F}_{q^n}$ sur $\mathbb{F}_q$, et le n-uplet de fonctions quadratiques en $x_1, \ldots, x_n$ représentant $t(f(s(x_1, \ldots, x_n)))$, noté

$$\mathbf{p}(\mathbf{x}) = (p_1(x_1, \ldots, x_n), \ldots, p_n(x_1, \ldots, x_n)) \in (\mathbb{F}_q[x_1, \ldots, x_n])^n,$$

Chiffrement : pour chiffrer un n-uplet $\mathbf{x} = (x_1, \ldots, x_n)$, calculer

$$\mathbf{y} = (p_1(x_1, \ldots, x_n), \ldots, p_n(x_1, \ldots, x_n)),$$

Déchiffrement : pour déchiffrer un message $\mathbf{y}$, calculer les solutions z du polynôme univarié $f(z) = t^{-1}(y)$ puis calculer $x = s^{-1}(z)$,

Le calcul des racines d'un polynôme univarié de degré d sur un corps fini $\mathbb{F}_{q^n}$ peut se faire en $O(M(d)\log(d))$ opérations dans le corps fini, avec $M(d)$ le coût de la multiplication de deux polynômes de degré d. Il faut donc que d soit assez petit pour pouvoir déchiffrer les messages en temps raisonnable ($d \leq 512$, voir [FJ03]). Nous noterons HFE(d) un système HFE dont le polynôme secret est de degré d.

Pour pouvoir analyser le comportement des systèmes HFE, et en particulier les comparer aux systèmes aléatoires, nous avons implanté en Magma la génération d'une clef publique HFE. Les calculs de bases de Gröbner ont été effectués par Jean-Charles Faugère avec l'algorithme F5/2 implantant l'algorithme F5-matriciel/2 décrit section 1.5.2.

5.3.2 Un distingueur pour les systèmes HFE

L'auteur du système HFE affirme qu'un système HFE est indistinguable d'un système aléatoire. Si l'on calcule la proportion de coefficients non nuls dans un système HFE et dans un système aléatoire, on obtient dans les deux cas environ 1/2.

Essayons de calculer une base de Gröbner pour des systèmes HFE et pour des systèmes aléatoires. En un temps raisonnable, nous pouvons construire expérimentalement le graphe de la figure 5.2, qui compare le degré maximal atteint lors d'un calcul de base de Gröbner avec l'algorithme F5-matriciel pour des systèmes aléatoires et des systèmes HFE. A ce stade, pour $n \leq 30$ les systèmes HFE($16 < d < 129$)

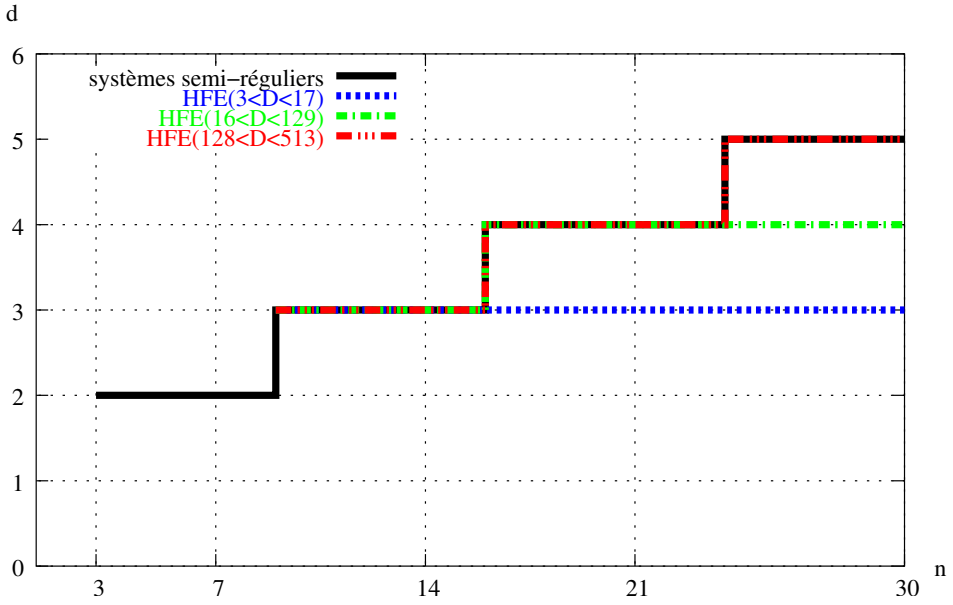


FIG. 5.2 – Comparaison des systèmes HFE et des systèmes aléatoires

semblent se comporter comme des systèmes aléatoires, mais il est difficile de poursuivre l'expérience car la complexité de résolution des systèmes devient très grande.

Or, la courbe correspondant aux systèmes aléatoires est théoriquement connue : c'est celle obtenue pour les systèmes semi-réguliers. Nous constatons que les calculs pour les systèmes semi-réguliers montent en degré 6 à partir de $n = 32$ variables. Or, dans ce cas les systèmes HFE($16 < d < 129$) continuent à se résoudre en degré au plus 5.

En utilisant la courbe théorique pour les systèmes aléatoires, nous pouvons obtenir la courbe de la figure 5.3 (fournie par Jean-Charles Faugère [Fau03]). L'algorithme décrit Figure 5.4 page 108 est un distingueur pour les systèmes HFE.

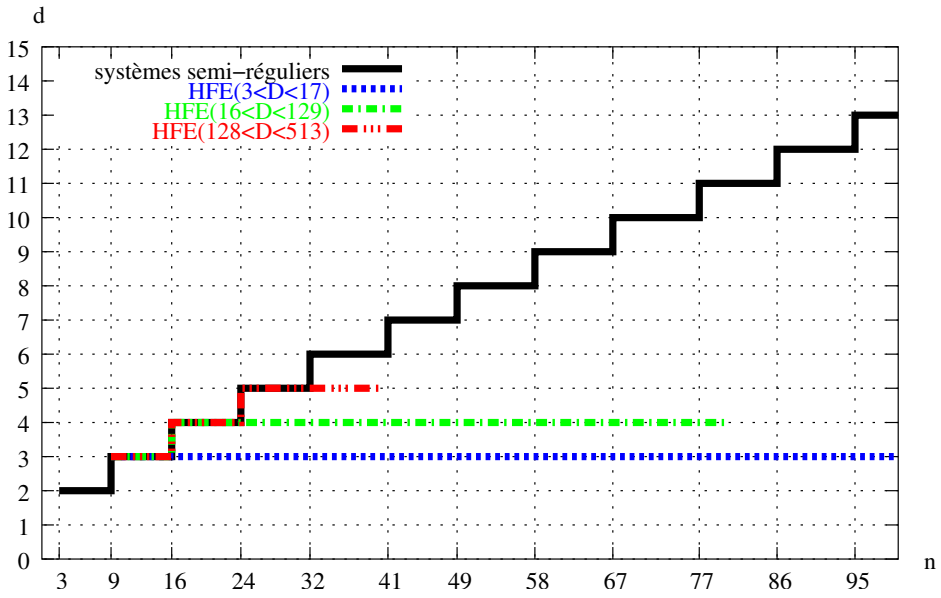


FIG. 5.3 – d-Régularité des systèmes HFE

5.3.3 Analyse de HFE connaissant sa d-régularité

Pour résoudre un système algébrique avec les bases de Gröbner, il suffit de mettre sous forme échelon une matrice de type Macaulay (Section 1.4) pour un degré suffisamment élevé. Dans le cas d'un système HFE d-régulier, pour obtenir des équations linéaires il suffit de se placer en degré d. Les valeurs de d en fonction du degré du polynôme secret sont données dans [FJ03] et reproduites table 5.1.

HFE(D)	$3 \leq D \leq 16$	$17 \leq D \leq 128$	$129 \leq D \leq 512$	$513 \leq D \leq 1280$
d-régularité	3	4	5	6

TAB. 5.1 – d-régularité des systèmes HFE

Pour l'algorithme F5-matriciel, la matrice en degré d possède $\sum_{k=0}^{d} \binom{n}{k} \sim n^d$ colonnes, et son nombre de lignes en degré d est donné par le $d$ème coefficient de la série génératrice

$$\begin{aligned}(1+z)^n \left(1 - \frac{1}{(1+z^2)^n}\right) &= nz^2 + n^2 z^3 + \frac{1}{2} n \left(n^2 - 2\,n - 1\right) z^4 \\ &\quad + \frac{1}{6} n^2 \left(n^2 - 6\,n - 1\right) z^5 \\ &\quad + \frac{1}{24} n \left(n^4 - 12\,n^3 + 15\,n^2 + 12\,n + 8\right) z^6 + O\left(z^7\right)\end{aligned}$$

Nous en déduisons le tableau suivant donnant la taille de la matrice à résoudre :

Entrée : $\begin{cases} F = (f_1, \ldots, f_n) \text{ un système d'équations quadratiques,} \\ D \text{ un entier} \end{cases}$

Sortie : $\begin{cases} \text{oui} \quad \text{si } F \text{ provient d'un système HFE}(D) \\ \text{non} \quad \text{sinon} \end{cases}$

Calculer d la d-régularité d'un système HFE(D) (en utilisant la table 5.1),
Exécuter l'algorithme F5-matriciel jusqu'au degré d,
Si en degré d on observe de nombreuses chutes de degré,
alors Retourner oui
sinon Retourner non.

FIG. 5.4 – Un distingueur pour les systèmes HFE

HFE(D)	$3 \leq D \leq 16$	$17 \leq D \leq 128$	$129 \leq D \leq 512$	$513 \leq D \leq 1280$
Nb. lignes	$\sim n^2$	$\sim \frac{1}{2}n^3$	$\sim \frac{1}{6}n^4$	$\sim \frac{1}{24}n^5$
Nb. colonnes	$\sim n^3$	$\sim n^4$	$\sim n^5$	$\sim n^6$

Chacune de ces matrices est très creuse : elle possède $\frac{1}{2}n^2$ termes non nuls par ligne.

Dans [FJ03], Faugère et Joux suggèrent l'utilisation d'un algorithme du type Wiedemann ou Lanczos par blocs [Cop94, Mon95]. Ils obtiennent une attaque probabiliste, de complexité globale de résolution de l'ordre de $O(n^2N^2)$ où N est le nombre de lignes de la matrice, ce qui donne le tableau suivant :

HFE(D)	$3 \leq D \leq 16$	$17 \leq D \leq 128$	$129 \leq D \leq 512$	$513 \leq D \leq 1280$	
d-régularité	3	4	5	6	d
Complexité	$O(n^6)$	$O(n^8)$	$O(n^{10})$	$O(n^{12})$	$O(n^{2d})$

La complexité de résolution d'un système HFE d-régulier est donc polynomiale en n, à d fixé.

5.4 Cryptosystèmes symétriques

Dans de nombreux articles, les auteurs dérivent pour des cryptosystèmes symétriques des ensembles d'équations vérifiées par les bits de messages clairs et chiffrés. Le problème est d'estimer la complexité de résolution de tels systèmes. En appliquant les résultats que nous avons obtenus pour les systèmes semi-réguliers, nous obtenons des bornes supérieures "a priori" de complexité pour leurs attaques.

Ainsi, pour le cryptosystème AES, il a été montré que l'on peut retrouver la clef secrète à partir d'un couple clair-chiffré à condition de pouvoir résoudre un système de 6000 équations quadratiques en 1600 variables sur $\mathbb{F}_2$ [CP02], ou en résolvant un autre système (creux) de 3840 équations quadratiques en 2578 variables sur $\mathbb{F}_{2^8}$.

Dans le premier cas, si la suite était semi-régulière le degré de régularité serait $D_{\text{reg}} = 56$, il faudrait donc résoudre un système linéaire de taille $\binom{1601}{56} \sim 2^{346}$. Dans le second cas, toujours en supposant la suite semi-régulière, le degré de régularité vaudrait $D_{\text{reg}} = 367$, et la taille de la plus grande matrice serait $\binom{2579}{367} \sim 2^{1517}$. Dans les deux cas, on est largement au-delà de la recherche exhaustive.

Dans [BDC03], les auteurs donnent pour divers cryptosystèmes des ensembles d'équations algébriques sur $\mathbb{F}_2$ vérifiées par les bits d'entrée et de sortie de l'algorithme de chiffrement, en essayant dans la mesure du possible d'obtenir des systèmes creux. Plus précisément, les équations sont choisies de sorte à "minimiser le nombre de termes libres", i.e. le nombre de monômes libres lorsque l'on considère le système comme un système linéaire en ses monômes, et lorsque cela est possible de sorte à "minimiser le nombre de termes de chaque polynôme".

La table 5.2 est recopiée de [BDC03] : elle décrit le nombre d'équations (linéaires ou quadratiques) trouvées reliant les bits d'entrée et de sortie de différents cryptosystèmes, ainsi que le nombre de variables. A partir de cette table nous pouvons

Cryptosystème	Variables	Éq. linéaires	Éq. quadratiques
Khazad	6464	1664	6000
Misty1	3856	2008	1848
Kasumi	4264	2264	2000
Camellia-128	3584	1920	4304
Rijndael-128	3296	1696	4600
Serpent-128	16640	8320	9360

TAB. 5.2 – Équations reliant les bits d'entrée/sortie de cryptosystèmes symétriques

construire la table 5.3. Nous ne considérons pas les équations linéaires (nous retirons autant d'équations linéaires que de variables). Nous pouvons pour chaque système donner le degré maximal D_{reg} d'un système semi-régulier de mêmes paramètres, et la taille de la plus grande matrice à résoudre. Il paraît clairement

Cryptosystème	Variables	Éq. quad.	D_{reg}	Taille matrice
Khazad	4800	6000	379	2^{2076}
Misty1	1848	1848	179	2^{1040}
Kasumi	2000	2000	193	2^{1129}
Camellia-128	1664	4304	78	2^{538}
Rijndael-128	1600	4600	69	2^{479}
Serpent-128	8320	9360	703	2^{4196}

TAB. 5.3 – Bornes de complexité pour la cryptanalyse algébrique

impossible de résoudre des systèmes génériques de cette taille. Cela ne signifie nullement qu'une attaque algébrique sur ces systèmes ne marchera pas, ni que ces

attaques ne marchent pas, mais que pour montrer la pertinence de ces attaques il faudrait montrer, comme dans le cas des systèmes HFE, que les systèmes construits sont beaucoup plus simples à résoudre que des systèmes semi-régulières. Enfin, les auteurs de [BDC03] se restreignent à des systèmes non surdéterminés, or notre étude montre que les systèmes surdéterminés sont *en général* plus faciles à résoudre.

Notons que les auteurs de ces articles cherchent à obtenir des systèmes le plus *creux* possible. Pour pouvoir exploiter cette propriété, il faudrait pouvoir quantifier la complexité de résolution de systèmes "creux". Un système ne contenant qu'un unique monôme a clairement une complexité de résolution polynomiale en le nombre de variables, et un système dense a une complexité de résolution exponentielle en le nombre de variables. Les systèmes creux se situent entre les deux, mais comment délimiter les bornes ?

Chapitre 6

Nouveaux algorithmes de décodage des codes cycliques

Dans ce chapitre, nous nous intéressons au problème du décodage des codes cycliques, qui peut se réécrire sous forme d'un problème algébrique, et se résoudre par un calcul de base de Gröbner. Les (nombreux) systèmes étudiés précédemment comportent tous des équations dites équations de corps, qui permettent de calculer les solutions du système dans un corps fini plutôt que dans sa clôture algébrique. Notre travail a porté sur la possibilité de conserver les bonnes propriétés de décodage de ces systèmes, tout en supprimant les équations de corps, qui sont de degré très élevé et rendent très vite impossible le calcul de la base de Gröbner.

Notre étude nous a permis de donner une classification précise de tous les systèmes de dimension zéro. Nous proposons de nouveaux systèmes, de dimension positive, qui possèdent les mêmes propriétés de décodage mais dont la résolution par base de Gröbner, étonnamment, est bien plus efficace. Nous donnons de nombreux exemples de décodages de codes cycliques, en particulier pour les codes à résidus quadratiques, pour lesquels aucun algorithme de décodage n'existait auparavant.

6.1 Introduction

Nous invitons le lecteur à se reporter au chapitre 2 pour une introduction aux codes correcteurs d'erreurs, ainsi qu'une présentation du principe de décodage algébrique par bases de Gröbner. Le principe du décodage algébrique est de trouver un système algébrique dont les solutions donnent l'erreur qui s'est produite, et pour lequel la résolution par bases de Gröbner soit effective. Tous les systèmes étudiés précédemment comportent des équations de corps de très haut degré, qui permettent de ne chercher que les solutions appartenant à ce corps fini, mais qui

compliquent la résolution par bases de Gröbner.

Nos contributions au problème de décodage des codes à résidus quadratiques binaires, et plus généralement des codes cycliques binaires, sont les suivantes :

Nouveaux systèmes algébriques et algorithmes de décodage. Nous introduisons de nouveaux systèmes algébriques, de *dimension positive*. Pour ces systèmes, les théorèmes de structure des idéaux de dimension zéro, qui conduisent aux algorithmes de décodage précédents, ne sont plus valables, mais nous avons prouvé que ces idéaux possèdent les mêmes propriétés théoriques de décodage que les systèmes de dimension zéro, et conduisent de la même manière à des algorithmes de décodage formel et en ligne.

Ces systèmes ne contiennent plus les polynômes de degré élevé qui provenaient des équations de corps, et se comportent mieux en pratique pour les calculs. Ils sont spécifiques aux codes binaires. Pour supprimer les équations de corps, nous avons été amenés à étudier très précisément tous les systèmes de dimension zéro qui ont été utilisés dans des algorithmes de décodage des codes cycliques. Nous donnons une classification fine de tous ces systèmes de dimension zéro.

Les algorithmes de décodage que nous donnons sont automatiques, valables pour n'importe quel code cyclique et permettent de décoder au-delà de la distance minimale du code. En effet, pour tout poids donné v, les algorithmes renvoient l'ensemble des mots de codes à distance au plus v du mot transmis (*décodage en liste*) .

Taille des formules. Nous étudions en détail l'exemple du code à résidus quadratiques [41,21,9]. Dans [RTCY92], les auteurs obtiennent des formules de décodage de degré 4 en les fonctions puissances, et donnent une méthode pour discriminer la bonne solution parmi les 4 obtenues. Grâce aux systèmes de dimension positive, nous pouvons calculer la base de Gröbner du système formel, et obtenir des formules linéaires. Nous montrons que la taille de ces formules est bien trop grande pour être utile, leur évaluation pour chaque mot prenant beaucoup trop de temps. Cela indique que l'approche par décodage en ligne est la plus efficace.

Trace du pré-calcul. Nous utilisons une méthode très utile pour des systèmes à paramètres, la "trace du pré-calcul". Cette méthode consiste, pour chaque code cyclique, à calculer une base de Gröbner d'un système spécialisé pour un mot particulier, en conservant (sous forme de programme C dans notre cas), la trace de tous les calculs effectués. Cette méthode utilise fortement des propriétés de spécialisation de la base de Gröbner formelle. Une fois ce programme généré, il suffit de l'exécuter sur un mot quelconque (de même poids que le mot test) pour obtenir avec une forte probabilité la solution, sans avoir à recalculer explicitement une base de Gröbner. On peut voir ce programme comme une manière efficace d'évaluer une formule.

Nous donnons de nombreux exemples de décodage en ligne, pour des codes de possédant aucun algorithme de décodage connu : des codes à résidus quadratiques

de longueur 71, 89, 113 et un code de longueur 75 n'appartenant à aucune classe particulière de codes cycliques. Nous donnons aussi des résultats pour plusieurs codes BCH de longueur 75 ou 511, et montrons que l'on peut décoder bien au-delà de la capacité de correction de ces codes.

Conclusion Nous obtenons des systèmes de dimension positive qui possèdent les mêmes propriétés de décodage théorique que les systèmes de dimension zéro. En pratique, nos systèmes sont nettement plus efficaces pour le calcul de la base de Gröbner, et nous donnons de nombreux exemples de codes cycliques pour lesquels aucun algorithme de décodage n'existait auparavant. Enfin, la méthode de la trace du précalcul permet un gain considérable sur les temps de décodage par rapport à un calcul direct de base de Gröbner.

Organisation du chapitre. Dans une première partie (section 6.2) nous donnons une classification des différentes mises en équations proposées en dimension zéro. Nous proposons dans la section 6.3 de nouveaux systèmes, de dimension positive, et prouvons qu'ils satisfont la propriété de décodage et sont très efficaces en pratique. On peut ainsi résoudre de nouveaux exemples : nous donnons des "formules" de décodage pour le code à résidus quadratiques de longueur 41. Enfin, la section 6.4 présente de nombreux exemples de décodage en ligne et d'applications de la méthode de trace du pré-calcul employée pour plus d'efficacité.

Tous les calculs ont été effectués à partir de Maple V.5 en utilisant une interface avec les programmes de calcul de bases de Gröbner FGb sur les corps finis, implantés par Jean-Charles Faugère.

Les résultats de ce chapitre ont été obtenus en collaboration avec Daniel Augot et Jean-Charles Faugère. Ils ont fait l'objet d'une présentation à la conférence ISIT à Yokohama (Japon) en Juillet 2003 [ABF03]. Un article correspondant est disponible sous forme de rapport de recherche INRIA [ABF02].

