Addition formulae on hyperelliptic curves : applications to cryptography

Christophe Tran

THÈSE / UNIVERSITÉ DE RENNES 1
sous le sceau de l'Université Européenne de Bretagne

pour le grade de

DOCTEUR DE L'UNIVERSITÉ DE RENNES 1

Mention : Mathématiques et applications

École doctorale Matisse

présentée par

Christophe TRAN

préparée à l'unité de recherche 6625 CNRS - IRMAR
Institut de Recherche de Mathématiques de Rennes U.F.R. de Mathématiques

---

Formules d'addition sur les jacobiennes de courbes hyperelliptiques : applications à la cryptographie

Thèse soutenue à Rennes le 1er décembre 2014

devant le jury composé de :

David KOHEL
Professeur, Université d'Aix-Marseille / *Rapporteur*

Pierrick GAUDRY
Directeur de recherche, CNRS / *Rapporteur*

Christophe RITZENTHALER
Professeur, Université de Rennes 1 / *Examinateur*

David LUBICZ
Ingénieur, DGA-MI / *Examinateur*

Damien ROBERT
Chargé de recherche, INRIA Bordeaux / *Examinateur*

Sylvain DUQUESNE
Professeur, Université de Rennes 1 / *Directeur de thèse*

Remerciements

A tous :

Salut, merci, et bonne lecture.

Table des matières

Introduction

Cette thèse s'intéresse aux courbes elliptiques et hyperelliptiques et à leur utilisation en cryptographie. Je rappellerai donc dans cette introduction leurs places dans l'histoire des mathématiques, avant de les resituer dans un contexte cryptographique. Ceci fait, j'évoquerai les principaux articles qui m'ont poussé à choisir mes axes de recherche. Enfin, je donnerai un plan détaillé de ce document.

0.1 Histoire des courbes elliptiques

0.1.1 Le début de l'histoire avec Diophante

Les courbes elliptiques sont de vieux objets d'étude mathématique. Outre la géométrie algébrique d'où elles proviennent, ces courbes apparaissent aussi bien en théorie des nombres qu'en analyse. On verra dans cette thèse quelques applications de leur loi de groupe en cryptographie, mais on peut noter qu'elles interviennent également en mécanique des fluides, science des matériaux ou électrostatique. Pour résumer la riche histoire de ces courbes, je me suis basé sur [24], [46], [8], [40].

La première apparition de ces courbes se trouve dans l'oeuvre la plus importante du mathématicien grec du IIIième siècle Diophante : les Arithmétiques. Cette somme en 13 livres ne sera "redécouverte" en occident qu'après la chute de Constantinople. Après plus de 1000 ans d'absence, cet ouvrage influencera énormément les savants de la Renaissance et du siècle des Lumières, faisant par exemple réfléchir Newton ou Euler. Fermat travailla beaucoup sur les problèmes posés par Diophante : c'est d'ailleurs dans une traduction latine de ces Arithmétiques que Fermat livrera en annotations son fameux dernier théorème. Mais n'allons pas trop vite, et revenons à Diophante.

Diophante étudia à la Bibliothèque d'Alexandrie les mathématiques de son époque : géométrie, trigonométrie, mécanique. Il s'intéressa à trouver des méthodes de construction (aujourd'hui on dirait des algorithmes) de solutions rationnelles à des équations algébriques à plusieurs variables.

C'est dans ce cadre qu'il inventa, sans le savoir, la méthode "corde-et-tangente" pour l'addition sur une courbe. Pour comprendre ce coup de force, prenons d'abord l'exemple des équations quadratiques. C'est un problème bien plus vieux que Diophante : les babyloniens par exemple connaissaient déjà les triplets pythagoriciens [1]. Diophante fit pourtant une découverte importante :

Proposition 0.1.1. *Soit $f(x, y) \in \mathbb{Q}[x, y]$ un polynôme quadratique. Si l'équation $f(x, y) = 0$ admet au moins une solution rationnelle, alors elle en admet une infinité.*

En effet, si on note P cette solution initiale, alors toute droite de pente rationnelle coupe la courbe d'équation $f(x, y) = 0$ en une autre solution rationnelle. Diophante appliqua cette même méthode aux cubiques. Plus précisément, le problème 24 du livre IV de ses Arithmétiques se réécrit de façon moderne ainsi :

Étant donné un paramètre a, trouver les solutions rationnelles à l'équation

$$y(a - y) = x^3 - x.$$

Diophante traite le cas $a = 6$ en faisant le changement de variable $x = 3y - 1$: autrement dit, Diophante a regardé l'intersection entre la courbe et sa tangente au point $(-1, 0)$. C'est bien la méthode de doublement d'un point sur une courbe elliptique.

Par la suite, il faudra attendre le XVIIième siècle pour que Fermat donne en toute généralité les formules algébriques pour l'addition et le doublement ; puis ce sera Newton qui donnera l'interprétation géométrique (corde-et-tangente) de ces formules.

0.1.2 Ellipses et intégrales elliptiques

La seconde partie de l'histoire débute au XVIIième avec Wallis (qui le premier parla d'intégrales elliptiques) pour finir au XIXième siècle avec Jacobi, Abel, Weierstrass et Eisenstein. Commençons par introduire la définition suivante :

Définition 0.1.2. Soit $R \in \mathbb{C}(x, y)$ une fonction rationnelle et $f \in \mathbb{C}[t]$ sans facteur carré. La différentielle $R(t, \sqrt{f(t)})\mathrm{d}t$ est dite *hyperelliptique*. Elle est dite *elliptique* si f est de degré 3 ou 4.

1. Pythagore vécut au VIième siècle av. JC, les Babyloniens environ 3000 ans avant cela.

Ces quantités apparaissent naturellement dans les problèmes de rectification d'une courbe, c'est-à-dire de calcul de longueur d'arc. Ainsi, si $y = f(x)$ est une courbe C^1 sur un intervalle $[a, b]$, alors la longueur d'arc est donnée par la formule

$$L_a^b = \int_a^b \sqrt{1 + f'(x)^2}\mathrm{d}x.$$

On comprend maintenant pourquoi Wallis parla d'intégrales elliptiques : il voulait dire par là intégrales dont le calcul permet de calculer la longueur d'arc d'une ellipse. Voyons rapidement leur histoire.

Au bout de 40 ans de travail, Legendre publia entre 1825 et 1828 un traité en trois volumes [2] dans lequel il conclut notamment que toute intégrale provenant de la rectification d'une ellipse peut s'exprimer comme une des trois types suivants, qu'il appela naturellement intégrales elliptiques de première, seconde et troisième espèce :

$$\begin{aligned} F(x) &= \int_0^x \frac{\mathrm{d}t}{\sqrt{(1-t^2)(1-k^2t^2)}}, \\ G(x) &= \int_0^x \frac{(1-k^2x^2)\mathrm{d}t}{\sqrt{(1-t^2)(1-k^2t^2)}}, \\ \Pi(x) &= \int_0^x \frac{\mathrm{d}t}{(1+nx^2)\sqrt{(1-t^2)(1-k^2t^2)}}. \end{aligned}$$

0.1.3 Des intégrales aux fonctions elliptiques

Par la suite, Abel réalisa que plutôt que d'essayer de calculer ces intégrales, par exemple $u = F(x)$, il était bien plus intéressant de regarder les fonctions inverses : $x = f(u)$. Cette idée provient de la trigonométrie : ainsi, si la fonction

$$x \mapsto \int_0^x \frac{dt}{\sqrt{1-t^2}}$$

est importante, son inverse, la fonction sinus, est elle fondamentale. Il appela ainsi fonctions elliptiques les inverses des intégrales elliptiques.

Jacobi partit de cette idée et trouva le résultat suivant :

Théorème 0.1.3. *Aucune fonction méromorphe ne peut avoir plus de deux périodes indépendantes.*
Les fonctions méromorphes doublement périodiques sont exactement les fonctions elliptiques.

2. *Traité des fonctions elliptiques et intégrales Eulériennes.*

C'est bien là la définition moderne des fonctions elliptiques, celle que l'on retrouve dans le livre de Silverman [42] par exemple. Il ne reste qu'une dernière étape pour retrouver nos courbes elliptiques.

0.1.4 Des fonctions aux courbes elliptiques

Utilisant cette double périodicité, Eisenstein décrivit les fonctions elliptiques comme des séries entières :

Théorème 0.1.4. *La série suivante*

$$y(z) = \sum_{m,n=-\infty}^{\infty} (z + m\omega_1 + n\omega_2)^{-2} - \sum_{m,n=-\infty}^{\infty} (m\omega_1 + n\omega_2)^{-2}$$

où $\omega_1, \omega_2 \in \mathbb{C}$ et $\omega_1/\omega_2 \notin \mathbb{R}$ définit une fonction elliptique.
Elle vérifie une équation différentielle de la forme

$$y'(z)^2 = p(y(z)),$$

pour un polynôme p de degré 3 sans facteur carré.

Finalement, en 1863, Weierstrass introduisit sa fameuse fonction $\wp$:

$$\wp(z) = z^{-2} + \sum_{\substack{m,n=-\infty \\ (m,n)\neq(0,0)}}^{\infty} [(z + m\omega_1 + n\omega_2)^{-2} - (m\omega_1 + n\omega_2)^{-2}].$$

Théorème 0.1.5. *Toute fonction elliptique de double période ω_1 et ω_2 est une fonction rationnelle en $\wp$ et $\wp'$.*
Ces deux fonctions vérifient l'équation

$$\wp'(z)^2 = 4\wp(z)^3 - g_2\wp(z) - g_3,$$

où g_2 et g_3 sont des constantes ne dépendant que de ω_1 et ω_2.

Ces courbes cubiques, on les a déjà rencontrées avec Diophante et Fermat. Newton avait montré la méthode corde-et-tangente d'addition de points. Et Weierstrass montra qu'effectivement ses fonctions $\wp$ et $\wp'$ admettaient bien une formule d'addition qui est celle des points de la courbe. Ainsi, ces courbes cubiques paramétrées par les fonctions elliptiques prirent elles le nom de courbes elliptiques. La boucle est bouclée.

0.2 Les courbes elliptiques en cryptographie

En 1976, Diffie et Hellman inventèrent la cryptographie à clef publique. Encore appelée cryptographie asymétrique, elle repose sur l'existence supposée de fonctions à sens unique, qui permettent de chiffrer facilement des messages, mais dont le calcul de l'inverse doit être impossible pour qui ne connaît pas le secret. On peut grosso modo diviser les protocoles de cryptographie asymétrique en deux familles.

Le très célèbre RSA, inventé en 1977, fut la première réalisation pratique à clef publique. Le protocole commence par la multiplication de deux grands nombres premiers, et repose sur la difficulté, connaissant seulement le résultat du produit, de retrouver les deux facteurs.
Les protocoles de la seconde famille reposent sur le problème suivant, dit du logarithme discret : me donnant un groupe $(\mathbb{G}, \times)$ et deux éléments g et g' de ce groupe reliés par la relation

$$g' = g^x,$$

comment puis je retrouver x ?

Il est par exemple à la base du protocole d'échange de clefs Diffie-Hellman (1976) ou du chiffrement Elgamal (1989). Dans les implémentations originales de ces protocoles, le groupe utilisé est le groupe multiplicatif d'un corps fini $\mathbb{F}_p^*$, pour un grand premier p.

En 1985, Koblitz et Miller proposèrent tous deux de façon indépendante d'utiliser un autre groupe : celui des points d'une courbe elliptique définie sur un corps fini. La raison à cela est l'absence d'algorithme en temps sous-exponentiel pour attaquer le problème du logarithme discret sur ces groupes (plus de détails sont donnés dans le chapitre 4). Ainsi, à niveau de sécurité égal, la taille du groupe de points d'une courbe elliptique à utiliser est plus petite que celle du groupe multiplicatif d'un corps fini. En pratique, cela signifie de plus petites tailles de clefs et de messages chiffrés échangés. Ces avantages sont particulièrement significatifs dans le domaine des systèmes embarqués qui sont très restreints en capacité de mémoire et de borne passante.

Comme rappelé dans [26], la cryptographie basée sur courbes elliptiques [3] divisa d'abord les cryptographes : du fait de son exotisme (les courbes elliptiques étant alors peu connues et peu étudiées par les cryptographes), elle suscita chez certains enthousiasme et curiosité, et chez d'autres scepticisme et méfiance. Ces derniers arguèrent que les problèmes de la factorisation et du logarithme discret sur corps finis avaient été à ce moment là déjà intensivement étudiés, et qu'il faudrait at-

3. en anglais, *ECC* pour elliptic curve cryptography.

tendre un certain temps avant que la communauté cryptographique s'empare et comprenne vraiment la nature des courbes elliptiques.
Ainsi, ECDSA (Elliptic Curve Digital Signature Algorithm), proposé en 1992 ([51]), ne fut finalement unanimement accepté et inclus dans les standards que presque 10 ans plus tard (en 2000 par la NSA : [37]).

Finalement, juste quand l'utilisation des courbes elliptiques en cryptographie fut bien acceptée, apparurent les premiers protocoles basés sur les couplages [4]. Vieil outil mathématique, le couplage de Weil n'était pas à ce moment là inconnu des cryptographes, puisque Menezes, Okamoto et Vanstone dans [32] en 1993 avaient déjà montré comment transporter le problème du logarithme discret des points d'une courbe elliptique vers un groupe $\mathbb{F}_{q^k}^*$ (ici k est le degré de plongement, notion définie dans le chapitre 1). Cet article avait d'ailleurs à l'époque jeté un peu plus la suspicion sur les courbes elliptiques, notamment celles de petit degré de plongement.
Ces nouveaux protocoles ([25], [6] et [7], présentés plus en détail au chapitre 1) marquèrent les esprits en répondant de façon simple et élégante à de vieilles questions cryptographiques. Ainsi, les schémas d'échange de clef Diffie-Helman étaient bien connus et étudiés avant Joux ([25]), mais ils nécessitaient tous plus d'un tour d'échanges de données. Quant au chiffrement basé sur l'identité, c'est une vielle question posée par Shamir en 1984 à laquelle Boneh et Franklin ([6]) répondirent pour la première fois.

Ces articles ouvrirent une décennie de recherches actives sur les couplages : constructions de nouvelles primitives cryptographiques, impossibles sans l'utilisation de couplages, nouvelles techniques pour accélérer le calcul de couplages, nouvelles méthodes de sélections de courbes adaptées aux couplages (au passage, les courbes admettant un petit degré de plongement furent à cette occasion réhabilitées)...

En 1989, soit quelques années seulement après l'introduction des courbes elliptiques en cryptographie, Koblitz suggéra pour la première fois dans [27] d'utiliser leurs généralisations aux genres plus élevés : les courbes hyperelliptiques. Contrairement à la situation dans le cas elliptique, l'ensemble des points de ces courbes ne forme pas un groupe. Par contre, on peut toujours associer à chacune de ces courbes un groupe algébrique : sa jacobienne (la définition précise, et plus de détails seront donnés dans le chapitre 1). L'intérêt de considérer ces nouveaux groupes réside dans leur cardinal : en effet, une courbe hyperelliptique de genre g définie sur un corps fini $\mathbb{F}_q$ admet une jacobienne de cardinal environ q^g. Ainsi, utiliser une courbe de grand genre permet, à niveau de sécurité comparable, de travailler sur des corps finis plus petits, et ainsi de diminuer la taille des clefs et

4. pairing en anglais.

des messages chiffrés.

Pour savoir si un groupe est utilisable en cryptographie, un critère majeur à considérer est l'efficacité de sa représentation et de son arithmétique. Et malheureusement, pendant longtemps, les courbes hyperelliptiques futrent considérées comme inutilisables en cryptographie en pratique à cause de la grande difficulté de leur arithmétique (voir par exemple [44]). Le premier algorithme pour calculer dans la jacobienne d'une courbe hyperelliptique fut proposé par Cantor ([11]). Comme on le verra dans le chapitre 1, cet algorithme repose sur la représentation dite de Mumford des éléments de la jacobienne. Des améliorations de cet algorithme furent progressivement obtenus en étudiant les formules explicites pour un genre fixé (voir par exemple [29] pour le cas $g = 2$). Dans [21], Gaudry utilisa les fonctions thêta au lieu de la représentation de Mumford pour donner un algorithme très compétitif pour calculer sur les jacobiennes des courbes de genre 2.

Il nous faut maintenant évoquer la sécurité des courbes hyperelliptiques en cryptographie, c'est-à-dire la difficulté du problème du logarithme discret dans leur jacobienne. Pour résoudre ce problème, il y a deux familles d'algorithmes : les algorithmes dits génériques, parce qu'ils fonctionnent dans n'importe quel autre groupe, dont la complexité est exponentielle ; et les algorithmes de type calcul d'index, de complexité sous-exponentielle, qui exploitent l'arithmétique particulière des courbes hyperelliptiques. Le principe de ces algorithmes sera rappelé dans le chapitre 4. Ces algorithmes firent leur apparition pour la première fois en 1994 dans [1], mais n'étaient valables que quand le genre était suffisamment grand par rapport à la taille du corps de définition. Depuis, de nombreuses variantes et améliorations de ces algorithmes furent publiées : aujourd'hui, les algorithmes de type calcul d'index sont plus efficaces que les algorithmes génériques dès lors que le genre vaut au moins 3.

Ainsi, aujourd'hui, comme le souligne [2], tant la sécurité que les performances arithmétiques des courbes hyperelliptiques de genre 2 en font une alternative solide aux courbes elliptiques.

0.3 Quatre articles à la genèse d'une thèse en deux parties

Je venais tout juste de débuter ma thèse lorsque je trouvai deux articles ([45] et [30]) proposant deux nouvelles méthodes de calcul de couplages, originales car totalement indépendantes du traditionnel algorithme de Miller. Après une lecture même superficielle, une chose frappe très nettement : la similitude entre les formules de calcul de couplages introduites dans ces articles. Pourtant, ces

deux travaux semblaient totalement déconnectés, et n'utilisaient pas les mêmes objets : alors que David Lubicz et Damien Robert parlent de fonctions thêta et de formules d'addition, Katherine Stange, elle, introduit des elliptic nets [5], qu'elle inscrit très clairement dans la théorie des suites récurrentes, les reliant par exemple aux elliptic divisibility sequences.

Trois mois plus tard, à l'heure de faire un premier bilan de mon travail, voilà où j'en étais arrivé :

— En réalité, les fonctions thêta d'une courbe elliptique $\mathcal{E}$ ne sont pas des fonctions proprement définies sur $\mathcal{E}$: on ne peut pas les évaluer individuellement sur un point $P \in \mathcal{E}$, il faut ou bien effectuer un quotient de fonctions thêta pour obtenir une véritable fonction elliptique, ou bien considérer ensemble tout un paquet de fonctions thêta bien choisies (on parle alors de coordonnées thêta). Pour schématiser, la première méthode est utilisée dans [45], et la seconde dans [30].

— Les fonctions thêta d'une courbe elliptique vérifient toute une série de formules d'addition, dérivées des relations de Riemann, qui traduisent l'arithmétique de la courbe. Ce sont ces formules d'addition qui sont à la base de l'algorithme donné dans [30], qui permet par un double-and-add de calculer les coordonnées thêta d'un point $Q + rP$, pour un grand entier r.

— La fonction σ de Weierstrass est une fonction thêta particulière, qui permet de construire tous les elliptic nets. Or, il se trouve que la relation de récurrence définissant les elliptic nets est une conséquence des formules d'addition vérifiées par cette fonction thêta. Quant aux autres fonctions thêta, ce sont presque des elliptic nets : elles vérifient une relation de récurrence assez similaire à celles des elliptic nets, mais un peu plus compliquée. Ceci explique la similitude relevée plus haut entre les formules de couplages données dans les deux articles [45] et [30].

— Une autre application des elliptic nets est mise en avant dans [41] : il s'agit d'attaquer le problème du logarithme discret sur une courbe elliptique. Plus précisément, il s'agit de décider quand un ensemble de n points $P_1, \ldots, P_n$ d'une courbe vérifie une relation du type

$$P_1 \pm \cdots \pm P_n = \mathcal{O}.$$

Ces relations se détectaient en pratique via les polynômes de Semaev ([43]) : le but de [41] est justement de relier elliptic nets et polynômes de Semaev.

Ces notions se généralisent aux genres supérieurs : le travail de [30] concerne d'ailleurs les variétés abéliennes en toute généralité, tandis que [50], paru au

5. aucune traduction française de ce terme ne sera proposé dans ce document.

printemps 2012, généralise la notion d'elliptic nets aux courbes hyperelliptiques.

Ainsi, en partant de ces quatre articles [30], [45], [50] et [41], j'avais mes deux thèmes de recherche pour ma thèse : travailler sur les couplages d'une part, et sur la recherche de relations pour les algorithmes de type calcul d'index attaquant le logarithme discret.

0.4 Résumé du document

Cette thèse s'organise en quatre chapitres : un premier chapitre de rappels, et trois autres dans lesquels je donne mes résultats.

Chapitre 1

Dans ce chapitre je rappelle les notions mathématiques nécessaires à la lecture du manuscrit. Je commence par donner la définition des courbes hyperelliptiques et de leur jacobienne, ainsi que la représentation de Mumford des diviseurs. Je rappelle ensuite la définition des fonctions de Miller, et leur utilisation pour le calcul de couplages. Enfin, je rappelle la théorie des fonctions thêta, et celle des elliptic (et hyperelliptic) nets.

Chapitre 2

Ce chapitre traite de calcul de couplages. Après avoir rappelé la méthode de calcul de couplages via les fonctions thêta donnée dans [30], je donne l'algorithme basé sur [45] permettant de calculer les couplages sur une courbe hyperelliptique avec les hyperelliptic nets. Cet algorithme est le premier résultat neuf énoncé dans cette thèse. Il admet deux variantes, selon que le genre de la courbe vérifie $g \equiv 1, 2 \pmod 4$ ou $g \equiv 0, 3 \pmod 4$. Ces deux variantes sont énoncées respectivement dans les sections 2.2.1 et 2.2.2. Je donne une analyse des coûts de ces deux variantes dans la section 2.2.4. Enfin, ce chapitre se termine par une comparaison des coûts pour l'utilisation des elliptic nets et celle des fonctions thêta en genre 1 (section 2.3).

Chapitre 3

Jusqu'à présent, les méthodes de calcul de couplages par les fonctions thêta ([30]) et par les nets ([45] et [50]) étaient traitées comme si elles étaient indépendantes. Le chapitre 3 est dédié à leur réunification. Je mets ainsi en évidence le fait que tout système de coordonnées de niveau l pair admet une loi d'addition (théorème 3.4.1), et que ces formules d'addition ont pour

conséquence une relation de récurrence (théorème 3.4.2), que j'appelle récurrence des hyperelliptic nets généralisés (en particulier, pour $l = 2$ on retombe sur la récurrence des hyperelliptic nets classiques).

Chapitre 4

Dans ce chapitre, je passe à ma deuxième thématique de recherches : la recherche de relations dans l'algorithme de calcul d'index. Contrairement à [41], je ne réécris pas les polynômes de Semaev en fonction des elliptic nets, mais directement avec la fonction sigma de Weierstrass : je montre en effet que c'est bien la loi d'addition vérifiée par cette fonction qui lui permet de détecter des relations. Je construis ainsi des polynômes avec la fonction sigma dans le lemme 4.3.4 puis le théorème 4.3.5. Puis je montre que ces polynômes sont bien les polynômes de Semaev dans la proposition 4.3.8 et le corollaire 4.3.9. A partir de ce travail préparatoire en genre 1, je généralise la notion de polynômes de sommation aux genres supérieurs (théorème 4.4.7), ce qui n'avait pas été fait jusque là. J'étudie ensuite l'impact de ces nouveaux polynômes dans la complexité de l'algorithme de calcul d'index (section 4.4.5). Je termine le chapitre par la construction de polynômes de sommation alternatifs (section 4.5), toujours basés sur la fonction sigma et ses lois d'addition. En effet, ma construction originelle des polynômes de sommation était dictée par les choix et le travail de Semaev dans [43]. Il s'agit alors pour moi de discuter de la pertinence de ces choix. Hormis dans un cas particulier, ces polynômes alternatifs ne semblent pas être plus efficaces que les originaux.

Notations

Dans l'ensemble de cette thèse :

— $\mathbb{K}$ est un corps, dont $\overline{\mathbb{K}}$ dénote une clôture algébrique ;
— $\mathbb{F}_q$ le corps fini à q éléments ; sa caractéristique sera notée p ;
— $\mathcal{E}$ désignera toujours une courbe elliptique, $\mathcal{C}$ une courbe hyperelliptique et $\mathcal{J}$ la jacobienne d'une courbe ;
— les courbes considérées seront toujours dans le modèle imaginaire (voir définition 1.1.1) ; ∞ désigne le point à l'infini de la courbe, et $\mathcal{O}$ l'élément neutre de $\mathcal{J}$;
— si $f : \mathcal{C} \to \mathcal{C}'$ est une application rationnelle non constante entre deux courbes définies sur un corps $\mathbb{K}$, on note $f^* : \mathbb{K}(\mathcal{C}') \to \mathbb{K}(\mathcal{C})$ son pullback défini par
$$f^*(r') = r' \circ f;$$
— avec les mêmes notations, est également défini l'homomorphisme
$$\begin{aligned} f^* : \mathrm{Div}(\mathcal{C}') &\to \mathrm{Div}(\mathcal{C}) \\ (Q) &\mapsto \sum_{\substack{P \in \mathcal{C} \\ f(P)=Q}} e_f(P)(P), \end{aligned}$$
où $e_f(P)$ est l'index de ramification de f en P ;
— un homomorphisme classique qui sera utilisé dans cette thèse est le morphisme de translation ; si $P \in \mathcal{E}$ est un point de la courbe elliptique $\mathcal{E}$, alors $\tau_P : \mathcal{E} \to \mathcal{E}$ est le morphisme défini par
$$\tau_P(Q) = P + Q;$$
— $a \in_R A$ signifie que l'on doit choisir un élément a uniformément aléatoirement dans l'ensemble fini A ;
— $\{0,1\}^*$ désigne l'ensemble des chaînes de bits de longueur finie ;
— pour un entier n et un point $P \in \mathcal{E}$, la multiplication scalaire de n et P est notée $[n]P$:
$$[n]P = \underbrace{P + P + \cdots + P}_{n};$$

— pour un entier n, $\mathcal{J}[n]$ est la n-torsion de $\mathcal{J}$, i.e.

$$\mathcal{J}[n] = \{D \in \mathcal{J} \mid [n]D = \mathcal{O}\}.$$

Chapitre 1

Préliminaires

Dans cette partie, je commencerai par rappeler des notions basiques sur l'arithmétique des courbes. Je consacrerai ensuite une section aux couplages, avant de rappeler les résultats de [30] sur le calcul de couplages via les fonctions thêta d'une part, et de [45] et [50] sur leur calcul par les (hyper)elliptic nets d'autre part.