6.2 Classification des systèmes de dimension zéro

Il existe une multitude d'articles consacrés au décodage algébrique des codes cycliques, qui diffèrent par le choix du système algébrique utilisé, l'obtention d'un algorithme de décodage en ligne ou formel, et l'utilisation de bases de Gröbner pour la résolution des systèmes algébriques, ou la résolution à la main de ces systèmes. Nous proposons dans cette section une classification de tous ces systèmes, et des algorithmes de décodage qui en découlent. Les liens algébriques entre les systèmes sont récapitulés sur la figure 6.1 page 114 et proviennent de la proposition 6.2.1. Pour le décodage en ligne, tous ces idéaux sont équivalents.

D'après les exemples des codes RQ de longueur 23 et 31, il semble que le système des équations de Newton soit contenu dans le système des fonctions puissances et

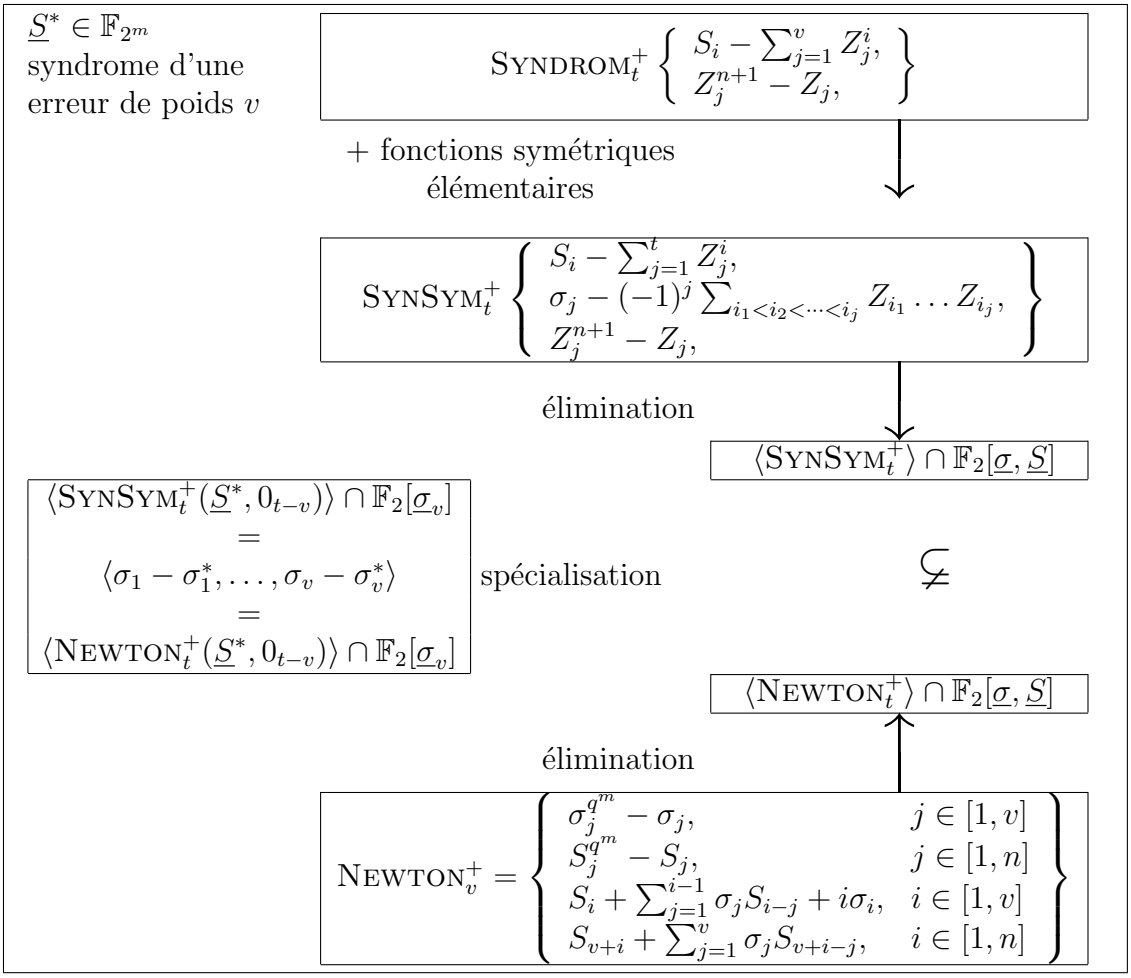


FIG. 6.1 – Idéaux zéro-dimensionnels utilisés pour le décodage des codes cycliques.

fonctions symétriques. La proposition 6.2.1 confirme cette intuition.

Proposition 6.2.1 *Les idéaux vérifient l'inclusion :*

$$\langle \textsc{Newton}_v^+ \rangle \cap \mathbb{F}_2[\underline{\sigma}_v, \underline{S}] \subset \langle \textsc{SynSym}_v^+ \rangle \cap \mathbb{F}_2[\underline{\sigma}_v, \underline{S}] \tag{6.1}$$

mais l'inclusion inverse est fausse en général. Cependant, si $\underline{S}^*$ est le syndrome d'une erreur e de poids $v \leq w$ et que la seule erreur de poids $\leq v$ et de syndrome $\underline{S}^*$ est e (ce qui est vérifié si $v \leq t$), alors[1]

$$\begin{aligned}\langle \textsc{SynSym}_w^+(\underline{S}^*), \sigma_{v+2}, \ldots, \sigma_w) \rangle \cap \mathbb{F}_2[\underline{\sigma}_v] &= \langle \textsc{Newton}_w^+(\underline{S}^*), \sigma_{v+2}, \ldots, \sigma_w) \rangle \cap \mathbb{F}_2[\underline{\sigma}_v] \\ &= \langle \sigma_1 - \sigma_1^*, \ldots, \sigma_v - \sigma_v^*, \sigma_{v+1}, \ldots, \sigma_w \rangle\end{aligned}$$

avec $\sigma_1^*, \ldots, \sigma_v^*$ les fonctions symétriques élémentaires associées à e.

Démonstration Pour l'inclusion 6.1, les deux idéaux sont radicaux donc il est équivalent de prouver l'inclusion inverse des variétés associées (théorème du Null-StellenSatz de Hilbert).

Soit $(\underline{\sigma}_v^*, \underline{S}^*) \in V(\langle \textsc{SynSym}_v^+ \rangle \cap \mathbb{F}_2[\underline{\sigma}_v, \underline{S}])$, alors (propriété de projection) il existe $\underline{Z}_v^* \in \mathbb{F}_{2^m}^v$ tels que $(\underline{Z}_v^*, \underline{\sigma}_v^*, \underline{S}^*) \in V(\textsc{SynSym}_v^+)$. Posons $S_i^* = \sum_{j=1}^v Z_j^{*i}$ pour $i \notin Q$, alors $\underline{S}^*$ et les $\overline{S}^*$ et $\underline{\sigma}_v^*$ sont solutions du système $\textsc{Newton}_v^+$ et $(\underline{\sigma}_v^*, \underline{S}^*) \in V(\langle \textsc{Newton}_v^+ \rangle \cap \mathbb{F}_2[\underline{\sigma}_v, \underline{S}])$.

Ces idéaux sont distincts pour le code de Golay $[23, 12, 7]$ (voir page 48). L'égalité des idéaux spécialisés sera prouvée plus généralement dans la proposition 6.3.1 page 118. □

La figure 6.1 récapitule les liens entre tous les idéaux de dimension zéro.

Efficacité des algorithmes Dans la pratique, il est souvent impossible de calculer la base de Gröbner formelle de ces idéaux. Cela est dû en partie aux deux points suivants :

Nombre de solutions. La complexité du calcul de la base de Gröbner d'un système de dimension zéro en N variables possédant D solutions peut être estimée à $O(ND^3)$ si l'on utilise un algorithme de changement d'ordre [FGLM93] (les systèmes $\textsc{SynSym}_v^+$ et $\textsc{Syndrom}_v^+$ sont déjà des bases de Gröbner). Or, l'idéal $\langle \textsc{Syndrom}_t^+ \rangle$ pour le poids maximal t a $D = (n+1)^t$ solutions, et le calcul de la base de Gröbner devient vite impraticable. L'idéal $\langle \textsc{SynSym}_t^+ \rangle \cap \mathbb{F}_2[\underline{\sigma}, \underline{S}]$ possède encore un nombre très important de solutions, plus de $\frac{(n+1)^t}{t!}$, même si le nombre de solutions parasites est réduit d'un facteur de l'ordre de $t!$ par rapport au système de Loustaunau et York. Le tableau 6.1 récapitule le nombre de solutions de chaque système pour différents codes à résidus quadratiques. Notons qu'un idéal de dimension zéro résolvant le problème du décodage comporte au minimum autant de solutions que le nombre

[1] Remarquons qu'il suffit d'ajouter à l'idéal $\sigma_{v+2}, \ldots, \sigma_t$ pour qu'il contienne σ_{v+1}

d'erreurs possibles, et il y en a $\sum_{k=0}^{t}\binom{n}{k}$ (ce qui donne le coût d'une méthode exhaustive). La complexité du calcul de la base de Gröbner est donc nécessairement élevée.

code	$[n,k,d]$	RQ 23	RQ 31	RQ 41	RQ 73
$\mathbb{F}_{2^m}$	$\mathbb{F}_{2^m}$	$\mathbb{F}_{2048}$	$\mathbb{F}_{32}$	$\mathbb{F}_{2^{20}}$	$\mathbb{F}_{512}$
t	$\lfloor\frac{d}{2}\rfloor$	3	3	4	6
nombre de solutions de $\langle \textsc{Newton}_t^+\rangle \cap \mathbb{F}_2[\underline{\sigma},\underline{S}]$	?	51 200	5 984	?	?
nombre de solutions de $\langle \textsc{Syndrom}_t^+\rangle$	$(n+1)^t$	13 824	32 768	3 111 696	74^6 $> 16\cdot 10^{10}$
nombre de solutions de $\langle \textsc{SynSym}_t^+\rangle \cap \mathbb{F}_2[\underline{\sigma},\underline{S}]$	? $\geq \frac{(n+1)^t}{t!}$	2 600 $\geq$ 2 304	5 984 $\geq$ 5 462	? $\geq$ 129 654	? $\geq$ 228 064 570
nombre d'erreurs	$\sum_{k=0}^{t}\binom{n}{k}$	2 048	4 992	112 792	186 404 114

Tab. 6.1 – Nombre de solutions d'idéaux avec équations de corps

Équations de degré élevé. Tous ces idéaux contiennent les équations de corps $S_i^{2^m}+S_i$, donc les polynômes qui interviennent dans la base de Gröbner sont de degré de l'ordre de 2^m. Dès que le corps des syndromes est gros le pré-calcul de la base de Gröbner devient impossible. Cette croissance des degrés est un facteur très limitant pour le calcul de la base de Gröbner, et même pour le code à résidus quadratiques de longueur 41 nous n'avons pas réussi à calculer la base de Gröbner de $\langle \textsc{Syndrom}_v^+\rangle \cap \mathbb{F}_2[\underline{\sigma}_v,\underline{S}]$. Pour le système $\textsc{Newton}_v^+$, la seule manière de calculer la base de Gröbner est de supprimer provisoirement les équations de corps, de calculer une base de Gröbner sans équations de corps, puis d'ajouter au résultat les équations de corps. Nous allons voir dans la section suivante que ce résultat peut se formaliser, car nous donnons des algorithmes de décodage utilisant le système des équations de Newton sans équations de corps.

Conclusion pour les systèmes de dimension zéro. Pour le calcul de la base de Gröbner formelle, le système le mieux adapté semble être $\textsc{SynSym}_v^+$: l'idéal $\langle \textsc{SynSym}_v^+\rangle \cap \mathbb{F}_2[\underline{\sigma}_v]$ est celui qui possède le moins de solutions, et le système $\textsc{SynSym}_v^+$ est déjà une base de Gröbner (il suffit donc d'utiliser un algorithme de changement d'ordre comme FGLM). Pour le système des équations de Newton, il est nécessaire de retirer les équations de corps pour pouvoir faire les calculs, mais dans ce cas il faut prouver que l'on obtient bien des formules de décodage (ce qui est fait dans la section suivante). Dans le cas d'un décodage en ligne, les deux systèmes ont la même solution et sont radicaux.

6.3 Nouvelles mises en équations

Dans cette section nous présentons les résultats obtenus pour le système des équations de Newton sans équations de corps. Ce système comporte à priori beaucoup plus de solutions que lorsque l'on garde les équations de corps, en particulier des solutions dans la clôture algébrique de $\mathbb{F}_{2^m}$. Nous montrerons que, dans les conditions du décodage, il n'y a en fait aucune solution parasite : l'idéal des équations de Newton (pour le bon poids) spécialisé sur un syndrome particulier possède une unique solution et est radical (proposition 6.3.1 et Algorithme 3). La base de Gröbner réduite formelle contient alors des formules de degré 1 en les σ_j qui, spécialisées sur un syndrome, donnent l'unique solution (proposition 6.3.2 et Algorithme 4).

Nous étudierons ensuite d'autres systèmes, obtenus en éliminant les fonctions puissances inconnues, et donnerons leur intérêt pour le décodage. La figure 6.2 récapitule tous les liens entre les systèmes de dimension positive, que nous prouverons dans les sections suivantes.

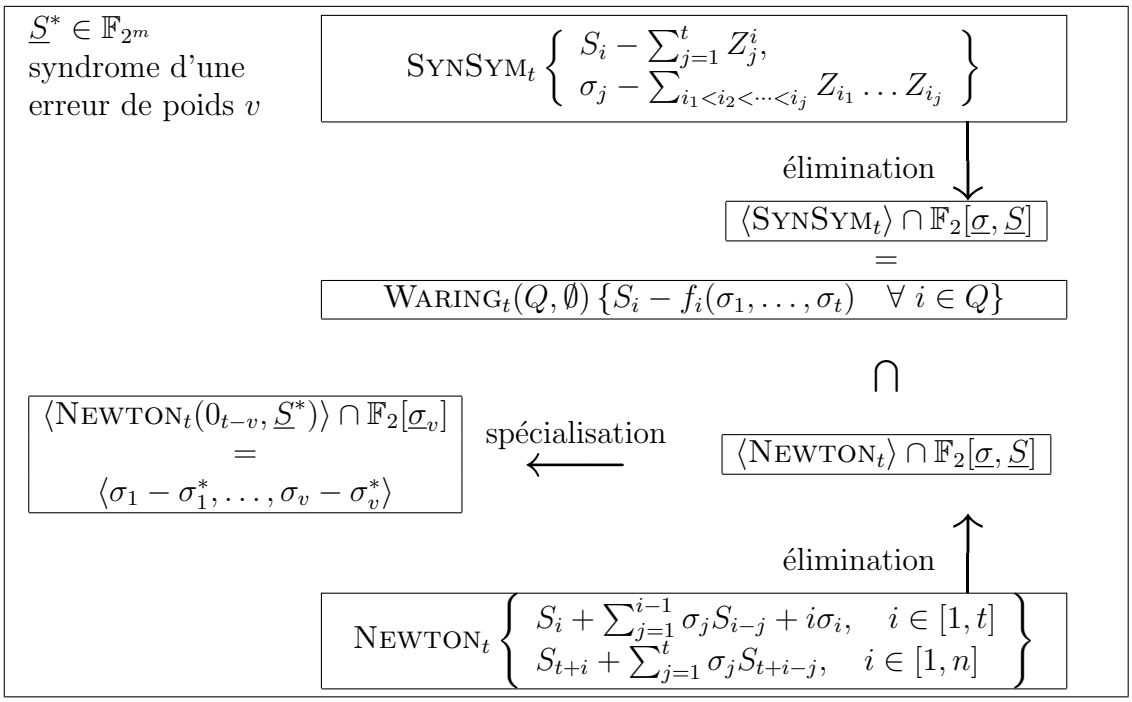


FIG. 6.2 – Idéaux de dimension positive utilisés pour le décodage des codes cycliques

6.3.1 Le système des équations de Newton

Considérons le système des équations de Newton en supprimant les équations de corps :

$$\textsc{Newton}_v = \left\{ \begin{array}{l} S_i + \sum_{j=1}^{i-1} \sigma_j S_{i-j} + i\sigma_i,\ i \in [1;v] \\ S_{v+i} + \sum_{j=1}^{v} \sigma_j S_{v+i-j},\ i \in [1;n] \end{array} \right\}$$

avec $\textsc{Newton}_v \subset \mathbb{F}_2[\underline{\sigma}_v, \underline{S}, \overline{S}]$.

Proposition 6.3.1 (Décodage en ligne) *Si $\underline{S}^* \subset \mathbb{F}_{2^m}$ est le syndrome d'une erreur e de poids $v \leq w$ et que la seule erreur de poids $\leq v$ et de syndrome $\underline{S}^*$ est e (ce qui est vérifié si $v \leq t$), alors*

$$\langle \textsc{Newton}_w(\underline{S}^*), \sigma_{v+2}, \ldots, \sigma_w \rangle \cap \mathbb{F}_{2^m}[\underline{\sigma}_w] = \langle \textsc{Newton}_w^+(\underline{S}^*), \sigma_{v+2}, \ldots, \sigma_w \rangle \cap \mathbb{F}_{2^m}[\underline{\sigma}_w]$$
$$= \langle \textsc{SynSym}_w^+(\underline{S}^*), \sigma_{v+2}, \ldots, \sigma_w \rangle \cap \mathbb{F}_{2^m}[\underline{\sigma}_w] = \langle \sigma_1 - \sigma_1^*, \ldots, \sigma_v - \sigma_v^*, \sigma_{v+1}, \ldots, \sigma_w \rangle$$

avec $\sigma_1^*, \ldots, \sigma_v^*$ les fonctions symétriques élémentaires associées à e.

Démonstration Nous pouvons sans restriction supposer $w = v + 1$. Les inclusions suivantes sont vérifiées :

$$\begin{aligned} I = \langle \textsc{Newton}_{v+1}(\underline{S}^*) \rangle \cap \mathbb{F}_{2^m}[\underline{\sigma}_{v+1}] &\subset \langle \textsc{Newton}_{v+1}^+(\underline{S}^*) \rangle \cap \mathbb{F}_{2^m}[\underline{\sigma}_{v+1}] \\ &\subset \langle \textsc{SynSym}_{v+1}^+(\underline{S}^*) \rangle \cap \mathbb{F}_{2^m}[\underline{\sigma}_{v+1}] \subset \langle \sigma_1 - \sigma_1^*, \ldots, \sigma_v - \sigma_v^*, \sigma_{v+1} \rangle \end{aligned}$$

La première inclusion est évidente (on ajoute les équations de corps), la seconde provient de l'équation 6.1, la dernière du fait que $\underline{\sigma}_{v+1}^*$ (avec $\sigma_{v+1}^* = 0$) est solution de tous ces systèmes et est l'unique solution de $\langle \sigma_1 - \sigma_1^*, \ldots, \sigma_v - \sigma_v^*, \sigma_{v+1} \rangle$ qui est radical. Il suffit donc de montrer que l'idéal I est radical et possède comme unique solution $\underline{\sigma}_{v+1}^*$.

Commençons par prouver l'unicité de la solution. Soit $\underline{\check{\sigma}}_{v+1}^*$ une solution dans $V_{\overline{\mathbb{F}}_2}(I)$, notons $\underline{Z}_{v+1}^*$ les racines du polynôme $L_{v+1}(Z) = Z^{v+1} + \check{\sigma}_1^* Z^v + \cdots + \check{\sigma}_{v+1}^*$ et $S_i^* = (Z_1^*)^i + \ldots + (Z_{v+1}^*)^i$ pour $i \in [1;n]$. Soit c le mot de $(\overline{\mathbb{F}}_2)^n$ de transformée de Fourier $\underline{S}^*, \overline{S}^*$. Nous allons montrer que les Z_i^* de multiplicité impaire sont solutions des équations de longueur $Z^{n+1} + Z$, puis que tous les Z_j^* sont des racines simples, qui sont en fait les localisateurs de e ($j \in [1;v]$) et $Z_{v+1}^* = 0$ (à ré-ordonnancement près des Z_j^*). Nous utilisons la relation suivante, où $L(Z) = Z^{v+1} + \sum_{j=1}^{v+1} \sigma_j Z^{v+1-j}$:

$$\begin{aligned} &(\sum_{i=1}^{n} S_i Z^{n-i} + (v+1)(1+Z^n))L(Z) + (Z^{n+1} + Z)L'(Z) = \\ &\sum_{i=1}^{v+1} \big(S_i + \sum_{j=1}^{i-1} \sigma_j S_{i-j} + i\sigma_i\big)(Z^{n+v+1-i} + Z^{v+1-i}) + \sum_{i=v+2}^{n+v+1} \big(S_i + \sum_{j=1}^{v+1} \sigma_j S_{i-j}\big) Z^{n+v+1-i} \end{aligned} \tag{6.2}$$

Le membre de droite de l'égalité (6.2) est une combinaison des équations de Newton triangulaires et généralisées. Soit $Z_{i_0}^*$ une racine de $L_{v+1}(Z)$ de multiplicité impaire

$2k+1$, ordonnons les racines de sorte que $i_0 = v+1$ et $Z^*_{v+1} = Z^*_v = \ldots = Z^*_{v+1-2k}$. Alors $L_{v+1}(Z) = (Z - Z^*_{v+1})^{2k}\tilde{L}_{v+1}(Z)$ et Z^*_{v+1} est racine simple de $\tilde{L}_{v+1}(Z)$. Pour tout $i \in [1;n]$ on a $S^*_i = \sum_{j=1}^{v+1}(Z^*_j)^i = \sum_{j=1}^{v+1-2k}(Z^*_j)^i$, et si l'on pose $\tilde{L}_{v+1}(Z) = Z^{v+1-2k} + \sum_{j=1}^{v+1-2k}\tilde{\sigma}^*_j Z^{v+1-2k-j}$ alors $(\underline{S}^*, \overline{S}^*, \underline{\tilde{\sigma}}^*_{v+1-2k})$ est solution des équations de Newton triangulaires et généralisées pour le poids $v+1-2k$ et la relation (6.2) donne l'égalité :

$$(\sum_{i=1}^{n} S^*_i Z^{n-i} + (v+1)(1+Z^n))\tilde{L}_{v+1}(Z) + (Z^{n+1}+Z)\tilde{L}'_{v+1}(Z) = 0.$$

Comme Z^*_{v+1} est une racine simple de $\tilde{L}_{v+1}(Z)$, c'est aussi une racine de l'équation de longueur $Z^{n+1}+Z$. Ordonnons maintenant les Z^*_j pour que les v' premiers soient deux à deux distincts et non nuls, et les suivants égaux par paires ou nuls, alors par injectivité de la transformée de Fourier, si on note $Z^*_j = \alpha^{i_j}$ pour $j \in [1;v']$ on a $c = \sum_{j=1}^{v'} x^{i_j}$ qui est un mot binaire de poids $\leq v$. Le théorème d'unicité 2.3.4 impose $e = c$ et donc $\check{\sigma}^*_{v+1} = 0$ et $\check{\sigma}^*_j = \sigma^*_j$ pour $j \in [1;v]$.