1.1 Arithmétique des courbes

Cette section est volontairement succincte. Plus de détails seront trouvés dans les références classiques que sont par exemple le chapitre 4 de [3], [18] ou les deux premiers chapitres de [42].

Définition 1.1.1. Une *courbe hyperelliptique* $\mathcal{C}$ de genre g sur un corps $\mathbb{K}$ est une courbe non singulière donnée par une équation du type

$$\mathcal{C} : y^2 + H(x)y = F(x),$$

avec H et $F \in \mathbb{K}[x]$, $2g+1 \leq \deg(F) \leq 2g+2$, $\deg(H) \leq g+1$.
Je ne considérerai que les courbes hyperelliptiques *imaginaires*[1] , c'est à dire avec F de degré $2g+1$ et $\deg(H) \leq g$. Dans ce cas, $\mathcal{C}$ a un point à l'infini, noté ∞.

Remarque 1.1.2. *Si on travaille sur un corps de caractéristique différente de* 2*, via la transformation* $y \mapsto y + H(x)/2$*, la courbe* $\mathcal{C}$ *donnée dans 1.1.1 est isomorphe à*

$$y^2 = F'(x)$$

avec $deg(F') = 2g+1$.

1. Plus précisément, *les cryptographes* ne considèrent généralement que les courbes imaginaires. La raison à cela est que l'arithmétique de ces courbes est plus rapide.

À chacune de ces courbes, est associé un groupe abélien : sa jacobienne. Je commence par rappeler la notion de diviseur.

Définition 1.1.3. Soit $\mathcal{C}/\mathbb{K}$ une courbe hyperelliptique. Un *diviseur* D de $\mathcal{C}$ est une somme formelle finie de points de $\mathcal{C}$:

$$D = \sum_{P_i \in \mathcal{C}} n_i(P_i),$$

où les n_i sont des entiers relatifs presque tous nuls.

L'ensemble des diviseurs muni de la somme formelle de points est un groupe commutatif noté $Div(\mathcal{C})$.

Définition 1.1.4. Le *degré* d'un diviseur est la somme de ses coefficients :

$$\deg\left(\sum_{P_i \in \mathcal{C}} n_i(P_i)\right) = \sum n_i.$$

L'ensemble des diviseurs de degré 0 forme un sous groupe $Div^0(\mathcal{C})$ de $Div(\mathcal{C})$.

Parmi les diviseurs de degré zéro, on distingue les diviseurs principaux :

Définition 1.1.5. Si $f \in \overline{\mathbb{K}}(\mathcal{C})$, on lui associe un diviseur de degré 0 :

$$\mathrm{div}(f) = \sum_{P_i \mathcal{C}} \mathrm{ord}_{P_i}(f)(P_i).$$

Un diviseur D de $\mathcal{C}$ est dit *principal* s'il existe une fonction $f \in \overline{\mathbb{K}}(\mathcal{C})$ tel que

$$D = \mathrm{div}(f).$$

L'ensemble $Princ(\mathcal{C})$ des diviseurs principaux forme un sous groupe de $Div^0(\mathcal{C})$.

Finalement, voici la jacobienne de $\mathcal{C}$:

Définition 1.1.6. La *jacobienne* $\mathcal{J}$ de $\mathcal{C}$ est le groupe des diviseurs de degré zéro quotienté par le groupe des diviseurs principaux :

$$\mathcal{J} = Div^0(\mathcal{C})/Princ(\mathcal{C}).$$

L'élément neutre de $\mathcal{J}$ est noté $\mathcal{O}$.
Deux diviseurs D et D' sont dits *linéairement équivalents* s'ils sont dans la même classe modulo $Princ(\mathcal{C})$. On note alors

$$D \sim D'.$$

Le théorème de Riemann Roch est un outil essentiel pour l'étude des courbes algébriques. Pour le moment, il va nous servir à énoncer la représentation de Mumford. Avant l'énoncé de ce théorème, il faut rappeler deux définitions.

Définition 1.1.7. Un diviseur D est dit *effectif* si tous ses coefficients sont positifs. On note $D \geq 0$.
Plus généralement, pour deux diviseurs D et D', on note $D \geq D'$ si le diviseur $D - D'$ est effectif.

Définition 1.1.8. Soit D un diviseur de $\mathcal{C}$. L'espace de Riemann Roch associé à D est l'espace vectoriel

$$\mathcal{L}(D) = \{f \in \overline{\mathbb{K}}(\mathcal{C})^* \mid \mathrm{div}(f) \geq -D\} \cup \{0\}.$$

C'est un $\overline{\mathbb{K}}$-espace vectoriel de dimension finie, sa dimension est notée $l(D)$.

Exemple 1.1.9 ([42])**.**
- *Si* $D < 0$*, alors* $\mathcal{L}(D) = \{0\}$.
- *Si* $D \sim D'$*, alors* $\mathcal{L}(D) \simeq \mathcal{L}(D')$.

Théorème 1.1.10 (Riemann-Roch)**.** *Il existe un diviseur* $K_{\mathcal{C}}$ *de* $\mathcal{C}$ *appelé diviseur canonique tel que pour tout* $D \in Div(\mathcal{C})$*,*

$$l(D) - l(K_{\mathcal{C}} - D) = \deg(D) - g + 1.$$

Si $\deg(D) > 2g - 2$*, alors*

$$l(D) = \deg(D) - g + 1.$$

Munis de ce théorème, nous avons le résultat suivant :

Proposition 1.1.11. *Toute classe* $D \in \mathcal{J}$ *admet un représentant unique de la forme*

$$D \sim \sum_{i=1}^{r} (P_i) - r(\infty)$$

avec $r \leq g$ *et pour tout* $i \neq j$*,* $P_i \neq -P_j$.

Définition 1.1.12. Une telle écriture de D est appelée *forme réduite* de D. Elle sera notée $\rho(D)$.

Corollaire 1.1.13. *Tout* $D \in \mathcal{J}$ *se représente de façon unique par une paire de polynômes* $u, v \in \mathbb{K}[x]$ *tels que*
- u *est unitaire ;*
- $deg(v) < deg(u) \leq g$ *;*
- $u | v^2 + vh - f$.

En pratique, cela signifie que pour

$$D \sim \sum_{i=1}^{r}(P_i) - r(\infty),$$

où pour tout i, $P_i = (x_i, y_i)$*, on a*
- $u(x) = \prod_i (x - x_i)$ *et*
- *pour tout* i*,* $v(x_i) = y_i$*.*

Définition 1.1.14. Une telle paire de polynômes (u, v) est la *représentation de Mumford* de D.

Dans [11], Cantor donne un algorithme prenant en entrée deux diviseurs réduits D_1 et D_2 pour donner en sortie la représentation réduite de $D_1 + D_2$.

J'aurai besoin dans la suite de la notation suivante :

Définition 1.1.15. Pour $D = \sum_{P_i} n_i(P_i) \in \mathcal{J}$, on note $\epsilon(D)$ la *partie effective* de D, c'est-à-dire

$$\epsilon(D) = \sum_{\substack{P_i \\ n_i > 0}} n_i(P_i).$$

Exemple 1.1.16. *Si* D *est réduit, alors* $D = \sum_{i=1}^{r}(P_i) - r(\infty)$ *pour un* $r \leq g$*, et alors on a simplement*

$$\epsilon(D) = \sum_{i=1}^{r}(P_i).$$

Enfin, un dernier objet à définir ici : le diviseur Θ.

Définition 1.1.17. Le diviseur $\Theta \subset \mathcal{J}$ est l'ensemble des diviseurs dont la forme réduite admet $r < g$ points. Ainsi,

$$\mathcal{J} \backslash \Theta = \{D \sim (P_1) + \cdots + (P_g) - g(\infty) | \forall i \neq j,\ P_i \neq -P_j\}.$$

On note également pour $i < g$, les sous ensembles

$$\Theta^{[i]} = \{D \sim \sum_{k=1}^{j}(P_k) - r(\infty) \mid j \leq i\}.$$

On a donc

$$\{\mathcal{O}\} = \Theta^{[0]} \subset \Theta^{[1]} = \{(P) - (\infty)\} \subset \cdots \subset \Theta = \Theta^{[g-1]}.$$

Exemple 1.1.18. *Les courbes qu'on étudiera le plus ici sont les courbes elliptiques et les courbes hyperelliptiques de genre* 2.
Pour $g = 1$, *on a* $\Theta = \{\mathcal{O}\}$.
Pour $g = 2$,

$$\Theta^{[1]} = \{\mathcal{O}\} \subset \Theta = \{(P) - (\infty) \mid P \in \mathcal{C}\},$$

tandis que

$$\mathcal{J} \setminus \Theta = \{(P_1) + (P_2) - 2(\infty) \mid P_1 \neq -P_2\}.$$

1.2 Couplages

Je commence par rappeler ce qu'est un couplage[2] en cryptographie, et par donner quelques brefs exemples de protocoles les utilisant.
J'expliquerai ensuite comment sont calculés ces couplages en pratique. Il existe plusieurs résumés récents de l'état de l'art sur les couplages, tel que [4] et [19] par exemple.

1.2.1 Définition

Définition 1.2.1. En cryptographie, un *couplage* est une application bilinéaire non dégénérée

$$e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T,$$

où $(\mathbb{G}_1, +)$, $(\mathbb{G}_2, +)$ et $(\mathbb{G}_T, \times)$ sont trois groupes.

Remarque 1.2.2. *En pratique, ces groupes seront supposés* cryptographiques*, c'est-à-dire dont l'arithmétique soit assez rapide pour permettre les calculs, mais sur lesquels le problème du logarithme discret soit difficile.*

1.2.2 Exemples d'utilisation

En 2000, Joux ([25]) présenta le premier protocole cryptographique utilisant les couplages. Depuis, des centaines d'articles[3] ont démontré l'utilité de cet outil. Ici, je présenterai rapidement, outre l'échange de clef tripartite à un tour de Joux, le chiffrement basé sur l'identité de Boneh et Franklin ([6]) et le schéma de signature courte de Boneh, Lynn et Shacham ([7]). Ces protocoles sont en effet les plus souvent cités dans les articles résumant la cryptographie basée sur les couplages.

Dans tous ces exemples, r sera un grand nombre premier, et le cardinal des trois groupes $\mathbb{G}_1$, $\mathbb{G}_2$ et $\mathbb{G}_T$. Le point $P \in \mathbb{G}_1$ sera un générateur du groupe.

2. en anglais, pairing.
3. le chapitre 10 de [5] en recensait 28 mi-2002 et environ 100 début 2004.

1.2.2.1 Le protocole de Joux [25]

Dans ce protocole, on suppose que $\mathbb{G}_1 = \mathbb{G}_2$. Trois parties A_1, A_2, et A_3 ont chacune une clef secrète $a_i \in \mathbb{Z}/r\mathbb{Z}$, et une clef publique correspondante $[a_i]P \in \mathbb{G}_1$.

Durant le tour d'échange, chaque participant envoie simplement sa clef publique aux deux autres participants.

Chaque participant A_i ayant reçu les deux autres clefs publiques $[a_j]P$ et $[a_k]P$ est alors en mesure de calculer avec sa propre clef secrète la clef commune :

$$e([a_j]P, [a_k]P)^{a_i} = e(P, P)^{a_i a_j a_k}.$$

1.2.2.2 Le chiffrement de Boneh et Franklin [6]

Dans un schéma de chiffrement basé sur l'identité, la clef publique de chaque utilisateur est dérivée de son identité (typiquement, de son adresse mail), ce qui règle le problème de la distribution des clefs.

Dans un schéma de chiffrement basé sur l'identité, si Alice veut envoyer un message à Bob, elle chiffre son message avec l'identité de Bob. Recevant le message, ce dernier s'authentifie auprès d'une autorité de confiance, le PKG (private key generator), qui lui remet sa clef privée.

Voici le premier protocole de chiffrement basé sur l'identité, proposé par Boneh et Franklin :

Setup On suppose que $\mathbb{G}_1 = \mathbb{G}_2$. On se donne un point $P \in \mathbb{G}_1$. Soient $s \in_R \mathbb{Z}/r\mathbb{Z}$, et $P_{pub} = [s]P$. On se donne également deux fonctions de hachage

$$H_1 : \{0,1\}^* \to \mathbb{G}_1 \text{ et } H_2 : \mathbb{G}_T \to \{0,1\}^n,$$

où n est la taille des messages. La clef maîtresse est s, et la clef publique globale est P_{pub}.

Extraction Étant donnée une identité (publique) $Bob \in \{0,1\}^*$, la clef publique se calcule par $Q_{Bob} = H_1(Bob) \in \mathbb{G}_1$, et la clef privée par $S_{Bob} = [s]Q_{Bob}$.

Chiffrement Alice commence par calculer

$$g_{Bob} = e(Q_{Bob}, P_{pub})$$

Soit $k \in_R \mathbb{Z}/r\mathbb{Z}$, le message M se chiffre en

$$C = \langle [k]P, M \oplus H_2(g_{Bob}^k) \rangle.$$

Déchiffrement Bob reçoit $C = \langle U, V \rangle$ de la part d'Alice. Il calcule alors

$$V \oplus H_2(e(S_{Bob}, U))$$

pour retrouver le message M.

1.2.2.3 La signature courte de Boneh Lynn et Shacham [7]

Génération On se donne une fonction de hachage $H : \{0,1\}^* \to \mathbb{G}_2$, une clef secrète $s \in_R \mathbb{Z}/r\mathbb{Z}$, et une clef publique $P_{pub} = [s]P \in \mathbb{G}_1$.

Signature Étant donné le secret s et le message $m \in \{0,1\}^*$, la signature se calcule simplement par

$$\sigma = [s]H(m).$$

Vérification Bob reçoit le couple $\langle m, \sigma \rangle$. Il accepte la signature si et seulement si l'égalité

$$e(P, \sigma) = e(P_{pub}, H(m))$$

est vérifiée.

1.2.3 Le couplage de Tate

Les couplages sont des outils mathématiques importants pour l'étude de la théorie des courbes. Ainsi, Weil introduisit son couplage en 1940 ([55]), et en 1985, Miller ([33]) publia son algorithme pour le calculer, et montra comment l'utiliser pour réduire le problème du logarithme discret sur la courbe elliptique au même problème sur le groupe multiplicatif d'un corps fini.

Les couplages utilisés en cryptographie sont le couplage de Tate et ses avatars (ate, ate-i, Rate...), dont voici la définition :

Définition 1.2.3. Soit $\mathcal{C}$ une courbe hyperelliptique de genre g définie sur le corps $\mathbb{F}_q$, et $\mathcal{J}$ sa jacobienne. Soit r un grand diviseur premier de $\sharp\mathcal{J}(\mathbb{F}_q)$. On suppose que r est premier à q.
Le *degré de plongement*[4] est le plus petit entier k tel que $r|(q^k - 1)$. Cela signifie que $\mathbb{F}_{q^k}$ est la plus petite extension de $\mathbb{F}_q$ à contenir les racines r-ièmes de l'unité μ_r.

On avait supposé que $r|\sharp\mathcal{J}(\mathbb{F}_q)$, on doit maintenant également demander que $r^2 \nmid \sharp\mathcal{J}(\mathbb{F}_q)$: ceci permet d'identifier $\mathcal{J}(\mathbb{F}_{q^k})[r]$ et $\mathcal{J}(\mathbb{F}_{q^k})/r\mathcal{J}(\mathbb{F}_{q^k})$.

Alors, pour $D_1 \in \mathcal{J}(\mathbb{F}_{q^k})[r]$, on note f_{r,D_1} la fonction[5] de diviseur

$$\mathrm{div}(f_{r,D_1}) = rD_1.$$

4. en anglais, embedding degree.
5. f_{r,D_1} est uniquement définie à constante multiplicative près.

Soit $D_2 \in \mathcal{J}(\mathbb{F}_{q^k})[r]$. On suppose que $\mathrm{supp}(D_1) \cap \mathrm{supp}(D_2) = \varnothing$. Alors la quantité

$$(f_{r,D_1}(D_2))^{\frac{q^k-1}{r}} = \left(\prod_P f_{r,D_1}(P)^{v_P(D_2)}\right)^{\frac{q^k-1}{r}}$$

définit une application bilinéaire non dégénérée

$$t : \mathcal{J}(\mathbb{F}_{q^k})[r] \times \mathcal{J}(\mathbb{F}_{q^k})[r] \to \mu_r.$$

Comme noté dans [20], si D_1 est correctement choisi, on a l'amélioration suivante :

Lemme 1.2.4 ([20]). *On suppose que $k > 1$, et que $D_1 \in \mathcal{J}(\mathbb{F}_q)[r]$. Alors*

$$t(D_1, D_2) = f_{r,D_1}(\epsilon(D_2))^{\frac{q^k-1}{r}}.$$

Ainsi, pour des raisons d'efficacité, le domaine de définition du couplage de Tate est réduit à $\mathbb{G}_1 \times \mathbb{G}_2$, où les groupes $\mathbb{G}_1$ et $\mathbb{G}_2$ sont définis de la façon suivante :

Définition 1.2.5. On note π le morphisme de Frobenius $\pi : \mathcal{C} \to \mathcal{C}$ donné par $\pi(x, y) = (x^q, y^q)$, qui se prolonge naturellement sur $\mathcal{J}$.
Les groupes notés $\mathbb{G}_1$ et $\mathbb{G}_2$ sont alors :

$$\mathbb{G}_1 = \mathcal{J}[r] \cap \mathrm{Ker}(\pi - [1]) = \mathcal{J}(\mathbb{F}_q)[r] \text{ et } \mathbb{G}_2 = \mathcal{J}(\mathbb{F}_{q^k})[r] \cap \mathrm{Ker}(\pi - [q]).$$

Ainsi, $\mathbb{G}_1 \neq \mathbb{G}_2$ sont deux $\mathbb{Z}/r\mathbb{Z}$ espaces vectoriels de dimension au moins 1, et $\mathbb{G}_1 \times \mathbb{G}_2 \subset \mathcal{J}(\mathbb{F}_{q^k})$ est au moins de dimension 2.

Dans la suite, $\mathbb{G}_1$ et $\mathbb{G}_2$ désigneront toujours les groupes introduits dans la définition 1.2.5, et on aura toujours $D_1 \in \mathbb{G}_1$ et $D_2 \in \mathbb{G}_2$.

Voyons maintenant l'algorithme de Miller ([33]) pour calculer $f_{r,D_1}(D_2)$.

Définition 1.2.6. Pour tout $i \in \mathbb{N}$, notons $D_i = \rho(iD)$. Par définition, $D_i \sim iD$, et la i-ième *fonction de Miller* $f_{i,D}$ est définie par

$$\mathrm{div}(f_{i,D}) = iD - D_i.$$

Le but est alors de calculer $f_{r,D}$, ce qui se fait par un algorithme double-and-add, via la relation

$$f_{i+j,D} = f_{i,D} \cdot f_{j,D} \cdot h_{D_i,D_j},$$

où h_{D_i,D_j} est la fonction apparaissant dans l'addition de D_i et D_j :

$$\mathrm{div}(h_{D_i,D_j}) = D_i + D_j - \rho(D_i + D_j).$$

1.2.4 Une variante : le couplage ate

Il s'agit d'une variante, plus rapide à calculer, du couplage de Tate. Avant de la présenter, il faut commencer par évoquer la normalisation des fonctions de Miller.

Définition 1.2.7. Soit $f \in \mathbb{K}(\mathcal{C})$ avec $v_\infty(f) = -n$. Soit u_∞ une uniformisante en ∞.[6]
Le *coefficient dominant* de f en ∞ est

$$lc_\infty(f) = (u_\infty^n f)(\infty).$$

Ainsi, si f est définie en ∞, on a simplement $lc_\infty(f) = f(\infty)$.

Alors la *normalisation* de f est le multiple de f

$$f^{norm} = f/lc_\infty(f)$$

qui a pour coefficient dominant 1 en l'infini.

Muni de cette définition, voici le couplage ate :

Théorème 1.2.8. *L'application*

$$\begin{aligned} a : \mathbb{G}_2 \times \mathbb{G}_1 &\to \mu_r \\ (D_2, D_1) &\mapsto f_{q,\rho(D_2)}^{norm}(\epsilon(D_1)) \end{aligned}$$

est un couplage bilinéaire non dégénéré. .
Dans le cas malheureux où

$$supp(\epsilon(D_1)) \cap supp(\rho(D_2)) \neq \varnothing,$$

on est obligé de remplacer $\epsilon(D_1)$ par un autre diviseur $D' \sim D_1$.

Définition 1.2.9. Ce couplage $a(D_2, D_1)$ est appelé *couplage ate.*

Voyez les deux grandes différences entre Tate et ate : dans le premier couplage, il faut calculer $f_{r,D}$, tandis que dans le second on ne va que jusqu'à $f_{q,D}$. Je rappelle que $r|\sharp\mathcal{J}(\mathbb{F}_q) \approx q^g$, ainsi la boucle de Miller est de l'ordre de g fois plus courte pour le calcul de ate que pour le calcul de Tate.
Ensuite, on remarquera qu'il n'y a pas d'exponentiation finale pour le calcul de ate.

Remarque 1.2.10. *Toujours dans l'optique de diminuer la taille de la boucle de Miller, d'autres variantes de Tate et ate ont été introduit, tels que les couplages R-ate, ou ate-i. Je ne le présente pas ici : quel que soit le couplage choisi, il est nécessaire de calculer une quantité du type*

$$f_{s,D}(D').$$

6. par exemple, on peut prendre $u_\infty = x^{2g}/y$.

1.3 Fonctions thêta

On a vu précédemment que l'algorithme de Miller est le moyen traditionnel de calcul de couplages. Je présente maintenant la théorie des fonctions thêta, qui est à la base de la méthode alternative introduite par Lubicz et Robert dans [30]. Pour plus de clarté, je présente d'abord ces outils en genre 1, avant de généraliser à g quelconque. La référence classique est [34].

1.3.1 Le cas elliptique

1.3.1.1 Définition et premières propriétés

Provisoirement, je me place sur une courbe elliptique $\mathcal{E}$ définie sur le corps des complexes $\mathbb{C}$. Je rappelle que cela signifie qu'il y a un réseau $\Lambda \subset \mathbb{C}$ tel que

$$\mathcal{E} \cong \mathbb{C}/\Lambda.$$

Concrètement, Λ est défini par :

$$\Lambda = (\mathbb{Z} + \tau\mathbb{Z}),$$

où $\tau \in \mathbb{C}$ est traditionnellement choisi avec $\mathrm{Im}(\tau) > 0$.
Pour $z \in \mathbb{C}$, $z \pmod{\Lambda}$ est le représentant de z dans le parallélogramme fondamental, c'est-à-dire celui engendré par 1 et τ.

Proposition 1.3.1. *Soit $a, b \in \mathbb{Q}$. La série*

$$\theta\begin{bmatrix} a \\ b \end{bmatrix}(z,\tau) = \sum_{n\in\mathbb{Z}} \exp\left(\pi i(a+n)^2\tau + 2\pi i(n+a)(z+b)\right)$$

définit une fonction holomorphe sur $\mathbb{C}$.

Définition 1.3.2. Cette fonction est appelée *fonction thêta avec caractéristiques* a et b.
L'évaluation d'une fonction thêta en $z = 0$ est appelée une *constante thêta*.

Remarque 1.3.3.

- *Si $a = b = 0$, $\theta\begin{bmatrix} 0 \\ 0 \end{bmatrix}$ est simplement noté θ.*
- *En pratique, on considérera $a, b \in \{0, \frac{1}{2}\}$, c'est-à-dire les fonctions thêta avec caractéristiques semi entières.*

Parmi les fonctions thêta, il y en a une particulière qu'on utilisera souvent : la fonction sigma, dont voici la définition :

Définition 1.3.4. Le produit infini

$$z \prod_{\substack{\omega \in \Lambda \\ \omega \neq 0}} \left(1 - \frac{z}{\omega}\right) e^{(z/\omega)+\frac{1}{2}(z/\omega)^2}$$

définit une fonction holomorphe sur tout $\mathbb{C}$ appelée *fonction σ de Weierstrass* (attachée à $\mathcal{E}$).

C'est une fonction θ dans le sens où il existe des constantes (que je ne précise pas) C et K telles que :

$$\sigma(z) = C.\exp\left(K.z^2\right) \theta\begin{bmatrix}1/2\\1/2\end{bmatrix}(z,\tau).$$

L'importance de cette fonction vient du fait qu'elle permet de construire n'importe quel fonction elliptique. Plus de détails sur σ pourront être trouvées dans [42], chapitre 6.

Proposition 1.3.5. *Pour tout $n \in \mathbb{Z}$,*

$$\theta\begin{bmatrix}a\\b\end{bmatrix}(z+n,\tau) = \exp(2\pi i a n)\ \theta\begin{bmatrix}a\\b\end{bmatrix}(z,\tau),$$

et

$$\theta\begin{bmatrix}a\\b\end{bmatrix}(z+n\tau,\tau) = \exp\left(-2\pi i b n - \pi i n^2\tau - 2\pi i n z\right)\ \theta\begin{bmatrix}a\\b\end{bmatrix}(z,\tau).$$

Définition 1.3.6. Toute fonction vérifiant les égalités de la proposition 1.3.5 est dite *quasi périodique* modulo Λ.