Il reste à montrer que l'idéal I est radical. Considérons pour cela l'idéal $I^* = \langle \textsc{Newton}_{v+1}(\underline{S}^*), \sigma_j + \sum_{l_1<l_2<\cdots<l_j} Z_{l_1}\cdots Z_{l_j},\ j \in [1;v+1]\rangle \subset \mathbb{F}_{2^m}[\underline{Z}_{v+1}, \overline{S}, \underline{\sigma}_{v+1}]$. Il vérifie $I^* \cap \mathbb{F}_{2^m}[\overline{S}, \underline{\sigma}_{v+1}] = I$ donc il suffit de prouver que I^* est radical[2]. En substituant dans l'équation de Newton $S_i + \sum_{j=1}^{\min(i-1,v+1)} \sigma_j S_{i-j}$ les S_k, $k<i$, et les σ_j, on obtient :

$$I^* = \left\langle \begin{array}{l} \sigma_j + \sum_{l_1<l_2<\cdots<l_j} Z_{l_1}\cdots Z_{l_j},\ j \in [1;v+1], \\ S^*_i + \sum_{j=1}^{v+1} Z^i_j,\ i \in Q, \qquad S_k + \sum_{j=1}^{v+1} Z^k_j,\ k \notin Q, \\ \sum_{j=1}^{v+1} Z^{i-1}_j (Z^{n+1}_j + Z_j),\ i \in [1;v+1] \end{array} \right\rangle$$

(les $v+1$ dernières relations provenant des équations $S_{n+i} + \sum_{j=1}^{v+1}\sigma_j S_{v+1+i-j}$, $i \in [1;v+1]$).

Notons $F_i = \sum_{j=1}^{v+1} Z^{i-1}_j (Z^{n+1}_j + Z_j), i \in [1;v+1]$ et $\check{\sigma}_j,\ j \in [1;v]$ les fonctions symétriques élémentaires de $Z_1, \ldots, Z_v$ alors $F := F_{v+1} + \sum_{j=1}^{v}\check{\sigma}_j F_{v-j} \in I^*$. Or, on a

$$F = (Z^{n+1}_{v+1} + Z_{v+1})\prod_{j=1}^{v}(Z_{v+1} + Z_j)$$

et l'équation $\prod_{j=1}^{v}(Z_{v+1}+Z_j)$ ne s'annule pas sur l'unique solution de I^* (les racines Z^*_j solutions sont toutes distinctes). En utilisant le théorème de NullStellenSatz, on en déduit qu'il existe un polynôme g tel que $g\prod_{j=1}^{v}(Z_{v+1}+Z_j) + 1 \in I^*$, et donc que $Z^{n+1}_{v+1} + Z_{v+1} \in I^*$.

Ainsi, $I^* = \langle \sigma_j + \sum_{l_1<l_2<\cdots<l_j} Z_{l_1}\cdots Z_{l_j}, j \in [1;v+1], S^*_i + \sum_{j=1}^{v+1} Z^i_j, i \in Q, S_k + \sum_{j=1}^{v+1} Z^k_j, k \notin Q, Z^{n+1}_j + Z_j, i \in [1;v+1]\rangle = \langle \textsc{SynSym}^+_{v+1}(\underline{S}^*), S_k + \sum_{j=1}^{v+1} Z^k_j, k \notin Q\rangle$ et d'après la proposition 2.4.1, I^* est radical. □

[2] L'intersection de deux idéaux radicaux est un idéal radical.

La proposition précédente montre que l'idéal des équations de Newton possède, comme les idéaux avec les équations de corps ou de longueur, la propriété de décodage : il est possible de trouver le poids v de l'erreur en annulant successivement $\sigma_w, \sigma_{w-1}, \ldots$ tant que l'idéal ne contient pas 1, et le système $\langle \text{NEWTON}_v(\underline{S}^*)\rangle \cap \mathbb{F}_{2^m}[\underline{\sigma}_v]$ possède une unique solution, donnée directement par la base de Gröbner réduite du système pour n'importe quel ordre monomial admissible (on choisit l'ordre grevlex pour lequel les calculs sont plus rapides en pratique). Remarquons que les solutions de $\langle \text{NEWTON}_w(\underline{S}^*)\rangle \cap \mathbb{F}_{2^m}[\underline{\sigma}_w]$ donnent l'ensemble des mots de code à distance au plus w du mot transmis, et que s'il n'y en a aucun alors $\langle \text{NEWTON}_w(\underline{S}^*)\rangle \cap \mathbb{F}_{2^m}[\underline{\sigma}_w] = \langle 1\rangle$.

Nous obtenons directement un algorithme de décodage en ligne :

Algorithme 3 (Décodage en ligne sans équations de corps)
Pour chaque mot $\tilde{c}$ reçu,

- *calculer le syndrome de l'erreur $\underline{S}^* = \{\tilde{c}(\alpha^i) \ : \ i \in Q\}$,*
- *calculer la base de Gröbner réduite de $\langle \text{NEWTON}_t(\underline{S}^*)\rangle$ en utilisant un ordre d'élimination des variables $\overline{S}$ et un ordre grevlex pour les variables $\underline{\sigma}_t$,*
- *en spécialisant successivement à zéro $\sigma_t, \sigma_{t-1}, \ldots$ en déduire v le poids de l'erreur et les polynômes $\sigma_j - \sigma_j^*$ pour $1 \leq j \leq v$,*
- *calculer les racines du polynôme localisateur de l'erreur.*

Remarque. En pratique, il est souvent plus rapide de calculer la base de Gröbner en deux fois : on calcule une première base de $\langle \text{NEWTON}_t(\underline{S}^*)\rangle$ pour un ordre grevlex, puis une base pour l'ordre d'élimination. Le lecteur peut se référer à la section 1.2.4 pour plus de détails sur les stratégies de calcul.

Pour le décodage formel, si G_t est une base de Gröbner de $\langle \text{NEWTON}_t\rangle \cap \mathbb{F}_2[\underline{\sigma}_t, \underline{S}]$ pour un ordre d'élimination $\underline{\sigma}_t > \underline{S}$, et que nous la spécialisons en un syndrome $\underline{S}^*$, nous obtenons un système de générateurs $G_t(\underline{S}^*)$ d'un idéal n'ayant qu'une seule solution de multiplicité un (à condition de spécialiser aussi $\sigma_{t-v} = 0, \ldots, \sigma_t = 0$). Cependant, calculer les solutions d'un idéal donné par des générateurs quelconques n'est pas facile, et il faut à priori recalculer une base de Gröbner de l'idéal spécialisé. Or, de la même façon que dans le cas zéro-dimensionnel, un théorème de spécialisation (Théorème 1.3.7 de Fortuna, Gianni, Trager) assure que si l'on spécialise toutes les variables sauf une, la base de Gröbner spécialisée est une base de Gröbner de l'idéal spécialisé. Plus précisément, la proposition suivante établit que, sous réserve de réussir à calculer la base de Gröbner formelle, on peut obtenir des formules de décodage linéaires :

Proposition 6.3.2 (Décodage formel) *Soit $v \leq t$ et G une base de Gröbner réduite de $\langle \text{NEWTON}_v\rangle \cap \mathbb{F}_2[\underline{\sigma}_v, \underline{S}]$ pour un ordre d'élimination $\underline{\sigma}_v > \underline{S}$ et un ordre lexicographique pour le bloc des variables $\underline{\sigma}_v$. Si l'on suppose que $\langle \text{NEWTON}_v\rangle \cap \mathbb{F}_2[\underline{S}]$ est un idéal* radical, *alors G contient pour tout syndrome $\underline{S}^*$ et pour tout j un polynôme de variable principale[3] σ_j, linéaire en σ_j, dont l'initial est dans $\mathbb{F}_2[\underline{S}]$ et*

[3] la variable principale d'un polynôme f est la plus grande variable apparaissant dans f et son initial est le coefficient dominant de f vu comme polynôme en sa variable principale.

ne s'annule pas en $\underline{S}^*$.

Avant de donner la preuve du théorème, nous pouvons faire les remarques suivantes :

1. Le théorème affirme que pour chaque syndrome, il existe un polynôme dont l'initial ne va pas s'annuler, mais ce polynôme n'est pas le même pour tous les syndromes,
2. En pratique, l'idéal $\langle \text{NEWTON}_v \rangle \cap \mathbb{F}_2[\underline{S}]$ est toujours radical, mais nous n'avons pas pu le prouver dans le cas général,
3. L'hypothèse de radicalité n'est nécessaire que pour prouver que G contient des polynômes *linéaires* en σ_j : si l'idéal n'était pas radical, alors G contiendrait pour tout syndrome $\underline{S}^*$ un polynôme de variable principale σ_j, qui spécialisé donnerait un polynôme linéaire $\sigma_j - \sigma_j^*$ à un facteur multiplicatif près dans $\mathbb{F}_{2^m}$,
4. Enfin, si l'on arrive à calculer la base de Gröbner G, et qu'elle ne contient aucun polynôme linéaire en un σ_j, on peut en déduire que les initiaux des polynômes de variables principales σ_j de plus petit degré vont tous s'annuler sur toute spécialisation, et donc que ces polynômes appartiennent au radical de l'idéal. On peut donc recalculer une base de Gröbner de G auquel on a ajouté tous ces initiaux, et répéter cette opération jusqu'à obtenir des polynômes linéaires en σ_j. On peut aussi calculer une base de Gröbner du radical de $\langle \text{NEWTON}_v \rangle \cap \mathbb{F}_2[\underline{S}]$, et ajouter ces polynômes à l'idéal $\langle \text{NEWTON}_v \rangle$ pour obtenir des formules linéaires.

Démonstration Remarquons tout d'abord que $V(\langle \text{NEWTON}_v \rangle \cap \mathbb{F}_2[\underline{S}])$ est exactement l'ensemble des syndromes correspondant à une erreur de poids au plus v (il est évident que $V(\langle \text{NEWTON}_v \rangle \cap \mathbb{F}_2[\underline{S}])$ contient tous les syndromes d'une erreur de poids au plus v, et l'inclusion réciproque est un corollaire de la preuve de la proposition 6.3.1). Soit σ_w la plus petite des variables $\underline{\sigma}_v$, et soit $g \in G$ un polynôme de variable principale σ_w et de degré minimal k en σ_w (dans l'ensemble des polynômes de G de variable principale σ_w). On peut écrire $g = q(\underline{S})\sigma_w^k + \dots$ où $q(\underline{S})$ est l'initial de g. Si $k > 1$, comme pour tout syndrome $\underline{S}^*$ d'une erreur de poids au plus w, $G(\underline{S}^*)$ contient un polynôme linéaire en σ_w, d'après le théorème de spécialisation 1.3.7 on doit avoir $q(\underline{S}^*) = 0$ et donc $q(\underline{S}) \in \langle \text{NEWTON}_v \rangle \cap \mathbb{F}_2[\underline{S}]$ car l'idéal est radical, ce qui contredit le fait que G soit une base de Gröbner réduite, et ainsi $k = 1$.

Si maintenant $\sigma_{v'}$ est une variable quelconque, d'après ce qui précède (en changeant l'ordre des variables $\underline{\sigma}_v$) on sait que l'idéal $\langle \text{NEWTON}_v \rangle$ contient un polynôme de la forme $q_{v'}(\underline{S})\sigma_{v'} + p_{v'}(\underline{S})$ avec $q_{v'}$ réduit par rapport à G. Il existe donc $g \in G$ tel que $\text{LT}(g)$ divise $\text{LT}(q_{v'})\sigma_{v'}$ mais ne divise pas $\text{LT}(q_{v'})$. On a donc $\text{LT}(g) = h\sigma_{v'}$ avec $h \in \mathbb{F}_2[\underline{S}]$ et g est un polynôme de G de variable principale $\sigma_{v'}$, linéaire en $\sigma_{v'}$ et d'initial dans $\mathbb{F}_2[\underline{S}]$. □

La base de Gröbner réduite formelle contient beaucoup de polynômes (375 polynômes en 7 variables pour le code RQ 41), mais très peu d'entre eux serviront

effectivement au décodage : en effet, pour décoder une erreur de poids v, il suffit de v équations de variable principale σ_i linéaires en σ_i pour $i \in [1; v]$ dont les initiaux ne s'annulent pas. Nous pouvons au moins ne garder pour tout $1 \leq j \leq v$ que les polynômes d'initial σ_j linéaires en σ_j, la proposition 6.3.2 assure que pour tout syndrome d'une erreur de poids $\leq v$ l'un de ces initiaux au moins ne s'annulera pas. Si nous arrivons à prouver (par exemple par recherche exhaustive, c'est ce qui est fait par Chen et al. [RYT90]) que l'un de ces initiaux ne s'annule sur aucun syndrome, alors nous pouvons ne garder que ce polynôme. Nous obtenons finalement l'algorithme de décodage suivant (valable si $\langle \text{NEWTON}_v \rangle \cap \mathbb{F}_2[\underline{S}]$ est radical) :

Algorithme 4 (Décodage formel sans équations de corps)

Pré-calcul.

1. *calculer la base de Gröbner réduite de* $\langle \text{NEWTON}_t(\underline{S}) \rangle \cap \mathbb{F}_2[\underline{\sigma_t}, \underline{S}]$ *en utilisant un ordre d'élimination* $\underline{\sigma_v} > \underline{S}$ *et un ordre Lex pour les variables* $\underline{\sigma_t}$,
2. *extraire de* G *un sous-ensemble* G' *tel que tout* $g \in G'$ *soit un polynôme de variable principale* σ_i *linéaire en* σ_i, *et que pour tout syndrome* $\underline{S^*}$ *correspondant à une erreur de poids* $v \leq t$ *il existe* $(g_1, \ldots, g_v) \in G'$ *avec* $g_i(\underline{S^*}) = \sigma_i - \sigma_i^*$ *(à un élément de* $\mathbb{F}_{2^m}$ *près),*

Notons que G' *contient au moins* t *polynômes.*

Décodage. *Pour chaque mot* e *à décoder, calculer le syndrome* $\underline{S^*}$, *spécialiser* G' *en* $\underline{S^*}$, *déterminer le poids de* e *et son polynôme localisateur, et calculer les racines de ce polynômes localisateur.*

Remarque. De même que pour l'algorithme 3, le calcul de la base de Gröbner peut se faire en plusieurs étapes : on calcule d'abord une base pour l'ordre grevlex, puis pour un ordre d'élimination (grevlex,grevlex), puis avec l'ordre (Lex,grevlex). Remarquons que cette dernière étape n'est pas forcément nécessaire, en général la base de Gröbner pour l'ordre d'élimination (grevlex,grevlex) contient déjà des polynômes linéaires en les σ_j. Et de même que dans le cas zéro dimensionnel, on donne un poids i aux variables S_i et σ_i.

Ces nouveaux systèmes conduisent donc comme ceux en dimension zéro à des algorithmes de décodage formel et en ligne pour les codes cycliques. Cependant, ils se comportent beaucoup mieux pour le calcul de la base de Gröbner.

6.3.2 Taille des formules

Nous avons déjà donné pages 48 et 49 les exemples des codes à résidus quadratiques de longueur 23 et 31 (les bases sans équations de corps ne comportent pas les polynômes $\sigma_j^{2^m} + \sigma_j$).

Dans le cas du code RQ 41, nous avons réussi à calculer une base de Gröbner de l'idéal $\langle \text{NEWTON}_4 \rangle \cap \mathbb{F}_2[\underline{\sigma_v}, \underline{S}]$ de la manière suivante : on commence par éliminer du système NEWTON_4 tous les syndromes inconnus, puis on calcule une base de

$$
\begin{cases}
\boldsymbol{\sigma_4} S_{23} + \ldots, \\
\boldsymbol{\sigma_4} S_9 + \ldots, \\
\boldsymbol{\sigma_4} S_5 + \ldots, \\
\boldsymbol{\sigma_4} S_1 + \ldots, \\
\boldsymbol{\sigma_3} + \sigma_2 S_1 + \ldots, \\
\boldsymbol{\sigma_2}(S_{23}S_9^2S_1 + S_{23}S_5S_1^{14} + S_{23}S_1^{19} + S_9^4S_1^6 + S_9^3S_5S_1^{10} + S_9^3S_1^{15} + S_9^2S_5S_1^{60} + S_9^2S_5S_1^{19} + S_9^2S_1^{24} + S_9S_5^3S_1^{18} + S_9S_5^2S_1^{23} + S_9S_5S_1^{28} + S_5^3S_1^{27} + S_5^2S_1^{32} + S_5S_1^{37} + S_1) + \ldots, \\
\boldsymbol{\sigma_2}(S_{23}S_5S_1^{13} + S_{23}S_1^{59} + S_9^4S_5 + S_9^4S_1^5 + S_9^3S_5S_1^9 + S_9^2S_5S_1^{59} + S_9^2S_1^{23} + S_9S_5^2S_1^{22} + S_9S_5S_1^{27} + S_9S_1^{32} + S_5^3S_1^{26} + S_5^2S_1^{31} + S_5S_1^{36} + S_1^{41}) + \ldots, \\
\boldsymbol{\sigma_2}(S_{23}S_5 + S_{23}S_1^{46} + S_{23}S_1^5 + S_9^3S_5S_1^{37} + S_9^3S_1 + S_9^2S_5^2 + S_9^2S_5S_1^5 + S_9S_5^3S_1^4 + S_9S_5^2S_1^{50} + S_9S_5^2S_1^9 + S_9S_1^{60} + S_9S_1^{19}) + \ldots, \\
\boldsymbol{\sigma_2}(S_9^3 + S_9S_5^2S_1^8 + S_9S_1^{59} + S_9S_1^{18} + S_5^2S_1^{58} + S_1^{27}) + \ldots, \\
\boldsymbol{\sigma_2}(S_9^3S_1^{38} + S_9^2S_5S_1^{42} + S_9^2S_5S_1 + S_9S_5^3 + S_9S_5^2S_1^{46} + S_9S_5S_1^{10} + S_9S_1^{15} + S_5^3S_1^{50}) + \ldots, \\
\boldsymbol{\sigma_2}(S_{23}S_9S_1^{32} + S_{23}S_1^{41} + S_9^3S_5S_1^{32} + S_9^2S_5S_1^{41} + S_9^2S_5 + S_9^2S_1^{46} + S_9^2S_1^5 + S_9S_5^2S_1^{45} + S_9S_1^{55} + S_5^3S_1^{49} + S_5^2S_1^{54} + S_5S_1^{59}) + \ldots, \\
\boldsymbol{\sigma_2}(S_9^3S_1^{33} + S_9^2S_5^2S_1^{32} + S_9^2S_5S_1^{37} + S_9^2S_1 + S_9S_5^2S_1^{41} + S_9S_5^2 + S_9S_1^{51} + S_5^3S_1^{45} + S_5S_1^{55} + S_1^{60}) + \ldots, \\
\boldsymbol{\sigma_2}(S_{23}S_5S_1^{31} + S_{23}S_1^{36} + S_9^2S_5^2S_1^{31} + S_9^2 + S_9S_5^2S_1^{40} + S_9S_1^{50} + S_9S_1^9 + S_5S_1^{54}) + \ldots, \\
\boldsymbol{\sigma_2}(S_9^4S_1^{20} + S_9^2S_1^{38} + S_9S_5S_1 + S_9S_1^{47} + S_9S_1^6 + S_5^3 + S_5^2S_1^{46} + S_5^2S_1^5 + S_5S_1^{51} + S_5S_1^{10}) + \ldots, \\
\boldsymbol{\sigma_2}(S_{23}S_1^{32} + S_9^2S_1^{37} + S_9S_5^3S_1^{31} + S_9S_5^2S_1^{36} + S_9S_5 + S_9S_1^{46} + S_9S_1^5 + S_5^2S_1^{45} + S_5S_1^{50} + S_1^{55}) + \ldots, \\
\boldsymbol{\sigma_2}(S_9^4S_5S_1^{10} + S_9^4S_1^{15} + S_9^3S_1^{24} + S_9^2S_5^2S_1^{23} + S_9^2S_1^{33} + S_9S_5^2S_1^{32} + S_9S_5S_1^{37} + S_5^2 + S_1^{51} + S_1^{10}) + \ldots, \\
\boldsymbol{\sigma_2}(S_{23}S_9^3 + S_{23}S_9S_1^{18} + S_9^4S_1^{14} + S_9^2S_5S_1^{27} + S_9^2S_1^{32} + S_9S_5^3S_1^{26} + S_9S_5^2S_1^{31} + S_9S_5S_1^{36} + S_9S_1^{41} + S_9 + S_5^2S_1^{40} + S_5S_1^{45}) + \ldots, \\
\boldsymbol{\sigma_2}(S_{23}S_9S_5S_1^{13} + S_{23}S_9S_1^{18} + S_{23}S_5S_1^{22} + S_{23}S_1^{27} + S_9^4S_1^{14} + S_9^3S_1^{23} + S_9^2S_5^2S_1^{22} + S_9^2S_5S_1^{27} + S_9^2S_1^{32} + S_9S_1^{41} + S_9 + S_5^3S_1^{35} + S_5S_1^{45} + S_1^9) + \ldots, \\
\boldsymbol{\sigma_2}(S_{23}S_9^2S_5 + S_{23}S_5S_1^{18} + S_9^4S_5S_1^5 + S_9^4S_1^{10} + S_9^3S_5S_1^{14} + S_9^2S_1^{28} + S_9S_5^2S_1^{27} + S_9S_5S_1^{32} + S_9S_1^{37} + S_5^3S_1^{31} + S_5^2S_1^{36} + S_5S_1^{41} + S_5 + S_1^{46}) + \ldots, \\
\boldsymbol{\sigma_2}(S_{23}S_9^2S_1^5 + S_{23}S_1^{23} + S_9^4S_5S_1^5 + S_9^3S_1^{19} + S_9^2S_5S_1^{23} + S_9S_5^3S_1^{22} + S_9S_1^{37} + S_5 + S_1^{46} + S_1^5) + \ldots, \\
\boldsymbol{\sigma_2}(S_{23}S_9S_5S_1^8 + S_{23}S_9S_1^{13} + S_{23}S_5S_1^{17} + S_{23}S_1^{22} + S_9^5 + S_9^4S_1^9 + S_9^2S_5S_1^{22} + S_9S_5^2S_1^{26} + S_9S_5S_1^{31} + S_5^3S_1^{30} + S_1^{45} + S_1^4) + \ldots, \\
\boldsymbol{\sigma_2}(S_{23}S_5S_1^9 + S_{23}S_1^{14} + S_9^4S_1 + S_9^3S_5^2 + S_9^2S_5^2S_1^9 + S_9S_5^2S_1^{18} + S_9S_5S_1^{23} + S_9S_1^{28} + S_5^2S_1^{27} + S_1^{37}) + \ldots, \\
\boldsymbol{\sigma_2}(S_{23}S_9S_1 + S_{23}S_5^2 + S_9^3S_1^6 + S_9S_5^2S_1^{14} + S_9S_1^{24} + S_1^{33}) + \ldots, \\
\boldsymbol{\sigma_2}(S_{23}S_1^{10} + S_9^3S_5S_1 + S_9^3S_1^6 + S_9^2S_5^3 + S_9^2S_5^2S_1^5 + S_9^2S_5S_1^{10} + S_9S_5S_1^{19} + S_5^3S_1^{18} + S_5^2S_1^{23} + S_5S_1^{28}) + \ldots, \\
\boldsymbol{\sigma_2}(S_9^2S_1^2 + S_9S_1^{11} + S_5^4 + S_5^2S_1^{10} + S_5S_1^{15} + S_1^{20}) + \ldots, \\
350 \text{ polynômes en } S_{23}, S_9, S_5, S_1.
\end{cases}
$$