En particulier, cette proposition signifie que les fonctions θ ne sont pas des fonctions elliptiques :
si $z \equiv z' \pmod{\Lambda}$, on n'a pas $\theta\begin{bmatrix}a\\b\end{bmatrix}(z,\tau) = \theta\begin{bmatrix}a\\b\end{bmatrix}(z',\tau)$, parce que ces deux termes diffèrent d'un terme exponentiel.
Par contre, ce terme exponentiel n'empêche pas leur lieu d'annulation d'être bien défini sur $\mathcal{E}$:

Proposition 1.3.7. *On fixe les caractéristiques a et b.*
Alors, dans le parallélogramme fondamental, $\theta\begin{bmatrix}a\\b\end{bmatrix}(z,\tau)$ admet comme seul zéro le point $(a+\frac{1}{2})+(b+\frac{1}{2})\tau$.

Remarque 1.3.8. *En particulier,* $\theta\begin{bmatrix}1/2\\1/2\end{bmatrix}$ *(et donc la fonction* σ*) admet un zéro simple en chaque point du réseau* Λ *et nulle part ailleurs.*
Chacune des trois autres $\theta\begin{bmatrix}a\\b\end{bmatrix}$ *avec* $a, b \in \{0, \frac{1}{2}\}$ *semi entières s'annule sur un des trois autres points de 2-torsion de* $\mathcal{E}$*.*

Pour $P \in \mathcal{E}$, comme la quantité $\theta\begin{bmatrix}a\\b\end{bmatrix}(P)$ seule n'a pas de sens, la stratégie de [30] est alors de calculer ensembles plusieurs évaluations de fonctions thêta bien choisies.

Définition 1.3.9. Soit $l \in \mathbb{N}$. Une fonction f sur $\mathbb{C}$ est quasi Λ-périodique de niveau l si, pour tout $z \in \mathbb{C}$ et $n \in \mathbb{Z}$,

$$f(z+n) = f(z), \text{ and } f(z+n\tau) = \exp\left(-\pi i l n^2 \tau - 2\pi i l n z\right) f(z).$$

L'ensemble de ces fonctions est un espace vectoriel de dimension finie noté R_l^τ.

Les fonctions thêta avec caractéristiques nous donnent des bases de ces espaces :

Proposition 1.3.10. *[[34] p. 124] On fixe un niveau* $l \in \mathbb{N}$*.*

- *Les ensembles*
$$\left\{\theta\begin{bmatrix}a/l\\0\end{bmatrix}(lz, l\tau) \mid 0 \le a \le l-1\right\} \text{ et } \left\{\theta\begin{bmatrix}0\\b/l\end{bmatrix}\left(z, l^{-1}\tau\right) \mid 0 \le b \le l-1\right\}$$
sont deux bases de R_l^τ*.*
- *Si* $2|l$*, alors l'ensemble*
$$\left\{\left(\theta\begin{bmatrix}0\\b/l\end{bmatrix}(z, 2\tau/l)\right)^2 \mid 0 \le b \le l-1\right\}$$
est une autre base de R_l^τ*.*
- *Enfin, si* $l = k^2$ *est un carré, alors une base alternative de* R_l^τ *est :*
$$\left\{\theta\begin{bmatrix}a/k\\b/k\end{bmatrix}(kz, \tau) \mid 0 \le a, b \le k-1\right\}.$$

Remarque 1.3.11. *Pour faire le lien avec ce qui a été rappelé en 1.1, notons que* R_l^τ *est isomorphe à l'espace de Riemann-Roch* $\mathcal{L}(l\mathcal{O})$*.*

J'ai dit avant la définition 1.3.9 qu'il allait nous falloir évaluer plusieurs fonctions thêta d'un coup en un point de la courbe. La proposition suivante nous explique comment :

Proposition 1.3.12. *[[34] p. 127] On fixe un niveau $l \in \mathbb{N}$. Soit $\phi_1^{(l)}, \ldots, \phi_l^{(l)}$ une base de R_l^τ. Alors la fonction suivante est bien définie :*

$$\begin{aligned} \Phi_l : \mathbb{C}/\Lambda &\to \mathbb{P}^{l-1} \\ z &\mapsto \left(\phi_1^{(l)}(z), \ldots, \phi_l^{(l)}(z)\right). \end{aligned}$$

Si $l \geq 3$, Φ_l est un plongement.
Ce n'est pas tout à fait le cas pour $l = 2$: Φ_2 est en fait un plongement de $\mathcal{K}$ vers $\mathbb{P}^1$, où $\mathcal{K}$ désigne la variété de Kummer *associée à $\mathcal{E}$, i.e. $\mathcal{K} \cong \mathcal{E}/\{\pm 1\}$.*

En pratique, on utilise les fonctions thêta de niveau 2 et 4. Les relations de Riemann (voir [34] p.20) nous donnent alors des formules d'addition satisfaites par les fonctions thêta avec caractéristiques semi entières. On a aussi la proposition suivante :

Proposition 1.3.13. *Soit $z_1, z_2 \in \mathbb{C}$. Pour tout $b \in \{0, 1/2\}$,*

$$\theta\begin{bmatrix}0\\b\end{bmatrix}(z_1+z_2;\tau)\theta\begin{bmatrix}0\\b\end{bmatrix}(z_1-z_2;\tau) = \sum_{a\in\{0,1/2\}} (-1)^{4ab}\theta\begin{bmatrix}a\\0\end{bmatrix}\left(z_1;\frac{\tau}{2}\right)\theta\begin{bmatrix}a\\0\end{bmatrix}\left(z_2;\frac{\tau}{2}\right).$$

Résumons ce que l'on a énoncé sur les fonctions thêta.
Soit $P, Q \in \mathcal{E}$. La proposition 1.3.12 nous explique que ces points peuvent être représentés par l'évaluation de certaines fonctions thêta bien choisies en eux.
La proposition 1.3.13 nous explique comment représenter le point $P+Q$, dès lors qu'on a la représentation des points P, Q et $P-Q$ via les fonctions thêta.
Avant de rappeler le calcul de couplages expliqué dans [30], je donne quelques formules pour expliciter mon propos.

Remarque 1.3.14. *On aura remarqué que j'ai présenté ici la théorie classique des fonctions thêta, définies sur $\mathbb{C}$. Evidemment, pour une utilisation cryptographique, il est nécessaire de la transposer aux corps finis. Cela se fait via le principe de Lefschetz. Ainsi, les propositions 1.3.10, 1.3.12 et 1.3.13 restent vraies pour une courbe $\mathcal{E}$ définie sur un corps fini. Comment se calculent alors les fonctions thêta ? C'est ce que j'énonce dans la proposition suivante 1.3.15.*

1.3.1.2 Arithmétique des fonctions thêta

Dans cette section, on travaille sur un corps fini $\mathbb{F}_q$. On supposera toujours que la caractéristique p de $\mathbb{F}_q$ est différente de 2.
Tout d'abord, voyons comment faire le passage entre la représentation de Weierstrass usuelle d'une courbe et sa représentation par les fonctions thêta. La proposition suivante est le cas $g = 1$ du théorème 7.6 de [35] (p. 3.113).

Proposition 1.3.15. *Soit $\mathcal{E}/\mathbb{F}_q$ une courbe elliptique. On suppose que $\mathcal{E}$ est donnée par une équation de Weierstrass*

$$y^2 = (x-a_1)(x-a_2)(x-a_3).$$

Si tous les $a_j - a_i$, $1 \leq i < j \leq 3$ sont des carrés dans $\mathbb{F}_q$, alors on a :

— *si $P \in \mathcal{E}\backslash\mathcal{O}$:*

$$\left(\frac{\theta\begin{bmatrix}0\\0\end{bmatrix}(P,\tau)}{\theta\begin{bmatrix}1/2\\1/2\end{bmatrix}(P,\tau)}\right)^2 = \frac{x_P - a_2}{\sqrt{(a_2-a_1)(a_3-a_2)}},$$

$$\left(\frac{\theta\begin{bmatrix}1/2\\0\end{bmatrix}(P,\tau)}{\theta\begin{bmatrix}1/2\\1/2\end{bmatrix}(P,\tau)}\right)^2 = \frac{x_P - a_3}{\sqrt{(a_3-a_1)(a_3-a_2)}},$$

$$\left(\frac{\theta\begin{bmatrix}0\\1/2\end{bmatrix}(P,\tau)}{\theta\begin{bmatrix}1/2\\1/2\end{bmatrix},\tau)}\right)^2 = \frac{x_P - a_1}{\sqrt{(a_3-a_1)(a_2-a_1)}};$$

— *si $P = \mathcal{O}$:*

$$\left(\frac{\theta\begin{bmatrix}0\\0\end{bmatrix}\mathcal{O},\tau)}{\theta\begin{bmatrix}1/2\\0\end{bmatrix}(\mathcal{O},\tau)}\right)^2 = \frac{\sqrt{a_3-a_1}}{\sqrt{a_2-a_1}},$$

$$\left(\frac{\theta\begin{bmatrix}0\\1/2\end{bmatrix}(\mathcal{O},\tau)}{\theta\begin{bmatrix}1/2\\0\end{bmatrix}(\mathcal{O},\tau)}\right)^2 = \frac{\sqrt{a_3-a_2}}{\sqrt{a_2-a_1}},$$

$$\theta\begin{bmatrix}1/2\\1/2\end{bmatrix}(\mathcal{O},\tau) = 0.$$

Les bases décrites dans la proposition 1.3.10 de niveau deux seront désignées ainsi :

— $[u_P : v_P] = \left[\theta\begin{bmatrix}0\\0\end{bmatrix}(P,\tau)^2 : \theta\begin{bmatrix}1/2\\0\end{bmatrix}(P,\tau)^2\right]$,

— $[u'_P : v'_P] = \left[\theta\begin{bmatrix}0\\0\end{bmatrix}(2P,2\tau) : \theta\begin{bmatrix}1/2\\0\end{bmatrix}(2P,2\tau)\right]$,

— $[\tilde{u}_P : \tilde{v}_P] = \left[\theta\begin{bmatrix}0\\0\end{bmatrix}(P,\tau/2) : \theta\begin{bmatrix}0\\1/2\end{bmatrix}(P,\tau/2)\right]$,

Et les constantes thêta ainsi :

— $a = \theta\begin{bmatrix}0\\0\end{bmatrix}(\mathcal{O},\tau/2)$, $b = \theta\begin{bmatrix}0\\1/2\end{bmatrix}(\mathcal{O},\tau/2)$;

— $\alpha = \theta\begin{bmatrix}0\\0\end{bmatrix}(\mathcal{O},2\tau)$, $\beta = \theta\begin{bmatrix}1/2\\0\end{bmatrix}(\mathcal{O},2\tau)$;

En niveau 4, la même proposition 1.3.10 nous donne la base suivante :

$$[X_P : Y_P : Z_P : T_P] = \left[\theta\begin{bmatrix}0\\0\end{bmatrix}(2P,\Omega) : \theta\begin{bmatrix}1/2\\0\end{bmatrix}(2P,\Omega) : \theta\begin{bmatrix}0\\1/2\end{bmatrix}(2P,\Omega) : \theta\begin{bmatrix}1/2\\1/2\end{bmatrix}(2P,\Omega)\right].$$

Les constantes thêta sont désignées ainsi :

$$A = \theta\begin{bmatrix}0\\0\end{bmatrix}(\mathcal{O},\Omega),$$
$$B = \theta\begin{bmatrix}1/2\\0\end{bmatrix}(\mathcal{O},\Omega),$$
$$C = \theta\begin{bmatrix}0\\1/2\end{bmatrix}(\mathcal{O},\Omega).$$

Remarque 1.3.16. *Notons que* $\theta\begin{bmatrix}1/2\\1/2\end{bmatrix}(\mathcal{O},\Omega) = 0$.

La proposition 1.3.13 nous donne les formules de changement de bases suivantes :

Proposition 1.3.17. *Pour tout point* $P \in \mathcal{E}$,

— $u_P = \frac{1}{2}(a\tilde{u}_P + b\tilde{v}_P)$, $v_P = \frac{1}{2}(a\tilde{u}_P - b\tilde{v}_P)$;

— $u'_P = \frac{1}{2}(\tilde{u}_P + \tilde{v}_P)$, $v'_P = \frac{1}{2}(\tilde{u}_P - \tilde{v}_P)$.

En particulier, en prenant $P = \mathcal{O}$, *on obtient les relations entre les constantes thêta.*

Les formules d'addition que vérifient les bases de niveau 2 décrivent l'arithmétique de $\mathcal{K}$:

Proposition 1.3.18. *Soient* P *et* $Q \in \mathcal{E}$.

— *pour la base* (u,v), *on a :*

$$u_{P+Q}u_{P-Q} = \left((u_P+v_P)(u_Q+v_Q) + \frac{\alpha^2}{\beta^2}(u_P-v_P)(u_Q-v_Q)\right)^2,$$
$$v_{P+Q}v_{P-Q} = \left((u_P+v_P)(u_Q+v_Q) - \frac{\alpha^2}{\beta^2}(u_P-v_P)(u_Q-v_Q)\right)^2;$$

— *pour* (u', v') *:*

$$u'_{P+Q}u'_{P-Q} = (u'^2_P + v'^2_P)(u'^2_Q + v'^2_Q) + \frac{A^2}{B^2}(u'^2_P - v'^2_P)(u'^2_Q - v'^2_Q),$$
$$v'_{P+Q}v'_{P-Q} = (u'^2_P + v'^2_P)(u'^2_Q + v'^2_Q) - \frac{A^2}{B^2}(u'^2_P - v'^2_P)(u'^2_Q - v'^2_Q);$$

— *enfin, pour* $(\tilde{u}, \tilde{v})$ *:*

$$\tilde{u}_{P+Q}\tilde{u}_{P-Q} = (\tilde{u}^2_P + \tilde{v}^2_P)(\tilde{u}^2_Q + \tilde{v}^2_Q) + \frac{A^2}{B^2}(\tilde{u}^2_P - \tilde{v}^2_P)(\tilde{u}^2_Q - \tilde{v}^2_Q),$$
$$\tilde{v}_{P+Q}\tilde{v}_{P-Q} = (\tilde{u}^2_P + \tilde{v}^2_P)(\tilde{u}^2_Q + \tilde{v}^2_Q) - \frac{A^2}{B^2}(\tilde{u}^2_P - \tilde{v}^2_P)(\tilde{u}^2_Q - \tilde{u}^2_Q).$$

Regardons maintenant ce qui se passe en niveau 4 :
La courbe $\mathcal{E}$ peut alors être décrite avec les fonctions de niveau 4 par le théorème suivant :

Théorème 1.3.19. *On reprend notre courbe* $\mathcal{E}$ *d'équation* $y^2 = (x - a_1)(x - a_2)(x - a_3)$*. On demande maintenant à ce que les* $a_j - a_i$ *soient des puissances quatrièmes dans* $\mathbb{F}_q$*. Alors, l'application*

$$\begin{array}{ccc} \mathcal{E} & \to & \mathcal{T} \\ P = [x_P : y_P] & \mapsto & [X_P : Y_P : Z_P : T_P] \\ \mathcal{O} & \mapsto & [A : B : C : 0] \end{array}$$

établit une équivalence birationnelle entre $\mathcal{E}$ *et la courbe*

$$\mathcal{T} : \left\{ \begin{array}{l} X^2A^2 = Z^2C^2 + Y^2B^2 \\ T^2A^2 = Z^2B^2 - Y^2C^2 \end{array} \right.$$

Explicitement, ce morphisme est :

$$X_P = (x_P - a_2)^2 + (a_2 - a_1)(a_3 - a_2),$$
$$Y_P = (x_P - a_1)^2 + (a_3 - a_2)(a_3 - a_1),$$
$$Z_P = (x_P - a_3)^2 + (a_3 - a_1)(a_2 - a_1),$$
$$T_P = \frac{((a_3 - a_1)(a_3 - a_2)(a_2 - a_1))^{\frac{1}{4}}}{2} y_P.$$

Proposition 1.3.20. *Les formules d'addition sur* $\mathcal{T}$ *sont données par les relations de Riemann :*

$$X_{P+Q}X_{P-Q}A^2 = X^2_P X^2_Q + T^2_P T^2_Q$$
$$Y_{P+Q}Y_{P-Q}B^2 = X^2_P X^2_Q - Z^2_P Z^2_Q$$

$$Z_{P+Q}Z_{P-Q}C^2 = X_P^2X_Q^2 - Y_P^2Y_Q^2$$
$$T_{P+Q}T_{P-Q}A^2 = T_P^2X_Q^2 - X_P^2T_Q^2$$

Pour le doublement, si on remplace Q par P dans la dernière équation, on n'obtient rien d'autre que $0 = 0$. À la place, il faut utiliser la formule suivante :

$$T_{2P}ABC = 2X_PY_PZ_PT_P$$

Remarque 1.3.21.

— *Sur $\mathcal{T}$, l'opposé du point $[X : Y : Z : T]$ est $[X : Y : Z : -T]$.*

— *Le lien entre niveau 2 et niveau 4 peut se faire via les relations de Riemann : par exemple*

$$\begin{aligned} XA^3 &= u^2 + t^2 = v^2 + w^2 \\ YB^3 &= u^2 - w^2 = v^2 - t^2 \\ ZC^3 &= u^2 - v^2 = w^2 - t^2 \\ TA^2B^2C^2 &= 4uvwt. \end{aligned}$$

L'application $[X : Y : Z : T] \mapsto [u : v]$ est donc bien $2 : 1$, et la fibre au dessus de chaque point $[u : v]$ consiste en un couple

$$\{[X : Y : Z : T], [X : Y : Z : -T]\}.$$

1.3.2 Fonctions thêta en tout genre

Après cette longue introduction en genre 1, il va être plus simple de comprendre la généralisation au cas hyperelliptique. Le parcours est le même : je commence par rappeler ce que sont les fonctions thêta, j'évoque ensuite les fonctions de niveau l, et finalement j'explique les coordonnées thêta, c'est-à-dire le plongement projectif de la jacobienne (ou de la kummer) donné par les fonctions thêta. Encore une fois, plus de détails se trouveront dans [34].

Comme en genre 1, on commence par travailler sur $\mathbb{C}$ avant de revenir à nos corps finis. Ainsi, soit $\mathcal{C}/\mathbb{C}$ une courbe hyperelliptique de genre g. Alors

$$\mathcal{J} \cong \mathbb{C}^g / \left(\mathbb{Z}^g + \Omega\mathbb{Z}^g\right),$$

où Ω est une matrice symétrique de taille g, que l'on choisit avec $\mathrm{Im}(\Omega)$ définie positive. Le réseau $\mathbb{Z}^g + \Omega\mathbb{Z}^g$ est noté Λ.

Définition 1.3.22. Soient $a, b \in \mathbb{Q}^g$. La fonction thêta de caractéristiques rationnelles (a, b) est la fonction analytique sur $\mathbb{C}^g$ définie par la série :

$$\theta\begin{bmatrix} a \\ b \end{bmatrix}(z;\Omega) = \sum_{n\in\mathbb{Z}^g} \exp\left(\pi i^t(a+n)\Omega(n+a) + 2\pi i^t(n+a)(z+b)\right).$$

Exemple 1.3.23. *En genre* 1*, il y avait* 4 *fonctions thêta avec caractéristiques semi entières. En genre* 2*, on voit qu'il y en a maintenant* 16*.*

Toujours comme en genre 1, ces fonctions nous donnent des bases des espaces de fonctions de niveau l (c'est l'équivalent de la proposition 1.3.10) :

Définition 1.3.24. Soit $l \in \mathbb{N}$. Une fonction f sur $\mathbb{C}^g$ est quasi Λ-périodique de niveau l si, pour tout $z \in \mathbb{C}^g$ et $m, m' \in \mathbb{Z}^g$,

$$f\left(z + m + \Omega m'\right) = f(z)\exp\left(-\pi i l^t m' \Omega m' - 2\pi i l^t z m'\right).$$

Proposition 1.3.25 ([34] p. 124)**.** *L'ensemble des fonctions de niveau l forme un espace vectoriel de dimension l^g noté R_l^Ω.*
Fixons un niveau l. Les fonctions thêta nous donnent plusieurs bases de R_l^Ω. On a par exemple les bases suivantes :

- $\left\{\theta\begin{bmatrix}a\\0\end{bmatrix}(lz, l\Omega) \mid a \in \left(\frac{1}{l}\mathbb{Z}^g/\mathbb{Z}^g\right)\right\}$,
- $\left\{\theta\begin{bmatrix}0\\b\end{bmatrix}(z, l^{-1}\Omega) \mid b \in \left(\frac{1}{l}\mathbb{Z}^g/\mathbb{Z}^g\right)\right\}$,
- *si* $2|l$, $\left\{\theta\begin{bmatrix}a\\b\end{bmatrix}\left(2z, \frac{4\Omega}{l}\right) \mid a \in \left(\frac{1}{2}\mathbb{Z}^g/\mathbb{Z}^g\right),\ b \in \left(\frac{2}{l}\mathbb{Z}^g/\mathbb{Z}^g\right)\right\}$,
- *si* $l = k^2$, $\left\{\theta\begin{bmatrix}a\\b\end{bmatrix}(kz, \Omega) \mid a, b \in \left(\frac{1}{k}\mathbb{Z}^g/\mathbb{Z}^g\right)\right\}$.

Remarque 1.3.26. *Pour $g = 1$, on avait vu le lien entre les fonctions de niveau l et les espaces de Riemann Roch. Pour g quelconque, ce lien demeure :*

$$R_l^\Omega = \mathcal{L}(l\Theta).$$

Voici maintenant l'équivalent de la proposition 1.3.12 : le plongement projectif induit par les fonctions thêta.

Proposition 1.3.27 ([34] p. 127)**.** *On fixe un niveau $l \in \mathbb{N}$. Soit $\phi_1^{(l)}, \ldots, \phi_{l^g}^{(l)}$ une base de R_l^Ω. Alors la fonction suivante est bien définie :*

$$\begin{aligned}\Phi_l : \mathbb{C}^g/\Lambda &\to \mathbb{P}^{l^g-1}\\ z &\mapsto \left(\phi_1^{(l)}(z), \ldots, \phi_{l^g}^{(l)}(z)\right).\end{aligned}$$

Si $l \geq 3$, Φ_l est un plongement.
Ce n'est pas le cas pour $l = 2$: Φ_2 est en fait un plongement de $\mathcal{K}$ vers $\mathbb{P}^{2^g-1}$, où $\mathcal{K}$ est la variété de Kummer *associée à $\mathcal{J}$:*

$$\mathcal{K} \simeq \mathcal{J}/\{\pm 1\}.$$

Exemple 1.3.28. *Les formules d'addition vérifiées par les fonctions thêta en genre 2, et les algorithmes dérivés se trouvent par exemple dans [21].*

Enfin, faisons le lien entre les fonctions θ et le diviseur Θ :

Proposition 1.3.29 ([35] pp. 3.80-3.82)**.** *Soit*

$$\delta = {}^t\left(\frac{1}{2},\dots,\frac{1}{2}\right),\ \delta' = {}^t\left(\frac{g}{2},\frac{g-1}{2},\dots,\frac{1}{2}\right) \in \mathbb{Q}^g.$$

On note

$$\theta\begin{bmatrix}\delta\\ \delta'\end{bmatrix}.$$

Alors

$$\mathrm{div}(\theta') = \Theta$$

Exemple 1.3.30. *Pour $g = 1$, on a $\theta' = \theta\begin{bmatrix}1/2\\ 1/2\end{bmatrix}$, et on retrouve le fait que θ' admet un unique zéro simple en $\mathcal{O}$.*

Je rappelle que quand on étudiait le cas $g = 1$, on avait spécifié une fonction thêta spéciale : la fonction sigma de Weierstrass.
Quand $g > 1$, son équivalent s'appelle fonction sigma de Klein. Comme pour $g = 1$, elle est reliée à θ' :

Définition 1.3.31 ([39])**.** Pour définir sigma, on a besoin des différentielles suivantes : pour $1 \leq j \leq g$

$$\eta_j = \frac{1}{2y}\sum_{k=j}^{2g-j}(k+1-j)f_{k+1+j}x^k dx,$$

où les f_i sont les coefficients de l'équation de $\mathcal{C}$:

$$y^2 = f(x) = x^{2g+1} + f_{2g}x^{2g} + \cdots + f_0.$$

On réunit ces différentielles dans un vecteur : $\eta = (\eta_j)_{1\leq j\leq g}$. Alors, il y a une constante c telle que la fonction sigma hyperelliptique soit la fonction de $\mathbb{C}^g$ définie par :

$$\sigma(z) = c\exp(\frac{1}{2}{}^t z\eta z)\theta\begin{bmatrix}\delta\\ \delta'\end{bmatrix}(z;\Omega)\,.$$

Une dernière proposition utile à garder en tête est la suivante :

Proposition 1.3.32 ([34] proposition 3.14 p.167)**.**
La fonction $\theta\begin{bmatrix}a\\b\end{bmatrix}$ *est paire ou impaire si et seulement si ses caractéristiques* a *et* b *sont semi entières.*
Elle est alors de la parité de $4^t ab$.

En particulier, σ est impaire pour $g \equiv 1, 2 \pmod 4$, et paire dans les deux autres cas. Étudions encore un peu σ avant d'aller plus loin :

Définition 1.3.33. Pour $i, j, k \in \{1, 2, \dots, g\}$ et $u \in \mathbb{C}^g$, les $\wp-$fonctions hyperelliptiques sont définies de la façon suivante :

$$\wp_{ij}(u) = -\frac{\partial^2}{\partial u_i \partial u_j}\log\sigma(u), \;\; \wp_{ijk}(u) = -\frac{\partial^3}{\partial u_i \partial u_j \partial u_k}\log\sigma(u).$$

Ce sont des fonctions bien définies sur $\mathcal{J} \simeq \mathbb{C}^g/\Lambda$.