FIG. 6.3 – Base d'élimination de $\langle \text{NEWTON}_4 \rangle \cap \mathbb{F}_2[\sigma_v, S]$ pour le code RQ [41,22,9]

Gröbner grevlex de l'idéal $\langle \text{NEWTON}_4 \rangle \cap \mathbb{F}_2[S_{40}, S_{36}, S_{32}, S_{20}, S_{18}, S_{16}, S_{10}, S_8, S_4, S_2, \sigma_4, \sigma_3, \sigma_2, \sigma_1, S_{39}, S_{37}, S_{33}, S_{31}, S_{25}, S_{23}, S_{21}, S_9, S_5, S_1]$. Cette première étape prend environ 176 secondes sur un Pentium 4 à 2.8 GHz. On peut alors éliminer les variables S_{40}, S_{36}, S_{32}, S_{20}, S_{18}, S_{16}, S_{10}, S_8, S_4, S_2, S_{39}, S_{37}, S_{33}, S_{31}, S_{25}, S_{21} et σ_1 pour lesquelles la base de Gröbner contient un polynôme linéaire, on a donc une base de Gröbner de $\langle \text{NEWTON}_4 \rangle \cap \mathbb{F}_2[\sigma_4, \sigma_3, \sigma_2, S_{23}, S_9, S_5, S_1]$. On calcule alors une base de Gröbner de ce système pour un ordre d'élimination (grevlex,grevlex) $\sigma_4 > \sigma_3 > \sigma_2 \gg S_{23} > S_9 > S_5 > S_1$. On obtient après 2h30 de calcul le système de la figure 6.3. Chaque polynôme linéaire en un σ_j contient de l'ordre de 1400 termes (entre 1348 et 1442) et est de degré environ 110 (entre 106 et 115). Le coût de l'évaluation de ces polynômes est important (il faut évaluer 3 polynômes pour chaque mot à décoder, chacun de degré 110 avec 1400 termes), et conduit à un algorithme de décodage qui n'est pas très efficace. Nous verrons dans les sections suivantes comment obtenir un décodage en ligne beaucoup plus efficace.

6.3.3 Les formules de Waring

Le système des équations de Newton comporte un très grand nombre d'inconnues qu'il faut éliminer : tous les S_i pour $i \notin Q$. Dans cette section nous donnons d'autres systèmes dans lesquels ces inconnues ont déjà été éliminées, et nous faisons le lien entre le système des équations de Newton et le système des fonctions puissances et fonctions symétriques.

Par substitutions successives dans l'équation $S_i + \sum_{j=1}^{\max(i,v)} \sigma_j S_{i-j}$ à l'aide des équations de Newton définissant $S_j, j < i$ nous pouvons éliminer ces $S_j, j < i$. Nous obtenons des expressions $S_i = f_i(\sigma_1, \ldots, \sigma_v)$ exprimant les fonctions puissances en fonction des fonctions symétriques élémentaires. Ces f_i sont les fonctions de Waring [LN97, p. 30], f_i peut être obtenue comme le $i^{\text{ème}}$ coefficient d'une série :

$$\begin{aligned} f_i(\sigma_1, \ldots, \sigma_v) &= [Z^i](Z^{n+1} + Z)\frac{\sum_{j=1}^{v} j\sigma_j Z^{j-1}}{1 + \sum_{j=1}^{v} \sigma_j Z^j} = [Z^i](Z^{n+1} + Z)\frac{\sigma'(Z)}{\sigma(Z)} \quad (6.3) \\ &= \sum (-1)^{i_2+i_4+i_6+\cdots} \frac{(i_1 + i_2 + \cdots + i_v - 1)!}{i_1! \cdots i_v!}\, i\, \sigma_1^{i_1} \cdots \sigma_v^{i_v} \end{aligned}$$

où la somme est prise sur l'ensemble des v-uplets d'entiers naturels $(i_1, \ldots, i_v)$ tels que $i_1 + 2i_2 + \cdots + vi_v = i$.

Utilisant le fait que les équations de Newton sont cycliques, nous pouvons aussi partir de l'équation qui définit S_{i+v}, et éliminer de cette équation les $S_{i+j}, j \geq 1$ (cette fois-ci en utilisant pour éliminer S_{i+j} l'équation de Newton définissant S_{i+j+v}). Nous obtenons une formule exprimant S_i en fonction des $S_j, j \in [1; v]$, il suffit alors d'utiliser la partie triangulaire des équations de Newton.

Nous obtenons une formule $\sigma_v^{n-i+1} S_i = \sigma_v \tilde{f}_{n-i}(\sigma_1, \ldots, \sigma_v)$ qui est de degré $n - i + 2$ au lieu de i pour les formules de Waring, et comporte σ_v en facteur. Nous appellerons les $\tilde{f}_{n-i}$ les fonctions de Waring inverses, $\tilde{f}_{n-i}$ peut être obtenue comme

le $(n-i)^{\text{ème}}$ coefficient d'une série :

$$\begin{aligned}\tilde{f}_{n-i}(\sigma_1,\dots,\sigma_v) &= [Z^{n-i}]\sigma_v^{n-i}(Z^{n+1}+Z)\frac{\sum_{j=0}^{v-1}(v-j)\sigma_j Z^{v-j-1}}{Z^v+\sum_{j=1}^{v}\sigma_j Z^{v-j}}\\ &= [Z^{n-i}]\sigma_v^{n-i}(Z^{n+1}+Z)\frac{\sigma'_{\text{rec}}(Z)}{\sigma_{\text{rec}}(Z)}\\ &= \sum(-1)^{j_{v-2}+j_{v-4}+\cdots}\frac{(n-i-1-j_v)!}{j_1!\cdots j_{v-1}!\,(n-i-\sum_{l=1}^{v}j_l)!}(n-i)\sigma_1^{j_1}\cdots\sigma_v^{j_v}\end{aligned}$$

où la somme est prise sur l'ensemble des v-uplets d'entiers naturels $(j_1,\dots,j_v)$ tels que $j_1+2j_2+\cdots+vj_v=(v-1)(n-i)$. Nous utiliserons la notation suivante :

$$\begin{aligned}&\textsc{Waring}_v(E_1,E_2)=\\ &\qquad\{S_i-f_i(\sigma_1,\dots,\sigma_v),i\in E_1\}\cup\{\sigma_v(\sigma_v^{n-j}S_i-\tilde{f}_{n-i}(\sigma_1,\dots,\sigma_v)),\ i\in E_2\}\end{aligned}$$

Pour relier ces formules à d'autres systèmes connus, considérons le système des fonctions puissances et fonctions symétriques élémentaires :

$$\textsc{SynSym}_v=\left\{\begin{array}{ll}S_i-\sum_{j=1}^{v}Z_j^i & \forall i\in Q\\ \sigma_j-\sum_{l_1<\cdots<l_j}Z_{l_1}\cdots Z_{l_j} & \forall j\in[1;v]\end{array}\right\}$$

avec $\textsc{SynSym}_v\subset\mathbb{F}_2[\underline{Z}_v,\underline{S},\underline{\sigma}]$. Nous avons $\langle\textsc{SynSym}_v\rangle\cap\mathbb{F}_2[\underline{S},\underline{\sigma}]\subset\langle\textsc{Newton}_v\rangle\cap\mathbb{F}_2[\underline{S},\underline{\sigma}]$, mais il n'y a pas égalité. Cependant, l'idéal $\langle\textsc{SynSym}_v\rangle\cap\mathbb{F}_2[\underline{S},\underline{\sigma}]$ admet comme système de générateurs les fonctions de Waring, et est donc inclus dans l'idéal $\langle\textsc{Newton}_v\rangle\cap\mathbb{F}_2[\underline{S},\underline{\sigma}]$:

Proposition 6.3.3 *Les trois idéaux* $\textsc{Newton}_v$, $\textsc{SynSym}_v$ *et* $\textsc{Waring}_v(Q,\emptyset)$ *vérifient les relations :*

$$\begin{array}{rcl}\langle\textsc{Waring}_v(Q,\emptyset)\rangle &=& \langle\textsc{SynSym}_v\rangle\cap\mathbb{F}_2[\underline{\sigma}_v,\underline{S}]\\ \langle\textsc{Waring}_v(Q,Q)\rangle &\subseteq& \langle\textsc{Newton}_v\rangle\cap\mathbb{F}_2[\underline{\sigma}_v,\underline{S}]\end{array}$$

et l'idéal d'élimination $\langle\textsc{Waring}_v(Q,\emptyset)\rangle$ *est un idéal premier (c'est l'idéal des relations entre les* σ_j*). Plus précisément,*

$$\langle\textsc{Newton}_v\rangle\cap\mathbb{F}_2[\underline{\sigma}_v,\underline{S}]=\langle S_i-f_i(\underline{\sigma}_v),i\in Q,S_i-f_{n+i}(\underline{\sigma}_v),i\in Q\cap[1;v]\rangle$$

Démonstration Pour la première égalité, nous avons

$$\begin{array}{rl}&\langle\textsc{Waring}_v(Q,\emptyset),\sigma_j+\sum_{l_1<l_2<\cdots<l_j}Z_{l_1}\cdots Z_{l_j},\ j\in[1;v]\ \rangle\\ =&\langle S_i+f_i(\sigma_1,\dots,\sigma_v),\sigma_j+\sum_{l_1<l_2<\cdots<l_j}Z_{l_1}\cdots Z_{l_j}\rangle\\ =&\langle S_i+f_i(Z_1+\cdots+Z_v,\dots,Z_1\cdots Z_v),\sigma_j+\sum_{i_1<\cdots<i_j}Z_{i_1}\cdots Z_{i_j}\rangle\\ =&\langle S_i+Z_1^i+\cdots+Z_v^i,\sigma_j+\sum_{l_1<l_2<\cdots<l_j}Z_{l_1}\cdots Z_{l_j}\rangle\\ =&\langle\textsc{SynSym}_v\rangle\end{array}$$

De plus, $\text{WARING}_v(Q,\emptyset)$ est une base de Gröbner de $\langle \text{WARING}_v(Q,\emptyset)\rangle$ pour un ordre Lex tel que $\underline{S} > \underline{\sigma}_v$. Soit G une base de Gröbner de $\langle \sigma_j + \sum_{l_1<l_2<\cdots<l_j} Z_{l_1}\cdots Z_{l_j},\ j \in [1;v]\ \rangle$ pour n'importe quel ordre Lex vérifiant $\underline{Z}_v > \underline{\sigma}_v$, alors $(\text{WARING}_v(Q,\emptyset), G)$ est une base de Gröbner de $\langle \text{SYNSYM}_v\rangle$ pour un ordre Lex avec $\underline{Z}_v > \underline{S} > \underline{\sigma}_v$ (cela provient du premier critère de Buchberger, cf. [CLO97]). Alors, d'après le théorème d'élimination 1.3.3, $\text{WARING}_v(Q,\emptyset)$ est une base de Gröbner de $\langle \text{SYNSYM}_v\rangle \cap \mathbb{F}_2[\underline{\sigma}_v, \underline{S}]$ pour un ordre Lex tel que $\underline{S} > \underline{\sigma}_v$. Le fait que $\text{WARING}_v(Q,\emptyset)$ soit premier provient du théorème 1.3.4 page 15.

La seconde égalité est évidente, et provient de la relation :

$$\langle \text{NEWTON}_v\rangle = \langle S_i - f_i(\sigma_1,\ldots,\sigma_v),\ i \in [1;n], S_i - f_{n+i}(\sigma_1,\ldots,\sigma_v),\ i \in [1;v]\rangle.$$

□

Remarque. Pour le décodage formel, nous cherchons à obtenir des formules linéaires en les σ_j. Il se peut qu'en considérant un système plus petit (c'est-à-dire avec plus de solutions) $\text{WARING}_t(E_1,E_2) \subset \text{NEWTON}_t$ nous obtenions déjà de telles formules (c'est vrai en pratique mais non prouvé en général), l'avantage du système $\text{WARING}_t(E_1,E_2)$ étant que la phase d'élimination des fonctions puissances inconnues est déjà réalisée. Or, les calculs sont d'autant plus rapides que le nombre de variables (et donc que $E_1 \cup E_2$) est petit.

Contrairement au cas des idéaux de dimension zéro, l'idéal $\langle \text{WARING}_t(E_1,E_2)\rangle$ ne contient pas toutes les équations $S_{2i \bmod n} = S_i^2$, nous ne pouvons donc pas nous contenter d'un représentant des syndromes par classe cyclotomique. Cependant cette propriété reste vraie pour les syndromes pairs : si $i, 2i \in E_1$ alors l'idéal $\langle \text{WARING}_t(E_1,\emptyset)\rangle$ contient le polynôme linéaire $S_{2i} + S_i^2$, donc il suffit de considérer dans E_1 des syndromes impairs. De la même manière, si $i, 2i-n \in E_2$ alors l'idéal $\langle \text{WARING}_t(\emptyset,E_2), \sigma_v T + 1\rangle$ (où nous avons ajouté une condition spécifiant que σ_v était non nulle, ce qui n'est pas réducteur car il est facile de déterminer le poids de l'erreur) contient le polynôme linéaire $S_{2i-n} + S_i^2$, donc il suffit de considérer dans E_2 des syndromes pairs.

Comme l'idéal $\text{WARING}_v(E_1,E_2)$ est toujours de dimension v, il faut au minimum v syndromes (impairs dans E_1 ou pairs dans E_2), et en pratique $v+1$ ou $v+2$ suffisent.

Remarque. Comme $\text{WARING}_v(E_1,E_2) \subset \text{NEWTON}_v \cap \mathbb{F}_2[\underline{\sigma}_v, \underline{S}]$, pour tout syndrome $\underline{S}^*$ les solutions de $\text{WARING}_v(E_1,E_2)(\underline{S}^*)$ contiennent nécessairement la ou les erreurs à distance au plus v de l'erreur. Si $\text{WARING}_v(E_1,E_2)(\underline{S}^*)$ n'a aucune solution, c'est qu'il n'existe aucun mot du code à distance au plus v de l'erreur.

La section suivante illustre ces algorithmes de décodage sur les exemples des codes à résidus quadratiques de longueur 31 et 41, et explicite les limites de ce type de décodage.

6.3.4 Exemples pratiques, efficacité des algorithmes

Cas du code RQ de longueur 31. Les carrés impairs modulo 31 sont $\{1, 5, 7, 9, 19, 25\}$ et il faut prendre au moins 4 syndromes impairs[4] pour avoir des relations linéaires en σ_2 et σ_3. Nous considérons donc le système $\text{WARING}_3(\{1,5,7,9\}, \emptyset)(\underline{S^*})$ qui est radical[5] pour toute spécialisation d'un syndrome correspondant à une erreur de poids 3. Le système est :

$$\begin{cases} S_9 + \sigma_1^9 + \sigma_2\sigma_1^7 + \sigma_3\sigma_1^6 + \sigma_2^2\sigma_1^5 + \sigma_2\sigma_3\sigma_1^4 + \sigma_2^4\sigma_1 + \sigma_3^2\sigma_2\sigma_1 + \sigma_2^3\sigma_3 + \sigma_3^3, \\ S_7 + \sigma_1^7 + \sigma_2\sigma_1^5 + \sigma_3\sigma_1^4 + \sigma_2\sigma_3\sigma_1^2 + \sigma_2^3\sigma_1 + \sigma_3^2\sigma_1 + \sigma_2^2\sigma_3, \\ S_5 + \sigma_1^5 + \sigma_2\sigma_1^3 + \sigma_3\sigma_1^2 + \sigma_2^2\sigma_1 + \sigma_3\sigma_2, \\ S_1 + \sigma_1 \end{cases}$$

La base de Gröbner pour l'ordre Lex $\sigma_3 > \sigma_2 > \sigma_1 > S_9 > S_7 > S_5 > S_1$ est donnée figure 6.4. Nous obtenons bien des polynômes linéaires en les σ_j, en particulier les relations obtenues à la main dans [RYT90]. Remarquons que la formule linéaire (15) de [RYT90] pour σ_2 est fausse, la formule correcte étant :

$\sigma_2(S_9S_7S_1^3 + S_9S_1^{10} + S_7^2S_5 + S_7^2S_1^5 + S_7S_5^2S_1^2 + S_7S_1^{12} + S_5^2S_1^9 + S_1^{19}) + S_9^2S_1^3 + S_9S_1^{12} + S_7^3 + S_7^2S_5S_1^2 + S_7S_5S_1^9 + S_7S_1^{14} + S_5^4S_1 + S_5^2S_1^{11}$

Le système triangulaire T suivant résout le problème du décodage pour le code à résidus quadratiques de longueur 31, pour les mots de poids 2 et 3 :

$$T = \begin{cases} \sigma_1 + S_1, \\ \sigma_3(\sigma_2 + S_1^2) + \sigma_2^2S_1 + \sigma_2S_1^3 + S_5 + S_1^5, \\ \sigma_2(S_9S_7S_1^3 + S_9S_1^{10} + S_7^2S_5 + S_7^2S_1^5 + S_7S_5^2S_1^2 + S_7S_1^{12} + S_5^2S_1^9 + S_1^{19}) + \\ S_9^2S_1^3 + S_9S_1^{12} + S_7^3 + S_7^2S_5S_1^2 + S_7S_5S_1^9 + S_7S_1^{14} + S_5^4S_1 + S_5^2S_1^{11} \end{cases}$$

Si l'équation en σ_3 est nulle, alors l'erreur est de poids 0 ou 1 et la première équation donne la valeur de σ_1. L'avantage de ces formules est qu'elles sont de taille beaucoup plus raisonnable que les formules obtenues à partir du système des équations de Newton, et permettent aussi de décoder.