Proposition 1.3.34. *[10]*
Soit $D_1, D_2 \in \mathcal{J}$*. Il existe des polynômes*

$$\mathcal{F}_g(D_1, D_2) \text{ et } \mathcal{G}_g(D_1) \in \mathbb{Z}[\wp_{ij}(D_1), \wp_{ijk}(D_1), \wp_{ij}(D_2), \wp_{ijk}(D_2)]$$

tels que

$$\frac{\sigma(D_1 + D_2)\sigma(D_1 - D_2)}{\sigma(D_1)^2\sigma(D_2)^2} = \mathcal{F}_g(D_1, D_2).$$
$$\frac{\sigma(2D_1)}{\sigma(D_1)^4} = \mathcal{G}_g(D_1).$$

Remarque 1.3.35.

- *Les* $\wp$ *fonctions sont un moyen alternatif de représenter les éléments de* $\mathcal{J}$*. Elles sont reliées à la représentation de Mumford via*

$$u = x^g - \sum_{l=1}^{g} \wp_{lg}(D)x^{l-1}, \;\; v = \sum_{l=1}^{g} \frac{\wp_{lgg}(D)}{2}x^{l-1}.$$

  Plus de détails se trouveront dans [9].

- *En réalité, cette propriété n'est pas simplement une propriété d'existence : les polynômes* $\mathcal{F}_g$ *sont bien construits. Comme je n'ai pas besoin de leurs valeurs exactes, je ne les donne pas ici. Elles se trouvent par exemple dans [10].*

Le corollaire suivant est à la base de la théorie des hyperelliptic nets :

Corollaire 1.3.36. *[50] Soit* $m > 2^g$ *un entier. On se donne* m *points* $u^{(1)}, \dots, u^{(m)} \in \mathbb{C}^g$*. La matrice*

$$\left(\sigma(u^{(i)} + u^{(j)})\sigma(u^{(i)} - u^{(j)})\right)_{1 \le i,j \le m}$$

est de déterminant nul.

1.4 La théorie des hyperelliptic nets

Dans cette section, je rappelle une autre théorie permettant le calcul des couplages : celle des hyperelliptic nets[7]. Comme je l'ai fait pour les fonctions thêta, je commencerai par présenter le cas elliptique avant sa généralisation aux genres supérieurs.

1.4.1 Le cas elliptique

Cette section rappelle les résultats de Stange dans [45].

Définition 1.4.1. Soit $\mathbb{G}$ un groupe abélien libre de type fini, et $\mathbb{A}$ un anneau intègre. Une fonction $W : \mathbb{G} \to \mathbb{A}$ est appelé *elliptic net* si pour tout quadruplet $(a, b, c, d) \in \mathbb{G}^4$:

$$\begin{aligned} W(a+b)W(a-b)W(c+d)W(c-d) \\ + W(a+c)W(a-c)W(d+b)W(d-b) \\ + W(a+d)W(a-d)W(b+c)W(b-c) = 0. \end{aligned}$$

Remarque 1.4.2. *On notera pour faire le lien avec le cas hyperelliptique que ceci est équivalent à demander à ce que pour tout quadruplet* $(u_1, \ldots, u_4) \in \mathbb{G}^4$*, on ait*

$$\det\left(W(u_i+u_j)W(u_i-u_j)\right)_{1\leq i,j\leq 4} = 0.$$

Dans un cadre cryptographique, et dans le but de calculer des couplages, $\mathbb{G}$ sera $\mathbb{Z}^2$ et $\mathbb{A}$ le corps sur lequel l'on travaille : $\mathbb{C}$ dans un premier temps pour mettre en place la théorie, un corps fini $\mathbb{F}_{q^k}$ ensuite pour faire les calculs en pratique.

Soit donc $\mathcal{E}$ une courbe elliptique, que je suppose pour le moment définie sur $\mathbb{C}$. Utilisant la fonction σ de Weierstrass, Stange commence par construire un premier elliptic net :

Proposition 1.4.3 ([45])**.** *Pour tout couple* $\mathbf{v} = (v_1, v_2) \in \mathbb{Z}^2$*,*

$$\Psi_{\mathbf{v}}(z_1, z_2) = \frac{\sigma(v_1 z_1 + v_2 z_2)}{\sigma(z_1)^{v_1^2 - v_1 v_2}\sigma(z_1+z_2)^{v_1 v_2}\sigma(z_2)^{v_2^2 - v_1 v_2}}$$

est une fonction bien définie $\mathcal{E} \times \mathcal{E}$*.*

7. dans ce document, le terme hyperelliptic nets regroupera les cas $g = 1$ et $g > 1$.

Remarque 1.4.4.

- *Je précise le sens de cette proposition : à priori σ (comme les autres fonctions theta) est définie sur $\mathbb{C}$, et quasi Λ-périodique. Cette proposition signifie qu'en prenant un quotient bien choisi de fonctions sigma, on construit une fonction bien définie sur $\mathcal{E}$, c'est-a-dire qui soit vraiment Λ-périodique.*
- *Dans son article [45], Stange définit des nets sur $\mathbb{Z}^n$ en toute généralité. Mais comme je l'ai signalé précédemment, pour le calcul de pairings, nous n'aurons besoin que du cas $n = 2$.*

Ainsi, dans la proposition précédente, le couple $\mathbf{v} = (v_1, v_2) \in \mathbb{Z}^2$ est fixé pour construire une fonction $\Psi_{\mathbf{v}}$ sur $\mathcal{E} \times \mathcal{E}$. L'idée maintenant est d'inverser les rôles des z_i et des v_i :

Proposition 1.4.5 ([45])**.** *Fixons donc z_1 et $z_2 \in \mathbb{C}$, avec z_1, z_2 et $z_1 + z_2 \notin \Lambda$. La fonction*

$$\begin{aligned} W : \mathbb{Z}^2 &\to \mathbb{C} \\ \mathbf{v} &\mapsto \Psi_{\mathbf{v}}(z_1, z_2) \end{aligned}$$

est un elliptic net.

Remarque 1.4.6. *Remarquons dès à présent qu'elle vérifie :*

$$W(1, 0) = W(0, 1) = W(1, 1) = 1.$$

Je rappelle que σ admet un zéro simple en $\mathcal{O}$. Cette information permet d'avoir facilement le diviseur de $\Psi_{\mathbf{v}}$:

Théorème 1.4.7 ([45])**.** *Soit $s : \mathcal{E}^2 \to \mathcal{E}$ et, pour $i = 1$ or 2, $p_i : \mathcal{E}^2 \to \mathcal{E}$ respectivement la somme des composantes et la projection sur la i-ième composante. Leurs pullbacks sont notés s^* et p_i^*. Avec ces notations, le diviseur de $\Psi_{\mathbf{v}}$ est :*

$$D_{\mathcal{E},\mathbf{v}} = (([v_1] \times [v_2])^* s^* \mathcal{O}) - v_1 v_2 \, (s^* \mathcal{O}) - (v_1^2 - v_1 v_2) p_1^* \mathcal{O} - (v_2^2 - v_1 v_2) p_2^* \mathcal{O}.$$

Remarque 1.4.8. *En langage humain, ce théorème signifie en particulier que pour $(z_1, z_2) \in \mathbb{C}$ correspondant aux points $(P_1, P_2) \in \mathcal{E}$,*

$$W(v_1, v_2) = 0 \Leftrightarrow [v_1]P_1 + [v_2]P_2 = \mathcal{O}.$$

Ce théorème a des conséquences importantes. D'abord, Stange établit le théorème de transport de $\mathbb{C}$ vers les autres corps (et notamment les corps finis) suivant :

Théorème 1.4.9 ([45])**.** *Soit $\mathcal{E}$ une courbe elliptique défini sur un corps $\mathbb{K}$ quelconque. Soient P_1 et P_2 deux points de $\mathcal{E}$. Il existe un elliptic net*

$$W_{\mathcal{E},(P_1,P_2)} : \mathbb{Z}^2 \to \overline{\mathbb{K}}$$

avec les propriétés suivantes :
- *$W_{\mathcal{E},(P_1,P_2)}(0,1) = W_{\mathcal{E},(P_1,P_2)}(1,0) = W_{\mathcal{E},(P_1,P_2)}(1,1) = 1$;*
- *$W_{\mathcal{E},(P_1,P_2)}(v_1,v_2) = 0 \Leftrightarrow [v_1]P_1 + [v_2]P_2 = \mathcal{O}$ sur la courbe $\mathcal{E}$.*

Définition 1.4.10. Un elliptic net vérifiant les propriétés du théorème 1.4.9 est dit *associé à* la courbe $\mathcal{E}$ et aux points P_1 and P_2.

Ensuite, elle montra comment calculer les couplages avec ces nets. Je rappellerai ce théorème dans le chapitre 2, justement dédié aux couplages.
Passons maintenant à la généralisation de cette théorie aux genres plus grands.

1.4.2 Le cas hyperelliptique

Ici, je rappelle les résultats qu'Uchida et Uchiyama ont énoncé dans [50]. Le point de départ de leur article est de généraliser la proposition 1.4.3. Comme dans le travail de Stange, la théorie débute sur $\mathbb{C}$:

Proposition 1.4.11. *Soit $\mathcal{C}/\mathbb{C}$ une courbe hyperelliptique de genre g.*
Pour $\mathbf{v} = (v_1, v_2) \in \mathbb{Z}^2$,

$$\Psi_{\mathbf{v}}(z_1,z_2) = \frac{\sigma(v_1z_1 + v_2z_2)}{\sigma(z_1)^{v_1^2 - v_1v_2}\sigma(z_1+z_2)^{v_1v_2}\sigma(z_2)^{v_2^2-v_1v_2}}$$

est une fonction bien définie sur $\mathcal{J} \times \mathcal{J}$.

Remarque 1.4.12. *Les mêmes remarques que pour le genre 1 sont ici valables.*

Vient ensuite la généralisation du théorème 1.4.7 :

Théorème 1.4.13. *Soient $s : \mathcal{J}^2 \to \mathcal{J}$ et, pour $i = 1$ et 2, $p_i : \mathcal{J}^2 \to \mathcal{J}$ respectivement la somme de toutes les composantes et les projections sur la i-ième composante. Pour $\mathbf{v} \neq 0 \in \mathbb{Z}^2$, le diviseur de $\Psi_{\mathbf{v}}$ est :*

$$D_{\mathcal{C},\mathbf{v}} = (([v_1] \times [v_2])^* s^* \Theta) - v_1v_2\,((p_1^* \times p_2^*)s^*\Theta) - (v_1^2 - v_1v_2)p_1^*\Theta - (v_2^2 - v_1v_2)p_2^*\Theta.$$

Enfin, voici le théorème de transport (équivalent du théorème 1.4.9)

Théorème 1.4.14. *Soit $\mathcal{C}/\mathbb{K}$ une courbe hyperelliptique définie sur un corps quelconque $\mathbb{K}$, et D_1et D_2 deux diviseurs de $\mathcal{C}$. Alors il existe une application $W_{\mathcal{C},(D_1,D_2)} : \mathbb{Z}^2 \to \overline{\mathbb{K}}$ vérifiant les deux propriétés suivantes :*

- *$W_{\mathcal{C},(D_1,D_2)} = 0 \Leftrightarrow [v_1]D_1 + [v_2]D_2 \in \Theta$.*
- *Soit $m > 2^g$ un entier. On se donne $v_1, \dots v_m$, et $w_1, \dots w_m$ dans $1/2\mathbb{Z}$ tels que pour tout i et j, $v_i \pm v_j$ et $w_i \pm w_j \in \mathbb{Z}$. Alors, la matrice*

$$\left(W_{\mathcal{C},(D_1,D_2)}(v_i + v_j, w_i + w_j)W_{\mathcal{C},(D_1,D_2)}(v_i - v_j, w_i - w_j)\right)_{1\leq i,j\leq m}$$

est de déterminant nul.

Définition 1.4.15. La fonction $W_{\mathcal{C},(D_1,D_2)}$ établie dans le théorème précédent 1.4.14 est appelée *hyperelliptic net* associé à $\mathcal{C}$ et (D_1, D_2).

Exemple 1.4.16. *Pour $g = 1$ et $m = 4$, on retrouve la définition initiale des elliptic nets.*

Le fait que $\mathbf{v} \mapsto \Psi_{\mathbf{v}}$ soit un hyperelliptic net est une conséquence du corollaire 1.3.36. Plus précisément, et comme dans la proposition 1.4.5 en genre 1 :

Théorème 1.4.17. *Fixons z_1 et $z_2 \in \mathbb{C}^g$. La fonction*

$$\begin{aligned} W : \mathbb{Z}^2 &\to \mathbb{C} \\ \mathbf{v} &\mapsto \Psi_{\mathbf{v}}(z_1, z_2) \end{aligned}$$

est un hypelliptic net.

Chapitre 2

Deux méthodes de calculs de couplages

Dans le chapitre précédent, j'ai rappelé les théories des fonctions thêta et des hyperelliptic nets, sous-jacentes respectivement à [30] et [50]. Ici, je m'intéresse maintenant aux méthodes de calculs de couplages qui y sont exposées. Après avoir rappelé les résultats déjà existants (théorèmes 2.1.1 et 2.2.1 [1]), je donnerai mes propres résultats : l'algorithme de calcul de couplages en tout genre utilisant les hyperelliptic nets. Enfin, je ferai une comparaison des coûts théoriques en genre 1.
De prime abord, cette juxtaposition de deux théories à priori indépendantes pourra sembler décousue, mais ce sera justement l'objet du troisième chapitre d'expliquer les liens profonds les unissant.