Cas du code RQ de longueur 41 Dans le cas du code RQ de longueur 41, de type $[41, 21, 9]$, on a $Q_{41} = \{1, 5, 9, 21, 23, 25, 31, 33, 37, 39\} \cup \{2, 4, 8, 10, 16, 18, 20, 32, 36, 40\}$. Ce code corrigeant 4 erreurs il est nécessaire de considérer de grands syndromes. Or le calcul de la base lexicographique de $\text{WARING}_4(\{1,5,9,21\}, \emptyset)$ n'aboutit pas, et le calcul de la base lexicographique du système $\text{WARING}_3(\{1,5,9\}, \emptyset)$ donne des formules de degré 5 en σ_3.

Considérons alors le système $\text{WARING}_4(\{1,5,9\}, \{36,40\})$ dans lequel on a divisé les équations de Waring inverses par σ_4 (on s'intéresse alors uniquement aux

[4] Les syndromes intéressants sont les plus petits, car ils donnent des équations de petit degré (l'équation de Waring f_i définissant S_i est de degré i).

[5] Le système contient les équations $F_1 = (Z_1^{32} + Z_1) + (Z_2^{32} + Z_2) + (Z_3^{32} + Z_3), F_5 = Z_1^4(Z_1^{32} + Z_1) + Z_2^4(Z_2^{32} + Z_2) + Z_3^4(Z_3^{32} + Z_3), F_9 = Z_1^8(Z_1^{32} + Z_1) + Z_2^8(Z_2^{32} + Z_2) + Z_3^8(Z_3^{32} + Z_3)$ et nous avons la relation $F_9 + (Z_1 + Z_2)^4F_5 + (Z_1Z_2)^4F_1 = (Z_3^{32} + Z_3)(Z_3 + Z_1)^4(Z_3 + Z_2)^4$ et donc $Z_i^{32} + Z_i \in \text{WARING}_3$ pour $i \in [1, 2, 3]$

$$
\begin{cases}
\sigma_{\mathbf{3}}^{\mathbf{3}} + \sigma_2^3 S_1^3 + \sigma_2(S_7 + S_5 S_1^2) + S_9 + S_5 S_1^4, \\
\sigma_{\mathbf{3}}^{\mathbf{2}} S_5 + \sigma_3 S_1^8 + \sigma_2^2 S_7 + \sigma_2 S_9 + S_9 S_1^2 + S_5^2 S_1, \\
\sigma_{\mathbf{3}}^{\mathbf{2}} S_1 + \sigma_3 S_1^4 + \sigma_2^2 S_1^3 + \sigma_2 S_5 + S_7 + S_1^7, \\
\sigma_{\mathbf{3}}(\sigma_2 + S_1^2) + \sigma_2^2 S_1 + \sigma_2 S_1^3 + S_5 + S_1^5, \\
\sigma_{\mathbf{3}} S_9 S_1^2 + \sigma_2^2 (S_5 + S_1^5)^2 + \sigma_2(S_9 S_1^3 + S_5^2 S_1^2 + S_5 S_1^7) + S_7^2 + S_7 S_1^7 + S_5^2 S_1^4 + S_1^{14}, \\
\sigma_{\mathbf{3}} S_7 + \sigma_2 S_1^8 + S_9 S_1 + S_5^2, \\
\sigma_{\mathbf{3}}(S_5 + S_1^5)^2 + \sigma_2^3(S_7 + S_1^7) + \sigma_2^2(S_9 + S_1^9) + \sigma_2(S_7 S_1^4 + S_5 S_1^6) + S_9 S_1^4 + S_5 S_1^8, \\
\sigma_{\mathbf{3}}(S_5 S_1 + S_1^6) + \sigma_2^2 S_5 + \sigma_2^2 S_1^5 + \sigma_2 S_7 + \sigma_2 S_1^7 + S_7 S_1^2 + S_5 S_1^4, \\
\sigma_{\mathbf{2}}^{\mathbf{4}}(S_7 + S_1^7) + \sigma_2^3(S_9 + S_1^9) + \sigma_2^2(S_5 S_1^6 + S_1^{11}) + \sigma_2(S_7 S_5 S_1 + S_1^{13}) + S_9 S_1^6 + S_5^3 + \\
S_5^2 S_1^5 + S_1^{15}, \\
\sigma_{\mathbf{2}}^{\mathbf{3}}(S_7 + S_1^7)^2 + \sigma_2^2(S_9(S_7 + S_1^7) + S_5^2 S_1^6 + S_1^{16}) + \sigma_2(S_9 S_1^9 + S_7^2 S_1^4 + S_7 S_1^{11} + S_1^{18}) + \ldots, \\
\sigma_{\mathbf{2}}^{\mathbf{3}}(S_5 + S_1^5) + \sigma_2^2(S_7 + S_1^7) + S_7 S_1^4 + S_5^2 S_1 + S_5 S_1^6 + S_1^{11}, \\
\sigma_{\mathbf{2}}^{\mathbf{2}}(S_9 S_5 + S_5^2)(S_5 + S_1^5) + \sigma_2(S_9 S_7 S_1 + S_9 S_1^8 + S_7^2 S_1^3 + S_7 S_5 S_1^5 + S_5^2 S_1^7 + S_5 S_1^{12}) + \ldots, \\
\sigma_{\mathbf{2}}^{\mathbf{2}}(S_7^3 + S_1^{21}) + \sigma_2(S_9 S_7^2 + S_9 S_5 S_1^9 + S_9 S_1^{14} + S_7^2 S_1^9 + S_7 S_5 S_1^{11} + S_7 S_1^{16} + S_5^3 S_1^8 + \\
S_5^2 S_1^{13}) + \ldots, \\
\sigma_{\mathbf{2}}^{\mathbf{2}}(S_7 S_5 + S_1^{12}) + \sigma_2(S_9 S_1^5 + S_7^2 + S_5^2 S_1^4 + S_5 S_1^9) + S_9 S_5 S_1^2 + S_7^2 S_1^2 + S_7 S_1^9 + S_5^3 S_1, \\
\sigma_{\mathbf{2}}^{\mathbf{2}} S_7 S_1 + \sigma_2^2 S_1^8 + \sigma_2 S_9 S_1 + \sigma_2 S_7 S_1^3 + \sigma_2 S_5^2 + \sigma_2 S_1^{10} + S_9 S_1^3 + S_7 S_5 + S_7 S_1^5 + S_5^2 S_1^2, \\
\sigma_{\mathbf{2}}^{\mathbf{2}}(S_5^{17} + S_1^{85}) + \sigma_2(S_9^9 S_5 S_1 + S_9^9 S_1^6 + S_9^8 S_7^2 S_1 + S_9^8 S_5^3 + S_9^5 S_7^6 + S_9^5 S_7 S_5^7 + \ldots) + \ldots, \\
\sigma_{\mathbf{2}}^{\mathbf{2}}(S_5^{15} S_1 + S_1^{76}) + \sigma_2(S_9^8 S_5 S_1 + S_9^8 S_1^6 + S_9^7 S_7^2 S_1 + S_9^7 S_5^3 + S_9^4 S_7^6 + S_9^4 S_7 S_5^7 + \ldots) + \ldots, \\
\sigma_{\mathbf{2}}^{\mathbf{2}}(S_5^{13} S_1^2 + S_1^{67}) + \sigma_2(S_9^7 S_5 S_1 + S_9^7 S_1^6 + S_9^6 S_7^2 S_1 + S_9^6 S_5^3 + S_9^3 S_7^6 + S_9^3 S_7 S_5^7 + \ldots) + \ldots, \\
\sigma_{\mathbf{2}}^{\mathbf{2}}(S_5^{11} S_1^3 + S_1^{58}) + \sigma_2(S_9^6 S_5 S_1 + S_9^6 S_1^6 + S_9^5 S_7^2 S_1 + S_9^5 S_5^3 + S_9^2 S_7^6 + S_9^2 S_7 S_5^7 + \ldots) + \ldots, \\
\sigma_{\mathbf{2}}^{\mathbf{2}}(S_5^9 S_1^4 + S_1^{49}) + \sigma_2(S_9^5 S_5 S_1 + S_9^5 S_1^6 + S_9^4 S_7^2 S_1 + S_9^4 S_5^3 + S_9 S_7^6 + S_9 S_7 S_5^7 + \ldots) + \ldots, \\
\sigma_{\mathbf{2}}^{\mathbf{2}}(S_5^7 S_1^5 + S_1^{40}) + \sigma_2(S_9^4 S_5 S_1 + S_9^4 S_1^6 + S_9^3 S_7^2 S_1 + S_9^3 S_5^3 + S_7^6 + S_7^3 S_5^2 S_1^{11} + \ldots) + \ldots, \\
\sigma_{\mathbf{2}}^{\mathbf{2}}(S_5^5 S_1^6 + S_1^{31}) + \sigma_2(S_9^3 S_5 S_1 + S_9^3 S_1^6 + S_9^2 S_7^2 S_1 + S_9^2 S_5^3 + S_9 S_5^4 S_1^4 + S_9 S_1^{24} + \ldots) + \ldots, \\
\sigma_{\mathbf{2}}^{\mathbf{2}}(S_5^3 S_1^7 + S_1^{22}) + \sigma_2(S_9^2 S_5 S_1 + S_9^2 S_1^6 + S_9 S_7^2 S_1 + S_9 S_5^3 + S_7^3 S_1^3 + S_7^2 S_1^{10} + \ldots) + \ldots, \\
\sigma_{\mathbf{2}}^{\mathbf{2}}(S_5 + S_1^5) S_1^8 + \sigma_2(S_9 S_5 S_1 + S_9 S_1^6 + S_7^2 S_1 + S_7 S_5 S_1^3 + S_5^3 + S_5^2 S_1^5) + S_7(S_7 S_1^3 + \\
S_5^2 + S_5 S_1^5 + S_1^{10}), \\
\sigma_{\mathbf{2}}(S_9^2 S_5 S_1^2 + S_9^2 S_1^7 + S_9 S_7^2 S_1^2 + S_9 S_1^{16} + S_7^3 S_1^4 + S_7^2 S_5^2 S_1 + S_7^2 S_5 S_1^6 + \ldots) + \ldots, \\
\sigma_{\mathbf{2}}(S_9 S_7^2 S_5 + S_9 S_5^2 S_1^9 + S_7^4 + S_7^3 S_5 S_1^2 + S_7^3 S_1^7 + S_7^2 S_1^{14} + S_7 S_5^2 S_1^{11} + S_7 S_1^{21} + S_5^4 S_1^8 + \\
S_1^{28}) + \ldots, \\
\sigma_{\mathbf{2}}(S_9 S_7 S_5 S_1 + S_9 S_1^{13} + S_7^3 S_1 + S_7 S_5^3 + S_5^2 S_1^{12} + S_5 S_1^{17}) + S_9^2 S_1^6 + S_9 S_5 S_1^{10} + S_7^2 S_5^2 + \ldots, \\
\sigma_{\mathbf{2}}(S_9 S_7 S_1^3 + S_9 S_1^{10} + S_7^2 S_5 + S_7^2 S_1^5 + S_7 S_5^2 S_1^2 + S_7 S_1^{12} + S_5^2 S_1^9 + S_1^{19}) + S_9^2 S_1^3 + S_9 S_1^{12} + \ldots, \\
\sigma_{\mathbf{2}}(S_9 S_5 S_1^{10} + S_9 S_1^{15} + S_7^3 S_1^3 + S_7^2 S_5^2 + S_7^2 S_5 S_1^5 + S_7 S_5 S_1^{12} + S_5^3 S_1^9 + S_5^2 S_1^{14}) + S_9^2 S_5 S_1^3 + \\
S_9^2 S_1^8 + S_7^3 S_5 + S_7 S_5^2 S_1^9 + S_7 S_5 S_1^{14} + S_7 S_1^{19} + S_5^5 S_1 + S_5^4 S_1^6, \\
\sigma_{\mathbf{2}}(S_7^4 S_1^3 + S_7^3 S_5^2 + S_7^3 S_5 S_1^5 + S_7^3 S_1^{10} + S_7^2 S_1^{17} + S_7 S_1^{24} + S_5 S_1^{26} + S_1^{31}) + S_9^2 S_7 S_5 S_1^3 + \ldots, \\
\sigma_{\mathbf{1}} + S_1, \\
\mathbf{S_9^3} S_5 S_1^3 + S_9^3 S_1^8 + S_9^2 S_7^2 S_1^3 + S_9^2 S_7 S_5 S_1^5 + S_9^2 S_7 S_1^{10} + S_9^2 S_5^3 S_1^2 + S_9^2 S_5^2 S_1^7 + S_9^2 S_1^{17} + \ldots,
\end{cases}
$$

FIG. 6.4 – Base Lex du système $\text{WARING}_3(\{1,5,7,9\},\emptyset)$ pour le code RQ [31,16,7]

mots de poids exactement 4) :

$$\begin{cases} S_1 + \sigma_1, \\ S_5 + \sigma_1^5 + \sigma_1^3\sigma_2 + \sigma_1^2\sigma_3 + \sigma_2^2\sigma_1 + \sigma_2\sigma_3 + \sigma_1\sigma_4, \\ S_9 + \sigma_1^9 + \sigma_1^6\sigma_3 + \sigma_2^4\sigma_1 + \sigma_2^3\sigma_3 + \sigma_3^3 + \sigma_1\sigma_4^2 + \sigma_2^2\sigma_1\sigma_4 + \sigma_1^5\sigma_4 + \sigma_1^7\sigma_2 + \sigma_3^2\sigma_2\sigma_1 + \\ \qquad \sigma_1^2\sigma_4\sigma_3 + \sigma_1^4\sigma_2\sigma_3 + \sigma_1^5\sigma_2^2, \\ S_{36}\sigma_4^5 + \sigma_3^5 + \sigma_4^2\sigma_2^2\sigma_3 + \sigma_4^3\sigma_2\sigma_1 + \sigma_4^3\sigma_3 + \sigma_4\sigma_3^3\sigma_2 + \sigma_4^2\sigma_3^2\sigma_1, \\ S_{40}\sigma_4 + \sigma_3, \end{cases}$$

Nous pouvons éliminer les variables σ_1 et σ_3 avec la première et la dernière équation. Il reste donc un système de 3 équations, g_5, g_9 et g_{36} et 7 variables, ordonnées $\sigma_2 > \sigma_4 > S_{40} > S_{36} > S_9 > S_5 > S_1$, et dont la base de Gröbner est toujours incalculable. Considérons alors séparément les deux systèmes $\{g_5, g_9\}$ et $\{g_5, g_{36}\}$, la base de Gröbner de chacun de ces systèmes est calculable, ce qui donne deux polynômes en σ_4. Une base lexicographique de $\{g_5, g_9\}$ fournit la formule de degré 5 en σ_4 :

$P_{5,9} = (S_{40}^5S_5 + S_{40}^5S_1^5 + S_{40}^4S_1^4 + S_{40}^3S_1^3)\sigma_4^5 + (S_{40}^4S_5S_1^3 + S_{40}^4S_1^8 + S_{40}^3S_1^7 + S_{40}^2S_5S_1 + S_{40}S_1^5 + S_1^4)\sigma_4^4 + (S_{40}^3S_9S_1^2 + S_{40}^3S_5^2S_1 + S_{40}^2S_1^{10})\sigma_4^3 + (S_{40}^2S_9S_5 + S_{40}^2S_5^2S_1^4 + S_{40}^2S_5S_1^9 + S_{40}^2S_1^{14} + S_{40}S_9S_1^4 + S_{40}S_5S_1^8 + S_{40}S_1^{13} + S_5^2S_1^2)\sigma_4^2 + (S_{40}S_9S_5S_1^3 + S_{40}S_5^3S_1^2 + S_{40}S_5S_1^{12} + S_{40}S_1^{17} + S_9S_1^7 + S_5^2S_1^6 + S_5S_1^{11} + S_1^{16})\sigma_4 + S_9^2S_1^2 + S_9S_1^{11} + S_5^4 + S_5^2S_1^{10} + S_5S_1^{15} + S_1^{20}$

et une base lexicographique de $\{g_5, g_{36}\}$ donne la formule de degré 4 pour σ_4 (en fait de degré 7 avec σ_4^3 en facteur) :

$P_{5,36} = (S_{40}^{10}S_1^2 + S_{40}^9S_1 + S_{40}^8 + S_{40}^4S_{36}S_1 + S_{40}^3S_{36} + S_{36}^2S_1^2)\sigma_4^4 + (S_{40}^8S_1^4 + S_{40}^7S_1^3 + S_{40}^5S_1 + S_{40}^4 + S_{40}^3S_{36}S_1^4 + S_{40}S_{36}S_1^2)\sigma_4^3 + (S_{40}^6S_5S_1 + S_{40}^5S_5 + S_{40}S_{36}S_1^6 + S_{36}S_1^5)\sigma_4^2 + (S_{40}^4S_5S_1^3 + S_{40}^4S_1^8 + S_{40}^3S_1^7 + S_{40}^2S_5S_1 + S_{40}S_1^5 + S_1^4)\sigma_4 + S_{40}^2S_5^2 + S_{40}^2S_1^{10} + S_{40}S_5S_1^4 + S_{40}S_1^9 + S_5S_1^3 + S_1^8$

Nous obtenons alors une formule linéaire pour σ_4 en calculant le pgcd de $P_{5,9}$ et $P_{5,36}$. Nous avons calculé ce pgcd formel en utilisant le logiciel Magma, le résultat final est un polynôme linéaire en σ_4, de degré total 170 en S_1, S_5, S_9, S_{36} et S_{40} contenant 29828 termes. La base de Gröbner formelle contient donc bien une formule linéaire, mais tellement grande que son évaluation sur chaque syndrome est beaucoup trop coûteuse.

Pré-calcul d'une formule L'idée du décodage formel est de pré-calculer des formules donnant chaque σ_j en fonction des syndromes $\underline{S}$, que l'on pourra évaluer rapidement dans la phase de décodage en ligne. Nous avons vu que le système des équations de Newton formel donne bien des formules, mais le coût de l'évaluation de ces formules les rend inutilisables. L'exemple du système $\textsc{Waring}_4(\{1, 5, 9, 21\}, \emptyset)$ montre également que la formule linéaire que l'on peut obtenir est bien trop grande pour être utile. Cependant, si nous considérons la formule pgcd($P_{5,9}$,$P_{5,36}$), l'évaluation consiste à d'abord spécialiser les syndromes, puis calculer le pgcd des polynômes en une variable σ_4 à coefficients dans $\mathbb{F}_{2^m}$. L'évaluation sur cet exemple coûte grossièrement 9 divisions et 134 multiplications dans le corps $\mathbb{F}_{2^{20}}$, plus un calcul de pgcd de polynômes à coefficients dans ce même corps fini (ici, il faut compter

28 multiplication et 9 divisions). Ainsi, l'évaluation de cette formule est beaucoup plus rapide que celle donnée dans la base de Gröbner du système des équations de Newton.

De la même manière dans [RTCY92], les auteurs donnent une formule de degré 4 pour σ_2 et des formules linéaires pour σ_1, σ_3, et σ_4. Ils calculent chacune des 4 solutions $(\sigma_1^i, \sigma_2^i, \sigma_3^i, \sigma_4^i)$ et déterminent la bonne (une seule donne un polynôme localisateur correspondant bien à une erreur). Remarquons que la résolution d'une équation de degré 4 sur le corps $\mathbb{F}_{2^{20}}$ n'est pas triviale, la moitié de l'article y est consacrée. Si, au lieu d'avoir une équation de degré 4 en σ_2 on prenait une équation de degré 4 en σ_4, on pourrait chercher ses racines avec la méthode du "Chien search" (cf. [MS77, p. 276], σ_4 est le produit des localisateurs Z_i, c'est donc une racine $41^{\text{ème}}$ de l'unité). On peut également sur cet exemple chercher les racines de $P_{5,9}$ et $P_{5,36}$ par la méthode du "Chien search" et comparer les racines.

Nous allons voir dans la section suivante que l'on peut remplacer le calcul de la base de Gröbner formelle par la trace d'un calcul spécialisé, qui peut être vu comme une manière rapide d'évaluer la base de Gröbner formelle.

6.4 Résultats pratiques, trace du pré-calcul

Dans cette section nous décrivons les systèmes utilisés en pratique, et la méthode de la trace du pré-calcul qui permet d'accélérer les calculs et de ne pas recalculer une base de Gröbner pour chaque mot à décoder. Nous donnons de nombreux exemples pratiques, avec les temps de décodage.

6.4.1 Systèmes utilisés en pratique pour le décodage en ligne

L'exemple du code RQ 41 indique que la méthode la plus efficace semble être celle du décodage en ligne. Pour chaque mot e de poids v on construit le système $\textsc{Newton}_v(\underline{S}^*)$ et on calcule sa base de Gröbner sur $\mathbb{F}_{2^m}$. Le système a une unique solution, est radical et la base de Gröbner réduite est de la forme (voir proposition 1.3.1 page 14) :

$$\{\sigma_1 + \sigma_1^*, \ldots, \sigma_v + \sigma_v^*\}$$

où les σ_i^* sont les coefficients du polynôme localisateur.

Remarque. Le système $\textsc{Newton}_v(\underline{S}^*)$ comporte beaucoup d'équations et de variables à éliminer (les S_k, $k \notin Q$). Nous avons donc intérêt à éliminer à l'avance ces fonctions puissances inconnues (et à remplacer σ_1 par S_1), et dès que possible à prendre un sous-ensemble des équations (en effet, le système $\textsc{Newton}_v(\underline{S}^*)$ est surdéterminé car il comporte de nombreuses équations redondantes, et l'on a intérêt à en extraire une suite dont le comportement pour le calcul de la base de Gröbner est le plus régulier possible, i.e. pour laquelle il y a le moins de réduction à zéro possible). Si l'on élimine tous les $S_j, j < i$ de l'équation de Newton définissant S_i, on

obtient le système $\text{WARING}_v(\{i\}, \emptyset)$, et si l'on élimine tous les $S_j, j > i$ de l'équation définissant S_{i+v} on obtient le système $\text{WARING}_v(\emptyset, \{i\})$. Remarquons qu'il est inutile d'éliminer les syndromes S_j, $j \in Q$ puisque ceux-ci vont être spécialisés. Nous utiliserons donc un sous-ensemble du système NEWTON_v, encore noté $\text{WARING}_v(E_1, E_2)$, et obtenu en éliminant uniquement les fonctions puissances inconnues et σ_1. Nous essayons d'extraire un système qui possède une unique solution, et pour lequel le degré maximal atteint au cours du calcul de la base de Gröbner soit minimal : c'est pour ce système que nous aurons une efficacité maximale. Nous choisissons les équations de degré minimal une fois éliminées les fonctions puissances inconnues.

Remarque. En calculant une base de Gröbner G_w de $\text{WARING}_w(E_1, E_2)(\underline{S}^*)$, il se peut que l'on obtienne $G_w = \{1\}$. Cela signifie qu'il n'y a aucun mot du code à distance $\leq w$ du mot que l'on cherche à décoder, et que l'on a détecté une erreur de poids $w + 1 \leq v \leq d - 1$. Il suffit alors d'augmenter w et de recalculer une base de $\text{WARING}_w(\underline{S}^*)$.

Exemple. Pour le code à résidus quadratiques de longueur 31, après recherche des équations de plus bas degré, nous utilisons le système suivant :

$$\text{WARING}_3(\{10\}, \{4, 16\}) = \begin{cases} S_{10} + S_9 S_1 + S_8 \sigma_2 + S_7 \sigma_3, \\ S_1^2 S_8 + \sigma_2 S_1 S_7 + S_1 S_9 + \sigma_2 \sigma_3 S_5 + \sigma_3^2 S_4 + \sigma_3 S_7, \\ \sigma_2 S_1 S_{19} + \sigma_3 S_1 S_{18} + \sigma_2^2 S_{18} + \sigma_2 S_{20} + \sigma_3^2 S_{16} + \sigma_3 S_{19} \end{cases}$$

Le degré maximal atteint au cours du calcul de la base de Gröbner est 3 (ou 2 pour des erreurs de poids au plus 1).

- l'erreur $1 + x + x^2$ donne le système $\{\sigma_1 + \alpha^2 + \alpha + 1, \sigma_2 + \alpha^3 + \alpha^2 + \alpha, \sigma_3 + \alpha^3\}$ de dimension 0, degré 1.
- l'erreur $1 + x$ donne le système $\{\sigma_1 + \alpha + 1, \sigma_2 + \alpha, \sigma_3\}$ toujours à une solution,
- l'erreur 1 donne le système $\{\sigma_1 + 1, \sigma_2 + \sigma_3\}$ de dimension 1, degré 1.
- Enfin l'erreur 0 (i.e. pas d'erreur) donne le système $\{0\}$. On a $\text{NEWTON}_3(0) = \{\sigma_1, \sigma_3\}$ de dimension 1, degré 1.
- l'erreur $1 + x + x^2 + x^3$ donne le système $\{1\}$, il n'y a donc aucune erreur de poids ≤ 3 correspondant.

6.4.2 Décodage au-delà de la distance minimale

Remarquons que l'ensemble des solutions du système $\text{NEWTON}_v(\underline{S}^*)$ est exactement l'ensemble des erreurs de poids au plus v qui ont pour syndrome $\underline{S}^*$. Il suffit donc qu'il existe une unique erreur de poids au plus v et de syndrome $\underline{S}^*$ pour que cette erreur soit décodable, même si v est plus grand que la capacité de correction t du code $\mathcal{C}$. Et dans le cas où plusieurs erreurs de poids au plus v ont le même syndrome, les méthodes présentées précédemment permettent de faire du décodage en liste jusqu'au poids v. La taille de la liste est inconnue, et peut être très grande. Remarquons aussi que la complexité du calcul de la base de Gröbner croît avec la taille de cette liste.

Conséquences.

- On peut ainsi décoder au-delà de la distance minimale du code pour certains mots,
- Si l'on obtient plusieurs solutions pour NEWTON_u, c'est que $d \leq 2u$, on a ainsi une borne supérieure sur la distance minimale d'un code.
- Plus précisément, si l'on trouve une solution de poids v_1 et une solution de poids v_2 qui ont le même syndrome, c'est que $d \leq v_1 + v_2$.

Illustrons cette propriété avec le code RQ $[31, 16, 7]$:

- l'erreur $1 + x + x^2 + x^3$ donne le système $\text{NEWTON}_4(\underline{S}^*) = \{\sigma_1 + \alpha^3 + \alpha^2 + \alpha + 1, \sigma_2 + \alpha^4 + \alpha + 1, \sigma_3 + \alpha^4 + \alpha^2 + \alpha + 1, \sigma_4 + \alpha^3 + \alpha\}$ de dimension 0, degré 1. On peut décoder cette erreur de poids 4,
- l'erreur $1+x+x^2+x^4$ donne un système $\text{NEWTON}_4(\underline{S}^*)$ de dimension 0, degré 4 dont toutes les solutions sont de poids 4 : elles correspondent aux erreurs $1+x+x^2+x^4$, $x^9+x^{12}+x^{16}+x^{21}$, $x^5+x^{13}+x^{25}+x^{26}$, et $x^8+x^{11}+x^{18}+x^{24}$. Cela implique que la distance minimale du code est $d \leq 8$. Or, d'après la proposition 2.2.8, on sait que $d^2 \geq 31$, *i.e.* $d \geq 6$ et d'après la proposition 2.2.7, d est impaire donc on retrouve $d = 7$.

Nous pouvons donc recalculer la distance minimale du code RQ 31, et décoder certaines erreurs de poids plus grand que la capacité de correction de ce code. Nous avons effectué une recherche exhaustive sur toutes les erreurs de poids 4 (il y en a 31465). Comme le montre la table 6.2, 31% de ces erreurs de poids 4 sont décodables (i.e. il y a un unique mot du code à distance au plus 4), et pour 4.9% des erreurs de poids 4 l'ensemble des solutions de $\text{NEWTON}_4(\underline{S}^*)$ contient une solution de poids 4 et une de poids 3. La liste est toujours de taille au plus 5. On peut dans tous les cas renvoyer une liste de solutions de taille au plus 5.

nombre et % d'erreurs de poids 4 pour lesquelles $\text{NEWTON}_4(\underline{S}^*)$ possède n_3 (resp. n_4) solutions de poids 3 (resp. 4)									
(n_3, n_4)	(0,1)	(0,2)	(1,1)	(0,3)	(1,2)	(0,4)	(1,3)	(0,5)	(1,4)
	9 765	9 300	1 550	4 650	1 860	1 860	1 395	465	620
	31%	29,6%	4,9%	14,8%	5,9%	5,9%	4,4%	1,5%	2%

TAB. 6.2 – Décodage des erreurs de poids 4 pour le code RQ [31,16,7].

Pour le poids 5 il y a 169911 erreurs possibles. La table 6.3 donne le nombre et le poids des solutions du système $\text{NEWTON}_5(\underline{S}^*)$. Un point d'interrogation indique que l'idéal est de dimension 1 et que la variable σ_5 est libre. Il est alors possible de trouver le nombre d'erreurs de poids < 5 en substituant $\sigma_5 = 0$ dans la base de Gröbner, puis $\sigma_4 = 0$, etc. Pour connaître exactement le nombre d'erreurs de poids 5 il faudrait essayer pour σ_5 toutes les valeurs α^j possibles $(0 \leq j \leq 30)$, calculer le polynôme localisateur et vérifier qu'il divise bien $x^{31} - 1$. Plus simplement, il est

possible de ne renvoyer que les erreurs de poids ≤ 4 correspondantes. Moins de 2% des erreurs de poids 5 se décodent en une unique erreur de poids 2, pour environ 15% des erreurs de poids 5 on décode en une unique erreur de poids 3, ou une liste de taille au plus 5 d'erreurs de poids 3 et 4, et enfin pour la majorité des erreurs de poids 5 (soit près de 83%) on décode en une liste de taille 6 contenant des erreurs de poids 4 et 5.

dimension, degré de l'idéal	nombre de solutions de poids				TOTAL	%
	$v=2$	$v=3$	$v=4$	$v=5$		
0, 6				6	67518	39.74
0, 6			1	5	48825	28.74
0, 6			2	4	18600	10.95
0, 6			3	3	4650	2.74
0, 6			4	2	930	.55
0, 6			5	1	93	.05
1, 1		1	4	?	465	.27
1, 1		1	3	?	1860	1.09
1, 1		1	2	?	4650	2.74
1, 1		1	1	?	9300	5.47
1, 1		1	0	?	9765	5.75
1, 1	1	0	?	?	3255	1.92

TAB. 6.3 – Décodage exhaustif des erreurs de poids 5 pour le code RQ [31,16,7].

6.4.3 Trace du pré-calcul

Nous utilisons une méthode générale de résolution de systèmes à paramètres, la méthode de la trace du pré-calcul. D'après les théorèmes de spécialisation de la base de Gröbner (Théorèmes 1.3.6 ou 1.3.7), le résultat d'un calcul de base de Gröbner en ligne (sur $\mathbb{F}_{2^m}$) est exactement la spécialisation de la base de Gröbner formelle sur $\mathbb{F}_2$. Nous conjecturons que cette propriété s'étend à l'algorithme : chaque étape du calcul de la base spécialisée est la spécialisation de l'étape correspondante pour la base formelle. En d'autres termes, le calcul de la base de Gröbner se comporte de la même manière pour toutes les spécialisations possibles du syndrome, pour un poids d'erreur donné. Il faut toute fois que le corps de base soit suffisamment gros (par exemple 2^{20}) : en effet, pour le calcul de la base de Gröbner spécialisée, on divise chaque polynôme par son terme de tête (qui est un élément non nul de $\mathbb{F}_{2^m}$), qui est la spécialisation d'un coefficient formel en les S_i. Or, si le corps est trop petit, la probabilité que ce coefficient non nul se spécialise à zéro est grande, alors qu'elle tend vers zéro quand m devient grand. Nous utilisons cette conjecture pour réduire la complexité du décodage en ligne (d'un facteur 1000).

Nous décrivons la méthode dans le cas particulier de l'algorithme F4 [Fau99] que nous avons utilisé en pratique, mais elle s'applique de la même manière à n'importe quel autre algorithme de calcul de base de Gröbner. L'algorithme F4 construit incrémentalement des matrices, et calcule leur forme Row Echelon (voir le chapitre 1 pour plus de détails).

Pour une erreur e_0 de poids v_0, calculons la base de Gröbner de $\text{NEWTON}_v(\underline{S}^*_{e_0})$ en mémorisant toutes les étapes du calcul (la trace du calcul). Nous enregistrons en pratique cette trace sous forme d'un programme C (voir schéma figure 6.4.3). Maintenant, si e est une autre erreur de poids v_0, nous pouvons exécuter le pro-

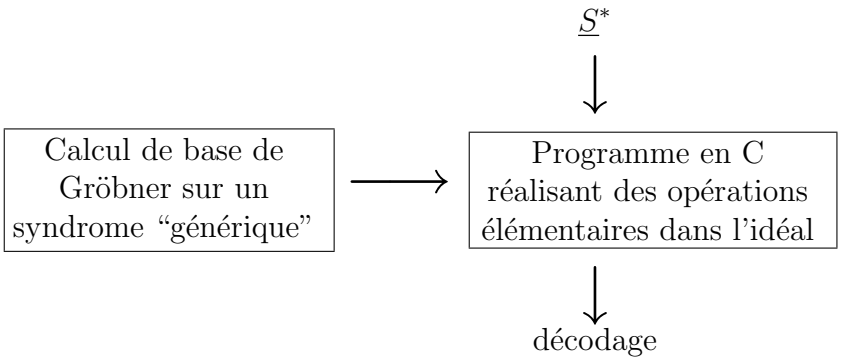


FIG. 6.5 – Méthode de la trace du pré-calcul

gramme généré sur le syndrome de e. Ce programme va construire incrémentalement des matrices, et calculer leur forme Row Echelon, de la même manière que pour e_0. Expérimentalement, le résultat est bien toujours une base de Gröbner du système $\text{NEWTON}_v(\underline{S}^*_e)$, ce qui renforce l'idée que chaque étape du calcul se spécialise bien. Ces considérations justifient l'algorithme suivant :

Algorithme 5

- TRACE DU PRÉ-CALCUL : *calculer une base de Gröbner de* $\text{NEWTON}_v(\underline{S}^*_{e_0})$ *pour une erreur* e_0 *de poids* v *choisie au hasard, et enregistrer (par exemple sous forme de programme C) la trace de toutes les opérations d'algèbre linéaire effectuées,*
- DÉCODAGE : *pour une erreur* e*, exécuter le programme C sur le syndrome* $\underline{S}^*_e$*, on obtient les valeurs des* σ^*_j*.*

Remarque. Si $I_e = \text{NEWTON}_v(\underline{S}^*_e)$, les opérations d'algèbre linéaire effectuées au cours de l'exécution du programme C correspondent, en termes de polynômes, à des opérations dans l'idéal I_e, i.e. tous les polynômes obtenus appartiennent nécessairement à I_e. Il se peut qu'au cours de l'exécution du programme sur $\underline{S}^*_e$ il y ait une division par 0 (si l'évaluation des paramètres pour le premier syndrome donnait un élément non nul de $\mathbb{F}_{2^m}$ et donne zéro sur le deuxième syndrome). Cependant, si l'on ne rencontre pas de telle division par zéro, on obtient des polynômes

$\{h_1, \ldots, h_l\}$ qui appartiennent à l'idéal $I(\underline{S}^*)$, et donc les solutions du système $\{h_1, \ldots, h_l\}$ contiennent les solutions du système $I(\underline{S}^*)$ (avec peut-être d'autres solutions parasites). Si le résultat du programme est $G = \{g_0, \ldots, g_s\}$, alors on a $g_i \in I_e$ pour tout i, et les solutions vérifient $V(\langle g_0, \ldots, g_s \rangle) \supset V(I_e)$. Si le système G a une unique solution, alors cette solution est bien celle de I_e. Si $V(\langle g_0, \ldots, g_s \rangle)$ contient plusieurs solutions, alors la bonne est l'une d'entre elles. En pratique, l'idéal contient toujours une unique solution.

Remarque. Une variante possible de l'algorithme 5 est de calculer en parallèle 1000 bases de Gröbner de $\textsc{Newton}_v(\underline{S}^*_{e_0})$, pour 1000 erreurs e_0 de poids v choisies au hasard. On génère de la même façon que précédemment la trace de ce précalcul, l'avantage étant que pour des corps de plus petite taille, la probabilité qu'un terme de tête se spécialise à zéro pour une erreur particulière, alors qu'il ne s'est spécialisé à zéro pour aucune des 1000 erreurs de référence, devient négligeable.

Par exemple, pour le code RQ 41, pour tous les essais effectués, le programme s'exécute normalement et donne une unique solution. On décode ainsi 4 erreurs en $1.7 \cdot 10^{-5}$ secondes (avec une arithmétique de calcul sur un corps fini). Le programme effectue environ 300 multiplications dans le corps fini (ce qui est proche du nombre de multiplications, environ 200, effectuées dans le cas d'un décodage utilisant la formule $\text{pgcd}(P_{5,9}, P_{5,36})$, voir page 129). Ce programme représente une formule et un ordre d'évaluation de cette formule.

En utilisant cette méthode plutôt que de recalculer une base de Gröbner à chaque erreur, on gagne un facteur d'efficacité non négligeable : c'est le temps utilisé par F4 pour construire les matrices. En effet, le programme C se contente d'effectuer les opérations d'algèbre linéaire. Le gain est semblable à celui que l'on aurait en effectuant une élimination de Gauss en connaissant tous les pivots et les opérations de ligne à effectuer à l'avance.

Remarquons que le programme C ne s'exécute à priori sans division par zéro que si l'erreur e a le même poids v que l'erreur test e_0 (ou éventuellement est de poids $v+1$ ou $v-1$). Pour un code donné $\mathcal{C}$ corrigeant t erreurs, l'algorithme de décodage consiste à pré-calculer t programmes $P_1, \ldots, P_t$, un pour chaque poids possible. Ensuite, pour décoder un mot en ligne, on exécute ces programmes dans l'ordre de P_1 à P_t, jusqu'à ce que l'un des systèmes rende une solution. Ces programmes peuvent être exécutés en parallèle. Un autre avantage est que maintenant, contrairement à un calcul de base de Gröbner générique, nous sommes capables de prédire le temps exact nécessaire au décodage. Comme nous n'utilisons que de l'algèbre linéaire, nous pouvons donner explicitement le nombre d'opérations arithmétiques effectuées dans le corps fini $\mathbb{F}_{2^m}$ pour le décodage d'un mot.

6.4.4 Résultats pratiques, exemples

Nous avons écrit une procédure Maple qui, étant donnés une longueur et l'ensemble de définition d'un code, trouve pour chaque poids le nombre minimal d'équa-

tions nécessaire au décodage.

Pour différents codes, nous donnons ci-dessous la complexité du décodage sous la forme du nombre de multiplications dans le corps fini $\mathbb{F}_{2^m}$ nécessaires à l'exécution du programme C. Ce nombre de multiplications est une bien meilleure mesure de complexité que le temps d'exécution : il ne dépend ni du processeur utilisé, ni de l'implantation de l'arithmétique sur les corps finis utilisés. Ce nombre de multiplications ne comprend pas celles nécessaires au calcul des syndromes, ni à la spécialisation du système et à la recherche des racines du polynôme localisateur.

Codes à résidus quadratiques La table 6.4 donne des algorithmes de décodage pour différents codes à résidus quadratiques. Excepté pour le code de longueur 73, il n'existait auparavant aucun algorithme permettant de décoder ces codes jusqu'à la capacité de correction, et l'algorithme que nous donnons permet également de décoder au-delà de cette capacité de correction. Ainsi, pour le code à résidus quadratiques de longueur 113, le code corrige 7 erreurs et notre algorithme permet de renvoyer une liste de taille au plus 2 pour 8 erreurs.

Codes BCH Nous donnons ci-dessous des exemples de décodages de codes BCH, y compris pour de grandes longueur ($n = 512$). En comparaison, la complexité du décodage des codes BCH jusqu'à la distance construite (pour l'algorithme de Berlekamp-Massey ou de Fitzpatrick) est de $2t^2$ multiplications, où t est le nombre d'erreurs corrigées [FJ98].

Exemple du code BCH($n = 31, \delta = 15$) Considérons le code BCH$(31, 15)$, qui corrige 7 erreurs. Pour 1000 erreurs tirées au hasard nous obtenons les résultats de la table 6.5. La colonne "dim > 0" signifie que l'idéal $\textsc{Newton}_{\text{poids}}$ est de dimension strictement positive. Les chiffres obtenus correspondent assez à la vraie distribution[6] reproduite table 6.5.

Exemple du code BCH($n = 127, \delta = 29$) Pour le code BCH de longueur 127 de distance construite $\delta = 29$, les algorithmes de décodage jusqu'à la distance construite permettent de corriger 14 erreurs. Or, ce code a une distance minimale réelle de 31, et corrige donc 15 erreurs. L'ensemble des syndromes de ce code contient $\tilde{Q}$={3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 33, 35}. Appliquons l'algorithme de décodage "en ligne" à partir du système $\textsc{Newton}_v(\tilde{Q})$. Pour le poids 15, l'idéal possède naturellement une unique solution (on est en dessous de la capacité de correction), et cela reste vrai pour le poids 16 (ce qui signifie que "presque toute" erreur de poids 16 se corrige). La table 6.6 donne le nombre de solutions du système pour 1000 erreurs de poids 17 tirées au hasard.

[6] http://vega.c.oka-pu.ac.jp/~wadayama/fujita/completewd.html

n	d	$\mathbb{F}_{2^m}$	v	nombre de multiplications	nombre de tests donnant i solutions $i=1$	2	3	4	5	7	∞
71	11	2^{35}	3	$2^{6.4}$	10 000						
			4	$2^{9.4}$	10 000						
			5	$2^{11.1}$	10 000						
			6	$2^{14.4}$	9 832	165	3				
			7	$2^{18.2}$	8 904	1 016	58				22
73	13	2^{9}	3	$2^{5.4}$	10 000						
			4	$2^{7.2}$	10 000						
			5	$2^{10.5}$	10 000						
			6	$2^{15.3}$	10 000						
			7	$2^{21.5}$	9 837	163					
			8	$2^{24.3}$	8 530	1 346	108	2	1	13	
89	17	2^{11}	3	$2^{5.1}$	10 000						
			4	$2^{8.9}$	10 000						
			5	$2^{13.3}$	10 000						
			6	$2^{17.3}$	10 000						
			7	$2^{21.2}$	10 000						
			8	$2^{26.2}$	9 996	4					
113	15	2^{28}	3	2^{7}	10 000						
			4	$2^{9.4}$	10 000						
			5	$2^{12.7}$	10 000						
			6	$2^{14.9}$	10 000						
			7	$2^{20.0}$	10 000						
			8	$2^{23.4}$	9 996	4					

TAB. 6.4 – Décodage de codes à résidus quadratiques $[n, \frac{n+1}{2}, d]$.

Exemple des codes BCH($n = 511, \delta = 93$) et ($n = 511, \delta = 91$) Notre algorithme de décodage reste efficace pour certains codes de grande longueur, comme le code BCH de longueur 511 de distance construite $\delta = 93$. Ce code a une distance minimale réelle de 97, et corrige donc 48 erreurs, alors que les algorithmes de décodage jusqu'à la moitié de la distance construite ne corrigent que 46 erreurs. L'ensemble de définition Q est engendré par $Q_0 = \{2i - 1, i \in [1; 47]\}$, et les impairs suivants modulo 511 sont $\{97, 99, 101, 105, 113, 121, 129, \dots\}$. La table 6.7 donne des complexités de décodage pour les deux codes de distance construite 91 et 93. En comparaison, l'algorithme de Berlekamp-Massey aura un coût d'environ 2^{12} multiplications (pour $t = 47$ ou 45 erreurs), ce qui n'est pas très éloigné des nombre de multiplications que nous obtenons pour $t + 1$ erreurs. Nous voyons sur la figure que pour le BCH(511,93) on peut corriger jusqu'à 52 erreurs presque toujours, et jusqu'à 50 erreurs pour le BCH(511,91).

Décodage et nombre de solutions pour 1000 mots de poids 7 à 10						
poids	1 sol.	2 sol.	3 sol.	4 sol.	> 4 sol.	dim > 0
7	1000	0	0	0	0	0
8	921	78	0	0	0	1
9	577	365	34	0	0	24
10	52	314	304	144	0	186

Vraie distribution des mots de poids 7 à 10 (pour 1000 mots).						
poids	1 sol.	2 sol.	3 sol.	4 sol.	5 sol.	6 sol.
7	1000	0	0	0	0	0
8	930.5	69.5	0	0.05	0	0
9	583	386.5	30	0.6	0	0
10	67	412	395	125	0	1

TAB. 6.5 – Décodage du code BCH$(31, 15)$ de type $[31, k, 15]$.

dimension, degré	poids 16	poids 17	Pour 1000 erreurs
0, 2		2	13
0, 3		3	978
0, 3	1	2	8
1, 1	1	?	1

TAB. 6.6 – Décodage du code BCH$(127, 29)$ de type $[127, k, 31]$ pour le poids 17

Comparaison avec un code "aléatoire" Enfin, la table 6.8 compare deux codes de longueur 75 : le code BCH [75,31,7] et un code de type [75,33,7] et d'ensemble de définition $\{1, 3, 25\}$ qui n'appartient à aucune classe connue de codes cycliques. Notre méthode de décodage ne dépend pas d'une classe particulière de codes cycliques, tout code cyclique peut être décodé de cette manière. Nous avons choisi ce code de longueur 75 car il est meilleur que le code BCH correspondant : ils sont de même longueur et de même distance minimale, mais la dimension du code choisi est plus grande que celle du code BCH, et il se comporte mieux au-delà de 4 erreurs.

δ	k	d	t	v_e	nombre de multiplications	nombre de tests donnant $i = 1$ solutions
93	175	95	47	48	$2^{16.2}$	10 000
				49	$2^{16.7}$	100 000
				50	$2^{17.6}$	10 000
				51	$2^{24.0}$	10 000
				52		10 000
91	184	91	45	46	$2^{15.4}$	10 000
				47	$2^{16.8}$	10 000
				48	$2^{16.8}$	10 000
				49	$2^{22.9}$	1 000
				50	$2^{26.7}$	1 000

TAB. 6.7 – Décodage des codes BCH $[511, k, \delta]$ sur $\mathbb{F}_{2^9}$, au-delà de t

	BCH $[75, 31, 7]$				Code aléatoire $[75, 33, 7]$ $Q = \{1, 3, 25\}$			
v	Nombre de multiplications pour v erreurs, $v = 4, 5, 6$:							
4	$2^{11.2}$ à $2^{13.5}$				$2^{11.4}$ à $2^{13.4}$			
5	$2^{11.7}$ à $2^{15.8}$				$2^{12.3}$ à $2^{16.6}$			
6	$2^{12.3}$ à $2^{16.4}$				$2^{18.1}$ à 2^{21}			
	Nombre de tests donnant i solutions							
i	$v = 3$	4	5	6	3	4	5	6
1	10 000	9 970	9 806	9 338	10 000	9 970	9 673	8 923
2	0	25	165	570	0	25	276	811
3	0	0	19	0	0	0	19	111
4	0	5	6	44	0	5	28	70
5	0	0	0	0	0	0	0	5
6	0	0	1	28	0	0	1	51
8	0	0	0	0	0	0	0	2
∞	0	0	3	20	0	0	3	27

TAB. 6.8 – Décodage du code BCH $[75, 31, 7]$ et $\mathcal{C}$ de type $[75, 33, 7]$ avec $Q = \{1, 3, 25\}$.

Conclusions et Perspectives

Motivés par les questions de Daniel Augot concernant le décodage algébrique des codes à résidus quadratiques et l'analyse du cryptosystème HFE, ainsi que celles de Jean-Charles Faugère sur la complexité de l'algorithme F5, les travaux de ma thèse ont abouti à des résultats généraux de complexité du calcul de bases de Gröbner pour des systèmes surdéterminés : définition des suites semi-régulières, développement asymptotique de leur degré de régularité mettant en oeuvre des méthodes de combinatoire analytique et d'analyse asymptotique. Cela améliore nettement les précédentes bornes connues : par exemple gain d'un facteur 11,65 sur le degré de régularité lorsque le nombre d'équations est le double du nombre de variables. Nous avons également obtenu des résultats de complexité particuliers à l'algorithme F5.

Ces résultats ont été adaptés dans le cas de polynômes à coefficients dans un corps fini, en particulier pour les systèmes booléens (à coefficients dans $\mathbb{F}_2$ contenant les équations de corps $x_i^2 - x_i$), et appliqués à la cryptologie (analyse du cryptosystème HFE, pertinence d'attaques algébriques). Nous avons effectué de nombreux tests expérimentaux, pour l'analyse du cryptosystème HFE ou le choix d'une bonne mise en équation pour le décodage algébrique des codes correcteurs d'erreur.

Ces thèmes de recherche dans lesquels s'inscrit ma thèse offrent des perspectives de recherche nombreuses et variées, tant d'un point de vue appliqué que théorique.

L'analyse du cryptosystème HFE (version basique) nous a permis de développer de nombreux outils d'analyse de cryptosystèmes algébriques. Il reste à appliquer ces méthodes aux variantes de HFE, réputées plus difficiles, ainsi qu'à d'autres cryptosystèmes ayant comme clef publique un système polynomial, comme le système TTM et ses variantes. Enfin les perspectives de développement en cryptanalyse algébrique pour des systèmes à clef secrète sont nombreuses, et attendues par la communauté cryptographique : nous avons vu que compter le nombre d'équations de bas degré vérifiées par un cryptosystème ne suffit pas à obtenir une attaque sur ce cryptosystème, la seule estimation de complexité que nous puissions donner (celle pour des suites semi-régulières) étant bien au-delà de la recherche exhaustive. Pour chaque cryptosystème, une phase de remise en équations est probablement nécessaire, suivie d'une analyse de régularité propre du système obtenu (en particulier prise en compte de l'aspect creux des systèmes).

Concernant le décodage algébrique des codes correcteurs, les systèmes que nous

avons proposés permettent en pratique des décodages extrêmement rapides. Les systèmes que nous donnons pour le décodage en ligne ne forment pas des suites semi-régulières, mais il doit être possible d'en extraire un système semi-régulier. Il sera très intéressant de pouvoir quantifier précisément la complexité de l'algorithme de décodage (algorithme polynômial, exponentiel ? bornes fines de complexité ?) en analysant la régularité de ces systèmes extraits.

La démarche utilisée dans ma thèse pour étudier les systèmes booléens est générale et peut s'appliquer à d'autres classes de problèmes : déterminer les réductions à zéro apparaissant au cours de l'algorithme F5, isoler un nouveau critère permettant de les éviter génériquement, en déduire une définition de régularité et une analyse de complexité du calcul de bases de Gröbner pour de tels systèmes. Ainsi l'étude des systèmes à coefficients dans un corps fini $\mathbb{F}_q$ ($q > 2$) contenant les équations de corps $x_i^q - x_i$ aura de nombreuses applications en protection de l'information. Un autre exemple concerne l'étude des systèmes formés des mineurs d'une matrice à coefficients polynomiaux. L'étude de tels systèmes permet par exemple d'analyser la complexité des systèmes bi-homogènes (voir [ST04]), ou peut servir à l'analyse de cryptosystèmes (par exemple le système Augot-Finiasz, voir [AFL03]). Tout nouveau critère intégré dans l'algorithme F5 fournit un algorithme adapté à une classe de systèmes et plus performant pour cette classe qu'un algorithme général de calcul de base de Gröbner.

Il sera enfin particulièrement intéressant de poursuivre l'étude fine de l'algorithme F5, effectuée dans cette thèse uniquement pour des systèmes non surdéterminés, en déterminant le nombre d'opérations élémentaires de cet algorithme pour des suites surdéterminés, et en étudiant son comportement asymptotique. Cela permettrait par exemple de comparer ce coût à celui de la recherche exhaustive sur des corps finis, ou d'une algèbre linéaire creuse sur la matrice de Macaulay en degré suffisamment élevé.

Annexe

A.1 Rappels sur les corps finis

Nous rappelons dans ce paragraphe quelques notions sur les corps finis. Nous ne donnons aucune preuve, le lecteur peut se référer à [LN97].

Théorème A.1.1 *Pour tout entier premier p et tout entier $m \geq 1$, il existe un corps fini d'ordre p^m, unique à isomorphisme près. On le note $\mathbb{F}_{p^m}$. Ses éléments sont les racines du polynôme*

$$x^{p^m} - x$$

Tout corps fini est isomorphe à un corps $\mathbb{F}_{p^m}$.

Définition A.1.2 *Tout élément α tel que $\alpha^k = 1$ et $\alpha^l \neq 1$ pour $0 < l < k$ est appelé une racine primitive $k^{ème}$ de l'unité.*

Théorème A.1.3 *Soit $q = p^m$ une puissance d'un nombre premier, alors le groupe multiplicatif $(\mathbb{F}_q \backslash \{0\}, *)$ est cyclique, i.e. il est engendré par un unique élément. Un tel générateur β est une racine primitive $q-1^{ème}$ de l'unité.*

Si n divise $q-1$, alors $\alpha = \beta^{(q-1)/n}$ est une racine primitive $n^{ème}$ de l'unité. Elle est définie par son polynôme minimal, qui est un diviseur irréductible de $x^{q-1} - 1$.

Exemple A.1.4 *Cherchons une racine* primitive $63^{ème}$ *de l'unité modulo 2. On cherche un facteur irréductible de $x^{63} - 1$ qui ne soit pas un facteur irréductible d'un $x^k - 1$ pour k divisant 63 (on utilise la commande Maple* `Factors`$(x^{63} - 1)$ mod 2*). Les facteurs irréductibles de $x^{63} - 1$ modulo 2 sont $x+1, x^2+x+1, x^3+x+1, x^3+x^2+1, x^6+x^3+1, x^6+x^5+x^4+x^2+1, x^6+x^4+x^2+x+1, x^6+x^4+x^3+x+1, x^6+x^5+1, x^6+x^5+x^3+x^2+1, x^6+x+1, x^6+x^5+x^2+x+1, x^6+x^5+x^4+x+1$. Or, $63 = 3^2 \cdot 7$, et $x^3+1 = (x+1)(x^2+x+1)$, $x^7+1 = (x+1)(x^3+x+1)(x^3+x^2+1)$, $x^9+1 = (x^3+1)(x^6+x^3+1)$ et $x^{21}+1 = (x^7+1)(x^2+x+1)(x^6+x^5+x^4+x^2+1)(x^6+x^4+x^2+x+1)$. Donc on peut prendre pour α n'importe quelle racine de l'un des polynômes restant, soit $x^6+x^4+x^3+x+1$, x^6+x^5+1, $x^6+x^5+x^3+x^2+1$, x^6+x+1, $x^6+x^5+x^2+x+1$ et $x^6+x^5+x^4+x+1$. Attention, l'élément défini par le facteur x^6+x^3+1 est une racine $9^{ème}$ de l'unité, et n'est donc pas une racine* primitive $63^{ème}$ *de l'unité.*

Théorème A.1.5 *Soit n un entier et q une puissance d'un nombre premier, considérons le polynôme $x^n - 1 \in \mathbb{F}_q[x]$.*

Si n et q sont premiers entre eux, alors le corps de décomposition de $x^n - 1$ est $\mathbb{F}_{q^m}$, où $m = \min\{\lambda > 0 : \ q^\lambda = 1 \mod n\}$ est l'ordre multiplicatif de q modulo n. Soit $\alpha \in \mathbb{F}_{q^m}$ une racine primitive $n^{ème}$ de l'unité, alors

$$x^n - 1 = \prod_{i \in [0,\ldots,n-1]} (x - \alpha^i)$$

Dans toute la suite, on prendra n et q premiers entre eux.

Définition A.1.6 *La* classe cyclotomique *d'un entier i modulo n sur un corps fini $\mathbb{F}_q$ (avec $pgcd(n, q) = 1$) est*

$$Cl(i) = \{i \cdot q^j \mod n, j \in \mathbb{N}\}$$

L'ordre multiplicatif de q modulo n est $m = Card(Cl(1))$.

Exemple. Les classes cyclotomiques modulo 9 sur $\mathbb{F}_2$ sont $Cl(1) = \{1, 2, 4, 8, 7, 5\}$ et $Cl(3) = \{3, 6\}$. L'ordre multiplicatif de 9 modulo 2 est 6.

Théorème A.1.7 *En utilisant les notations précédentes, le polynôme $M_i(x) = \prod_{s \in Cl(i)}(x - \alpha^s)$ est un polynôme à cœfficients dans $\mathbb{F}_q$. C'est le polynôme minimal de α^i.*

Réciproquement, tout polynôme $g(x)$ divisant $x^n - 1$, et à cœfficients dans $\mathbb{F}_q$ est un produit de polynômes minimaux et s'écrit donc $g(x) = \prod_{s \in D}(x - \alpha^s)$ où $D \subset [0, \ldots, n-1]$ est une réunion de classes cyclotomiques.

A.2 Rappels sur les idéaux de polynômes

Définition A.2.1 *Soit R un anneau. I est un* idéal *de R si :*

- $a + b \in I \quad \forall (a, b) \in I \times I$
- $-a \in I \quad \forall a \in I$
- $p \cdot a \in I \quad \forall a \in I \quad \forall p \in R$

Un idéal est radical *si* $(\exists n \in \mathbb{N} \quad f^n \in I) \Rightarrow (f \in I)$
Un idéal est principal *s'il est engendré par un unique élément. Un anneau est principal si tous ses idéaux le sont.*
Par exemple, l'anneau $\mathbb{K}[x]$, où $\mathbb{K}$ est un corps, est principal.
Un idéal est premier *si* $\forall (f, g) \in R \times R, \qquad f \cdot g \in I \Rightarrow f \in I$ *ou* $g \in I$.

Lemme A.2.2 (Seidenberg) *Soit I un idéal de $k[x_1, \ldots, x_l]$ tel que pour tout $i \in [1; n]$, il existe $g_i \in I \cap k[x_i] \neq 0$ tel que $pgcd(g_i, g_i') = 1$ alors I est radical.*

Corollaire A.2.3 *Soit I un idéal de $\mathbb{F}_q[x_1, \ldots, x_l]$, alors $\langle I, x_1^q - x_1, \ldots, x_n^q - x_n \rangle$ est radical.*

Démonstration Une preuve directe : grâce aux équations de corps, pour tout $g \in J = \langle I, x_1^q - x_1, \ldots, x_n^q - x_n \rangle$ on a $g^q = g$. Ainsi, si $g \in \sqrt{J}$ alors il existe un entier r tel que $g^r \in J$, et il existe un entier m tel que $q^m > r$, alors $g = g^{q^m} = g^{q^m - r} \cdot g^r \in J$. □

A.3 Représentation d'un polynôme de $\mathbb{F}_{q^n}[x]$ sur $\mathbb{F}_q$

Soit p un nombre premier, et $q = p^m$. Considérons le corps fini $\mathbb{F}_q$ et une extension $\mathbb{F}_{q^n}$ de degré n. Nous pouvons considérer des éléments du corps fini $\mathbb{F}_{q^n}$ comme des n-uplets d'éléments du corps fini $\mathbb{F}_q$, en utilisant la représentation de $\mathbb{F}_{q^n}$ comme quotient d'anneau de polynôme $\mathbb{F}_q[z]/(g(z))$ où $g(z)$ est un polynôme irréductible de degré n sur $\mathbb{F}_q$. Si w est une racine de $g(z)$, alors tout élément de $\mathbb{F}_{q^n}$ peut être vu comme un n-uplet d'éléments de $\mathbb{F}_q$ par l'isomorphisme suivant :

$$\begin{aligned} \mathbb{F}_{q^n} &\to (\mathbb{F}_q)^n \\ a = \sum_{i=0}^{n-1} a_{i+1} w^i &\mapsto (a_1, \ldots, a_n) \end{aligned}$$

Toute fonction $g : \mathbb{F}_{q^n} \to \mathbb{F}_{q^n}$ peut s'écrire $g(x) = f(x)$ avec $f \in \mathbb{F}_{q^n}[X]/(X^{q^n} - X)$.

De la même manière, nous pouvons considérer une fonction polynôme en une variable à coefficients dans $\mathbb{F}_{q^n}$ comme un n-uplet de fonctions polynômes en n variables à coefficients dans $\mathbb{F}_q$. Notons qu'une fonction polynôme en une variable x à coefficients dans $\mathbb{F}_{q^n}$ est un polynôme de $\mathbb{F}_{q^n}[x]/(x^{q^n} - x)$. Il suffit alors, pour toute fonction $f(x)$, de substituer $x = \sum_{i=0}^{n-1} x_{i+1} w^i$ dans $f(x)$, puis de regrouper les termes en fonctions des puissances de w qui apparaissent. Plus précisément on a l'isomorphisme suivant :

Lemma A.3.1 *L'application suivante est un isomorphisme de l'ensemble des fonctions en une variable à coefficients dans $\mathbb{F}_{q^n}$ dans l'ensemble des n-uplets de fonctions en n variables à coefficients dans $\mathbb{F}_q$:*

$$\begin{aligned} \mathbb{F}_{q^n}[x]/(x^{q^n} - x) &\to \mathbb{F}_{q^n}[x_1, \ldots, x_n]/(x_1^q - x_1, \ldots, x_n^q - x_n) \\ f(x) &\mapsto f(\sum_{i=0}^{n-1} x_{i+1} w^i) = \sum_{i=0}^{n-1} f_{i+1}(x_1, \ldots, x_n) w^i \end{aligned}$$

avec $f_i(x_1, \ldots, x_n)$ polynôme à coefficients dans $\mathbb{F}_q$, et on a l'isomorphisme :

$$\begin{aligned} \mathbb{F}_{q^n}[x_1, \ldots, x_n]/(x_1^q - x_1, \ldots, x_n^q - x_n) &\to (\mathbb{F}_q[x_1, \ldots, x_n]/(x_1^q - x_1, \ldots, x_n^q - x_n))^n \\ \sum_{i=0}^{n-1} f_{i+1}(x_1, \ldots, x_n) w^i &\mapsto \qquad (f_1(x_1, \ldots, x_n), \ldots, f_n(x_1, \ldots, x_n)) \end{aligned}$$

Démonstration Soit $\phi : \mathbb{F}_{q^n}[x]/(x^{q^n} - x) \to (\mathbb{F}_q[x_1, \ldots, x_n]/(x_1^q - x_1, \ldots, x_n^q - x_n))^n$ la composée des applications ci-dessus. Montrons que ϕ est surjective : soit $f_1, \ldots, f_n$ des fonctions polynômes en n variables dans $\mathbb{F}_q$, et $f(x) \in \mathbb{F}_{q^n}[x]/(x^{q^n} - x)$ la fonction polynôme qui, pour $y = \sum_{i=0}^{n-1} y_{i+1} w^i \in \mathbb{F}_{q^n}$ vaut $\sum_{i=0}^{n-1} f_{i+1}(y_1, \ldots, y_n) w^i$ (on la trouve par interpolation sur l'ensemble des points de $\mathbb{F}_{q^n}$). Alors on a $\phi(f) = (f_1, \ldots, f_n)$. L'injectivité provient du fait que tout polynôme de $\mathbb{F}_{q^n}[x]$ identiquement nul comme fonction polynôme peut s'écrire comme un polynôme en x^{q^n}. □

Si m est un monôme de $\mathbb{F}_{q^n}[x]/(x^{q^n} - x)$, qui s'écrit $m = x^{2^{i_1} + \ldots + 2^{i_v}}$ avec $0 \leq i_v < \ldots < i_1 \leq n-1$, alors chaque polynôme du n-uplet $\phi(m)$ sera de degré au plus égal à v le poids de Hamming binaire du degré de m (cela provient de la linéarité de la fonction $x \mapsto x^q$ sur $\mathbb{F}_{q^n}$).

A.4 Fonction Ai d'Airy

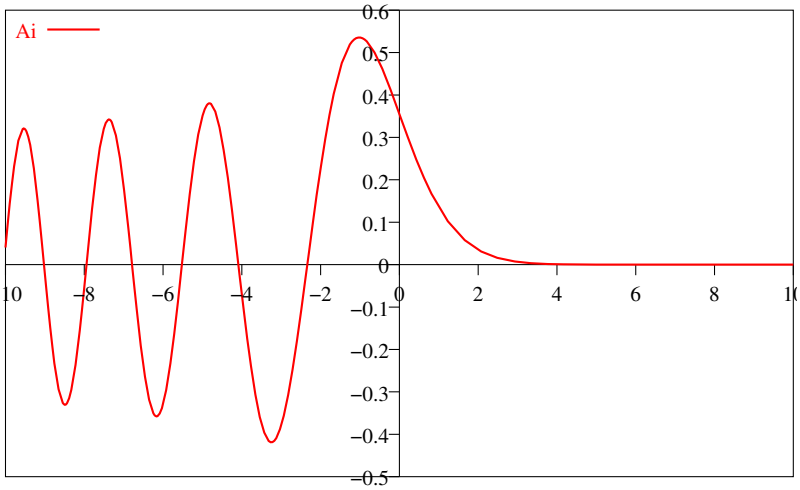
La fonction d'Airy [AS92, §10.4] apparaît couramment dans des domaines de la physique, comme l'optique, la mécanique quantique, l'électromagnétisme. Elle est définie comme l'une des deux solutions linéairement indépendantes de l'équation différentielle

$$y'' - yz = 0$$

et peut être écrite sous forme intégrale

$$\mathrm{Ai}(x) = \frac{1}{2i\pi} \int_{C_1} e^{\frac{v^3}{3} - xv} dv$$

où C_1 est un chemin d'origine un point à l'infini dans le secteur $-\frac{\pi}{2} \leq \arg(v) \leq -\frac{\pi}{6}$ et de fin un point dans le secteur conjugué.



Les plus grandes racines de la fonction d'Airy valent approximativement -2.33811, -4.08795, -5.52056, etc.

Table des figures

Liste des tableaux

Index

Bibliographie

[ABF02] D. Augot, M. Bardet, and J.-C. Faugère. Efficient decoding of (binary) cyclic codes above the correction capacity of the code using Gröbner bases. Research Report RR-4652, INRIA, Novembre 2002.

[ABF03] D. Augot, M. Bardet, and J.-C. Faugère. Efficient decoding of (binary) cyclic codes above the correction capacity of the code using Gröbner bases. In *Proc. IEEE International Symposium on Information Theory 2003 (ISIT, June 29 - July 4, Yokohama, Japan)*, page 362, 2003.

[AFL03] D. Augot, M. Finiasz, and P. Loidreau. Using the trace operator to repair the polynomial-based reconstruction cryptosystem presented at eurocrypt 2003. *Cryptology ePrint archive, Report 2003/209*, 2003.

[AK03] F. Armknecht and M. Krause. Algebraic attacks on combiners with memory. In *Advances in Cryptology - CRYPTO 2003*, volume 2729 of *Lecture Notes in Computer Science*, pages 162 – 175, Heidelberg, October 2003. Springer-Verlag.

[AM72] E. F. Assmus, Jr. and H. F. Mattson, Jr. On weights in quadratic-residue codes. *Discrete Math.*, 3 :1–20, 1972.

[Ani86] D. J. Anick. Thin algebras of embedding dimension three. *J. Algebra*, 100(1) :235–259, 1986.

[AS92] M. Abramowitz and I. A. Stegun, editors. *Handbook of mathematical functions with formulas, graphs, and mathematical tables*. Dover Publications Inc., New York, 1992. Reprint of the 1972 edition.

[Aug93] D. Augot. *Etude algébrique des mots de poids minimum des codes cycliques, Méthodes d'algèbre linéaire sur les corps finis*. PhD thesis, Université Paris VI, Décembre 1993.

[Aug96] D. Augot. Description of minimum weight codewords of cyclic codes by algebraic systems. *Finite Fields Appl.*, 2 :138–152, 1996.

[Bar97] A. Barg. Complexity issues in coding theory. *Electronic Colloquium on Computational Complexity (ECCC)*, 4(046), 1997.

[Bar98] A. Barg. Complexity issues in coding theory. In *Handbook of coding theory, Vol. I, II*, pages 649–754. North-Holland, Amsterdam, 1998.

[BDC03] A. Biryukov and C. De Cannière. Block ciphers and systems of quadratic equations. In *Proceedings of Fast Software Encryption*, pages 291–306, 2003.

[Ber68] E. R. Berlekamp. *Algebraic coding theory.* McGraw-Hill Book Co., New York, 1968.

[BF01] E. Byrne and P. Fitzpatrick. Gröbner bases over Galois rings with an application to decoding alternant codes. *J. Symbolic Comput.*, 31(5) :565–584, 2001.

[BFS03] M. Bardet, J.-C. Faugère, and B. Salvy. Complexity of gröbner basis computation for semi-regular overdetermined sequences over GF(2) with solutions in GF(2). Research Report RR-5049, INRIA, Décembre 2003. 19 pages.

[BFS04] M. Bardet, J.-C. Faugère, and B. Salvy. On the complexity of gröbner basis computation of semi-regular overdetermined algebraic equations. In *Proc. ICPSS International Conference on Polynomial System Solving Paris, November 24-25-26 2004 in honor of Daniel Lazard*, 2004.

[BG01] I. Bermejo and P. Gimenez. Computing the Castelnuovo-Mumford regularity of some subschemes of $\mathbb{P}^n_K$ using quotients of monomial ideals. *J. Pure Appl. Algebra*, 164(1-2) :23–33, 2001. Effective methods in algebraic geometry (Bath, 2000).

[BMMT94] E. Becker, T. Mora, M. G. Marinari, and C. Traverso. The shape of the shape lemma. In *International Symposium on Symbolic and Algebraic Computation*, pages 129–133, 1994.

[BMvT78] E. R. Berlekamp, R. J. McEliece, and H. C. A. van Tilborg. On the inherent intractability of certain coding problems. *IEEE Trans. Information Theory*, IT-24(3) :384–386, 1978.

[BS88] D. Bayer and M. Stillman. On the complexity of computing syzygies. *J. Symbolic Comput.*, 6(2-3) :135–147, 1988. Computational aspects of commutative algebra.

[Buc65] B. Buchberger. *Ein Algorithmus zum Auffinden der Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal.* PhD thesis, Innsbruck, 1965.

[Buc83] B. Buchberger. A note on the complexity of constructing Gröbner-bases. In *Computer algebra (London, 1983)*, volume 162 of *Lecture Notes in Comput. Sci.*, pages 137–145. Springer, Berlin, 1983.

[BW93] T. Becker and V. Weispfennig. *Gröbner bases*, volume 141 of *Graduate Texts in Mathematics.* Springer-Verlag, New York, 1993. A computational approach to commutative algebra, In cooperation with Heinz Kredel.

[CFU57] C. Chester, B. Friedman, and F. Ursell. An extension of the method of steepest descents. *Proc. Camb. Philos. Soc.*, 53 :599–611, 1957.

[CKM97] S. Collart, M. Kalkbrener, and D. Mall. Converting bases with the gröbner walk. *J. Symbolic Comput.*, 24 :465–469, 1997.

[CLO97] D. Cox, J. Little, and D. O'Shea. *Ideals, varieties, and algorithms*. Undergraduate Texts in Mathematics. Springer-Verlag, New York, second edition, 1997. An introduction to computational algebraic geometry and commutative algebra.

[CM02] M. Caboara and T. Mora. The Chen-Reed-Helleseth-Truong decoding algorithm and the Gianni-Kalkbrenner Gröbner shape theorem. *Appl. Algebra Engrg. Comm. Comput.*, 13(3) :209–232, 2002.

[CM03] N. T. Courtois and W. Meier. Algebraic attacks on stream ciphers with linear feedback. In *Advances in Cryptology – EUROCRPYT '03 (Warsaw, Poland, 2003)*, volume 2656 of *Lecture Notes in Computer Science*, pages 345 – 359, Heidelberg, January 2003. Springer-Verlag.

[Cop94] D. Coppersmith. Solving homogeneous linear equations over $GF(2)$ via block Wiedemann algorithm. *Math. Comput.*, 62(205) :333–350, 1994.

[CP02] N. T. Courtois and J. Pieprzyk. Cryptanalysis of block ciphers with overdefined systems of equations. In Y. Zheng, editor, *Advances in Cryptology - ASIACRYPT 2002*, number 2501 in LNCS, pages 267 – 287. Springer-Verlag Heidelberg, December 1-5 2002.

[CRHT94a] X. Chen, I. S. Reed, T. Helleseth, and T.-K. Truong. Algebraic decoding of cyclic codes : a polynomial ideal point of view. In *Finite fields : theory, applications, and algorithms (Las Vegas, NV, 1993)*, volume 168 of *Contemp. Math.*, pages 15–22. Amer. Math. Soc., Providence, RI, 1994.

[CRHT94b] X. Chen, I. S. Reed, T. Helleseth, and T.-K. Truong. General principles for the algebraic decoding of cyclic codes. *IEEE Trans. Inform. Theory*, 40(5) :1661–1663, 1994.

[CRHT94c] X. Chen, I. S. Reed, T. Helleseth, and T.-K. Truong. Use of Gröbner bases to decode binary cyclic codes up to the true minimum distance. *IEEE Trans. Inform. Theory*, 40(5) :1654–1661, 1994.

[CRT94] X. Chen, I. S. Reed, and T.-K. Truong. Decoding the (73,37,13) quadratic residue code. *IEE Proc., Comput. Digit. Tech.*, 141(5) :253–258, 1994.

[CW90] D. Coppersmith and S. Winograd. Matrix multiplication via arithmetic progressions. *J. Symbolic Comput.*, 9(3) :251–280, 1990.

[dB81] N. G. de Bruijn. *Asymptotic methods in analysis*. Dover Publications Inc., New York, third edition, 1981.

[Eis95] D. Eisenbud. *Commutative algebra*, volume 150 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1995. With a view toward algebraic geometry.

[Fau99] J.-C. Faugère. A new efficient algorithm for computing Gröbner bases (F_4). *J. Pure Appl. Algebra*, 139(1-3) :61–88, 1999. Effective methods in algebraic geometry (Saint-Malo, 1998).

[Fau02] J.-C. Faugère. A new efficient algorithm for computing Gröbner bases without reduction to zero (f_5). In T. Mora, editor, *ISSAC 2002*, pages 75–83, 2002.

[Fau03] J.-C. Faugère. Algebraic cryptanalysis of hfe using gröbner bases. Research Report RR-4738, INRIA, Février 2003.

[FGLM93] J.-C. Faugère, P. Gianni, D. Lazard, and T. Mora. Efficient computation of zero-dimensional Gröbner bases by change of ordering. *J. Symbolic Comput.*, 16(4) :329–344, 1993.

[FGT01] E. Fortuna, P. Gianni, and B. Trager. Degree reduction under specialization. *J. Pure Appl. Algebra*, 164(1-2) :153–163, 2001. Effective methods in algebraic geometry (Bath, 2000).

[FH94] R. Fröberg and J. Hollman. Hilbert series for ideals generated by generic forms. *J. Symbolic Comput.*, 17(2) :149–157, 1994.

[FJ98] P. Fitzpatrick and S. M. Jennings. Comparison of two algorithms for decoding alternant codes. *Appl. Algebra Engrg. Comm. Comput.*, 9(3) :211–220, 1998.

[FJ03] J.-C. Faugère and A. Joux. Algebraic cryptanalysis of Hidden Field Equation (HFE) cryptosystems using Gröbner bases. In D. Boneh, editor, *Advances in Cryptology - CRYPTO 2003*, volume 2729 of *LNCS*, pages 44–60. Springer, 2003.

[Frö85] R. Fröberg. An inequality for Hilbert series of graded algebras. *Math. Scand.*, 56(2) :117–144, 1985.

[Frö97] R. Fröberg. *An introduction to Gröbner bases*. Pure and Applied Mathematics. John Wiley & Sons Ltd., Chichester, 1997.

[FY79] A. S. Fraenkel and Y. Yesha. Complexity of problems in games, graphs and algebraic equations. *Discrete Appl. Math.*, 1(1-2) :15–30, 1979.

[FY80] A. S. Fraenkel and Y. Yesha. Complexity of solving algebraic equations. *Inform. Process. Lett.*, 10(4-5) :178–179, 1980.

[Gia89] P. Gianni. Properties of Gröbner bases under specializations. In J. H. Davenport, editor, *EUROCAL '87 (Leipzig, 1987)*, volume 378 of *Lecture Notes in Computer Science*, pages 293–297. Springer, Berlin, 1989.

[Giu84] M. Giusti. Some effectivity problems in polynomial ideal theory. In *EUROSAM 84 (Cambridge, 1984)*, volume 174 of *Lecture Notes in Comput. Sci.*, pages 159–171. Springer, Berlin, 1984.

[GM89] P. Gianni and T. Mora. Algebraic solution of systems of polynomial equations using groebner bases. In *Applied algebra, algebraic algorithms and error-correcting codes, Proc. 5th Int. Conf., AAECC-5,*

Menorca/Spain 1987, Lect. Notes Comput. Sci. 356, 247-257 , pages 247–257, 1989.

[GMN+91] A. Giovini, T. Mora, G. Niesi, L. Robbiano, and C. Traverso. "One sugar cube, please" or selection strategies in the Buchberger algorithm. In *Watt, Stephen M. (ed.), ISSAC '91. Proceedings of the 1991 international symposium on Symbolic and algebraic computation. Bonn, Germany, July 15–17, 1991. New York, NY : ACM Press, 49-54* , 1991.

[HH93] R. J. Higgs and J. F. Humphreys. Decoding the ternary Golay code. *IEEE Trans. Inform. Theory*, 39(3) :1043–1046, 1993.

[Hum92] J. F. Humphreys. Algebraic decoding of the ternary (13, 7, 5) quadratic residue code. *IEEE Trans. Inform. Theory*, 38(3) :1122–1125, 1992.

[Kob98] N. Koblitz. *Algebraic aspects of cryptography*, volume 3 of *Algorithms and Computation in Mathematics*. Springer-Verlag, Berlin, 1998. With an appendix by Alfred J. Menezes, Yi-Hong Wu and Robert J. Zuccherato.

[Lan02] S. Lang. *Algebra*, volume 211 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, third edition, 2002.

[Laz83] D. Lazard. Gröbner bases, Gaussian elimination and resolution of systems of algebraic equations. In *Computer algebra (London, 1983)*, volume 162 of *Lecture Notes in Comput. Sci.*, pages 146–156. Springer, Berlin, 1983.

[Laz01] D. Lazard. Solving systems of algebraic equations. *ACM SIGSAM Bulletin*, 35(3) :11–37, Septembre 2001.

[LN97] R. Lidl and H. Niederreiter. *Finite fields*. Cambridge University Press, Cambridge, second edition, 1997. With a foreword by P. M. Cohn.

[LVY97] P. Loustaunau and E. Von York. On the decoding of cyclic codes using Gröbner bases. *Appl. Algebra Eng. Commun. Comput.*, 8(6) :469–483, 1997.

[Mac02] F. S. Macaulay. On some formula in elimination. In *London Mathematical Society*, number 33 in 1, pages 3–27, May 1902.

[Mac16] F. S. Macaulay. *The Algebraic Theory of Modular Systems*. Cambridge Mathematical Library. Cambridge University Press, 1916.

[Mat89] H. Matsumura. *Commutative ring theory*, volume 8 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, Cambridge, translated from the japanese by m. reid. second edition edition, 1989.

[McE78] R. J. McEliece. A public-key cryptosystem based on algebraic coding theory. DSN Progress Report 42-44, Jet Propulsion Laboratory, CA, January-February 1978. pp. 114-16.

[MI88] T. Matsumoto and H. Imai. Public quadratic polynomial-tuples for efficient signature-verification and message-encryption. In *Advances in cryptology – EUROCRYPT '88 (Davos, 1988)*, volume 330 of *Lecture Notes in Comput. Sci.*, pages 419–453. Springer, Berlin, 1988.

[MM82] E. W. Mayr and A. R. Meyer. The complexity of the word problems for commutative semigroups and polynomial ideals. *Advances in Mathematics*, 46(3) :305–329, 1982.

[Moh99] T. T. Moh. A public key system with signature and master key functions. *Communications in Algebra*, 27(5) :2207–2222, 1999.

[Mon95] P. L. Montgomery. A block Lanczos algorithm for finding dependencies over GF(2). In L. C. Guillou, editor, *Advances in cryptology - EUROCRYPT '95. International conference on the theory and application of cryptographic techniques, Saint-Malo, France. Proceedings*, volume 921 of *LNCS*, pages 106–120. Springer-Verlag, Berlin, May 21-25 1995.

[MR02] S. Murphy and M. J. B. Robshaw. Essential algebraic structure within the aes. In M. Yung, editor, *Advances in Cryptology – CRYPTO 2002*, volume 2442 of *Lecture Notes in Computer Science*, pages 1–16, Berlin, 2002. Springer.

[MS77] F. J. MacWilliams and N. J. Sloane. *The Theory of Error-Correcting Codes*. Amsterdam : North Holland, 1977.

[MS91] G. Moreno-Socías. *Autour de la fonction de Hilbert-Samuel (escaliers d'idéaux polynomiaux)*. PhD thesis, Ecole Polytechnique, 1991.

[MS03a] T. Mora and M. Sala. On the Gröbner bases of some symmetric systems and their application to coding theory. *J. Symbolic Comput.*, 35(2) :177–194, 2003.

[MS03b] G. Moreno-Socías. Degrevlex Gröbner bases of generic complete intersections. *J. Pure Appl. Algebra*, 180(3) :263–283, 2003.

[Par00] K. Pardue. Generic sequences of polynomials. Unpublished, 2000.

[Pat95] J. Patarin. Cryptanalysis of the Matsumoto and Imai public key scheme of Eurocrypt '88. In *Coppersmith, Don (ed.), Advances in cryptology – CRYPTO '95 (Santa Barbara, 1995)*, volume 963 of *Lecture Notes in Comput. Sci.*, pages 248–261. Springer-Verlag, Berlin, 1995.

[Pat96a] J. Patarin. Hfe first challenge. http://www.minrank.org/challenge1.txt, 1996.

[Pat96b] J. Patarin. Hidden fields equations (HFE) and isomorphisms of polynomials (IP) : Two new families of asymmetric algorithms. In *Advances in cryptology – EUROCRYPT '96 (Saragossa, 1996)*, volume 1070 of *Lecture Notes in Comput. Sci.*, pages 33–48. Springer, Berlin, 1996.

[PR03] K. Pardue and B. Richert. Syzygies of semi-regular sequences. Preprint available at : http://www.math.lsa.umich.edu/~brichert/publications/, 2003.

[RH99] C. Rong and T. Helleseth. On methods of using grobner bases to decode cyclic codes up to actual minimum distance. unpublished, 1999.

[RRTC01] H. Ruhua, I. S. Reed, T.-K. Truong, and X. Chen. Decoding the (47,24,11) quadratic residue code. *IEEE Trans. Inform. Theory*, 47(3) :1181–1186, 2001.

[RTCY92] I. S. Reed, T.-K. Truong, X. Chen, and X. Yin. The algebraic decoding of the (41, 21, 9) quadratic residue code. *IEEE Trans. Inform. Theory*, 38(3) :974–986, 1992.

[RYT90] I. S. Reed, X. Yin, and T.-K. Truong. Algebraic decoding of the (32,16,8) quadratic residue code. *IEEE Trans. Inform. Theory*, 36(4) :876–880, 1990.

[ST04] M. Safey El Din and P. Trébuchet. Strong bi-homogeneous bezout theorem and its use in effective real algebraic geometry. Submitted to Journal of Complexity. Preprint Available at : http://www-calfor.lip6.fr/~safey/publi.html, 2004.

[Sza01] A. Szanto. Multivariate subresultants using Jouanoulou's resultant matrices. accepted to Journal of Pure and Applied Algebra. Preprint Available at : http://www.mathpreprints.com/math/Preprint/aszanto/20011204/2, 2001.

[vL99] J. H. van Lint. *Introduction to coding theory*, volume 86 of *Graduate Texts in Mathematics*. Springer-Verlag, Berlin, third edition, 1999.

[Win84] F. Winkler. On the complexity of the Gröbner-bases algorithm over $K[x, y, z]$. In *EUROSAM 84 (Cambridge, 1984)*, volume 174 of *Lecture Notes in Comput. Sci.*, pages 184–194. Springer, Berlin, 1984.

[Won89] R. Wong. *Asymptotic approximations of integrals*. Computer Science and Scientific Computing. Academic Press Inc., Boston, MA, 1989.