2.1 Couplages et fonctions thêta

Maintenant que les définitions et propriétés des fonctions thêta ont été rappelées, voici le résultat principal de [30] :

Théorème 2.1.1 ([30], théorème 2)**.** *On se replace dans la situation de la définition 1.2.3 : $\mathcal{C}$ est une courbe hyperelliptique de genre g définie sur le corps $\mathbb{F}_q$, et $\mathcal{J}$ sa jacobienne ; r est un grand diviseur premier de $\sharp\mathcal{J}(\mathbb{F}_q)$ premier à q ; k est le degré de plongement ; D_1 est un élément de $\mathcal{J}(\mathbb{F}_{q^k})[r]$.*

On se donne $\theta\begin{bmatrix} a \\ b \end{bmatrix}$ de niveau l, où l et r sont premiers entre eux. Alors on a la

1. en fait, comme expliqué plus bas, 2.2.1 est ma version améliorée du théorème énoncé dans [50].

formule suivante pour le couplage de Tate de D_1 et D_2 :

$$(t(D_1,D_2))^l = \frac{\theta\begin{bmatrix}a\\b\end{bmatrix}(D_2+rD_1)}{\theta\begin{bmatrix}a\\b\end{bmatrix}(D_2)}\frac{\theta\begin{bmatrix}a\\b\end{bmatrix}(0)}{\theta\begin{bmatrix}a\\b\end{bmatrix}(D_1+D_2)}.$$

Ainsi, dans [30], la stratégie pour calculer le couplage de D_1 et D_2 avec la formule 2.1.1 est de chercher les coordonnées thêta du point D_2+rD_1.

Remarque 2.1.2. *Rappelons que les fonctions thêta ne sont pas Λ-périodiques. Ainsi, même si D_1 est dans la r-torsion de $\mathcal{J}$, nous n'avons pas forcément*

$$\theta\begin{bmatrix}a\\b\end{bmatrix}(D_2+rD_1) = \theta\begin{bmatrix}a\\b\end{bmatrix}(D_2)$$

Cela se fait par une échelle de Montgomery construite avec les formules d'addition satisfaites par les fonctions thêta : à chaque étape, connaissant les coordonnées de jD_1, $(j+1)D_1$ et jD_1+D_2, on va chercher les coordonnées de $2jD_1$, $(2j+1)D_1$ et $2jD_1+D_2$ ou celles des points $(2j+1)D_1$, $(2j+2)D_1$ et $(2j+1)D_1+D_2$, selon la décomposition binaire de r. En pratique, les fonctions thêta les plus efficaces sont celles de niveau 2 et 4, c'est-à-dire celles qui ont été exposées dans la seconde partie de la section 1.3.1.1.

2.2 Couplages et nets

Grâce au théorème 1.4.13, Uchida et Uchiyma ([50]) ont exprimé les couplages en fonction des hyperelliptic nets, généralisant par là le travail de Stange :

Théorème 2.2.1. *On reste dans la situation de la définition 1.2.3. Soit W l'hyperelliptic net associé à $D_1 \in \mathbb{G}_1$ et $D_2 \in \mathbb{G}_2$ défini dans le théorème 1.4.17 (c'est-à-dire issu de la fonction Ψ définie dans la proposition 1.4.11). Le couplage de Tate de D_1 et D_2 peut se calculer grâce à W :*

$$t(D_1,D_2) = \left(\frac{W(r+1,1)}{W(1,1)}\right)^{\frac{q^k-1}{r}}.$$

Ceci est une légère amélioration du théorème donné dans [50] : en effet, la formule initiale est

$$f_{r,D_1}(D_2) = \frac{W(r+1,1)W(1,0)}{W(r+1,0)W(1,1)}.$$

Pour simplifier cette formule, j'utilise le lemme suivant :

Lemme 2.2.2. *Dans les conditions du théorème, en particulier avec W l'hyperelliptic net associé à $D_1 \in \mathbb{G}_1$ et $D_2 \in \mathbb{G}_2$ défini dans le théorème 1.4.17, le fait que D_1 soit dans $\mathcal{J}(\mathbb{F}_q)$ nous assure que, pour tout $i \in \mathbb{N}$, $W(i,0) \in \mathbb{F}_q$.*

Démonstration. Grâce à la proposition 1.3.34, pour tout $a, b \in \mathbb{Z}$:

$$\frac{W(a+b,0)W(a-b,0)}{W(a,0)^2W(b,0)^2} = \mathcal{F}_g([a]D_1, [b]D_1),$$
$$\frac{W(2a,0)}{W(a,0)^4} = \mathcal{G}_g([a]D_1).$$

Or, comme

$$\mathcal{F}_g(D_1, D_2), \mathcal{G}_g(D_1) \in \mathbb{Z}[\wp_{ij}(D_1), \wp_{ijk}(D_1), \wp_{ij}(D_2), \wp_{ijk}(D_2)],$$

si $D_1 \in \mathcal{J}(\mathbb{F}_q)$, alors $\mathcal{F}_g([a]D_1, [b]D_1) \in \mathbb{F}_q$.
Enfin, dans le théorème 1.4.14 définissant les nets, je peux choisir $W(1,0) \in \mathbb{F}_q$ (en pratique, on prendra même $W(1,0) = 1$) pour s'assurer par récurrence qu'on ait bien tous les $W(i,0)$ dans $\mathbb{F}_q$.

□

Ainsi, avec ce lemme et l'exponentiation finale, j'obtiens la formule donnée dans le théorème 2.2.1.

Maintenant que le lien entre couplages et nets est établi, il faut expliquer comment construire les algorithmes exploitant ce théorème. Ils sont dérivés de la formule définissant les nets :

$$\det\left(W(v_i+v_j, w_i+w_j)W(v_i-v_j, w_i-w_j)\right)_{1\leq i,j\leq m} = 0. \tag{2.1}$$

Ce travail a été fait en genre 1 par Stange dans [45], et en genre 2 par Uchida et Uchiyama dans [50]. J'ai par la suite généralisé pour tout genre dans [48], ce que j'expose maintenant.

Soient D_1 et D_2 les deux diviseurs dont je veux calculer le couplage. L'hyperelliptic net associé à ces deux diviseurs est noté W. L'algorithme se déroule en deux phases :

Initialisation Dans cette phase, on utilise les polynômes $\mathcal{F}_g$ et $\mathcal{G}_g$ pour calculer un certain nombre (je préciserai plus loin) de valeurs $W(a,0)$ et $W(a,1)$, via :

$$\frac{W(a+b,0)W(a-b,0)}{W(a,0)^2W(b,0)^2} = \mathcal{F}_g([a]D_1, [b]D_1),$$
$$\frac{W(2a,0)}{W(a,0)^4} = \mathcal{G}_g([a]D_1).$$

Double – and – add Une fois cela fait, on utilise l'équation 2.1 pour obtenir les formules qui donneront au final la quantité $W(r,1)$ souhaitée.

Il faut signaler ici que ces formules utilisées dans la seconde phase diffèrent selon que $g \equiv 1,2 \pmod 4$ ou $g \equiv 0,3 \pmod 4$.
En effet, dans le premier cas, σ est impaire, donc les nets $\mathbf{v} \mapsto \Psi_{\mathbf{v}}$ construits dans 1.4.11 aussi. Ainsi, la matrice apparaissant dans l'équation 2.1 est antisymétrique. Cela implique deux choses : d'abord, si on choisit une taille m de matrice impaire, la seule équation que l'on obtient est $0 = 0$ (une matrice antisymétrique de taille impaire est toujours de déterminant nul). Ceci explique en particulier pourquoi dans le cas $g = 1$, Stange a travaillé avec $m = 4$. D'autre part, pour $m = 2n$, le déterminant d'une matrice antisymétrique A est le carré d'un polynôme de degré n, son pfaffien, noté $\mathrm{Pf}(A)$:

$$\mathrm{Pf}(A) = \frac{1}{2^n n!} \sum_{\sigma \in S_{2n}} \mathrm{sgn}(\sigma) \prod_{i=1}^{n} a_{\sigma(2i-1),\sigma_{2i}},$$

où S_{2n} est le groupe symétrique et $\mathrm{sgn}(\sigma)$ est la signature de σ.
Dans le cas $g \equiv 0,3 \pmod 4$ au contraire σ est paire, la matrice de 2.1 est symétrique, et il faudra trouver d'autres formules.

2.2.1 Le cas $g \equiv 1,2 \pmod 4$

Avant de donner les théorèmes formels, j'explique le type de formules qu'on utilisera. Notons A la matrice apparaissant dans l'équation 2.1 :

$$A = (W(v_i + v_j, w_i + w_j)W(v_i - v_j, w_i - w_j))_{1 \leq i,j \leq m}.$$

Comme signalé plus haut, la taille m de la matrice doit vérifier deux propriétés : m est paire et $m > 2^g$. Evidemment, pour avoir les formules les plus simples possibles, on prend la valeur $m = 2^g + 2$ minimale.

Choisissons les familles d'indices (v_i) et (w_i) de façon à rendre l'équation

$$\mathrm{Pf}(A) = 0$$

intéressante. Par exemple, pour a un entier, on choisit :

- pour tout i, $w_i = 0$;
- $v_1 = a + 1$, $v_2 = a - 1$;
- pour tous les autres $i \notin \{1,2\}$, $v_i = m - i$.

Je l'ai déjà signalé plus haut : A est antisymétrique. En particulier, ses termes diagonaux sont nuls, et ses autres coefficients valent

- $A_{12} = W(2a, 0)W(2, 0)$, et pour $i \geq 3$,

$$A_{1i} = W(a + 1 + m - i, 0)W(a + 1 - m + i, 0);$$

- $A_{21} = -A_{12}$, et pour $i \geq 3$,

$$A_{2i} = W(a + m - i - 1, 0)W(a - m + i - 1, 0);$$

- enfin, pour i et $j \in \{3, \ldots, m\}$, les autres A_{ij} valent

$$A_{ij} = W(2m - i - j, 0)W(j - i, 0).$$

Autrement dit, l'équation $\mathrm{Pf}(A) = 0$ est une relation polynomiale permettant de calculer $W(2a, 0)$ à partir
- des $W(a + i, 0)$, pour $-(m - 2) \leq i \leq m - 2$;
- des $W(j, 0)$, pour $1 \leq j \leq m - 3$ [2].

On voit donc se dégager l'idée de l'algorithme : c'est bien un double-and-add, mais il faut faire attention au fait que $W(2a, 0)$ ne s'obtient pas à partir de seulement $W(a, 0)$ et de quelques valeurs initiales. On a également besoin de termes autour de $W(a, 0)$.

Par ailleurs, je rappelle que pour le calcul de couplage je dois calculer le terme $W(r + 1, 1)$. Autrement dit, il faut que je donne les formules permettant de calculer des termes du type $W(2a, 1)$ ou $W(2a + 1, 1)$. Cela se fait naturellement avec d'autres choix d'indices v_i et w_i que j'expliciterai plus loin. Pour le moment, il me suffira de dire que le même phénomène observé intervient : on n'aura pas seulement besoin de $W(a, 0)$ ou $W(a, 1)$, mais également de tout un tas d'autres valeurs de W autour de $(a, 0)$.

Muni de cette observation, définissons comme l'ont fait Stange, puis Uchida et Uchiyama respectivement pour les genres 1 et 2 la notion de bloc centré autour d'un entier a :

Définition 2.2.3. Les termes dits *initiaux* du net W sont :
— les $2m - 7$ termes $\{W(i, 0) \mid 1 \leq i \leq 2m - 7\}$ (ou dans le cas $g = 1$ et $m = 4$, les deux termes $W(1, 0)$ and $W(2, 0)$),
— les $2m - 2$ termes $\{W(i, 1) \mid 1 - m \leq i \leq m - 2\}$.

Soit $a \in \mathbb{N}$, le *bloc centré autour de* a est formé des
— $4m - 8$ termes $\{W(k + i, 0) \mid -2m + 4 \leq i \leq 2m - 5\}$,

2. Je rappelle que W est impaire : $W(-j, 0) = -W(j, 0)$.

— $2m-5$ termes $\{W(k+i,1) \mid 3-m \leq i \leq m-3\}$.

Remarque 2.2.4.

- *Dans mon exemple, j'avais décomposé* $2a$ *en* $(a+1)+(a-1)$*. On pourrait se demander s'il n'aurait pas été plus simple de tout simplement écrire* $2a = a + a$*, autrement dit de faire le choix* $(v_1, w_1) = (v_2, w_2) = (a, 0)$*. En réalité, un tel choix est impossible, et ce pour deux raisons : tout d'abord, parce qu'en faisant ainsi, on aurait eu les deux premières lignes de ma matrice* A *égales, et donc en calculant son pfaffien, on n'aurait pas obtenu d'autre équation que* $0 = 0$*; et de toute manière, on aurait eu* $A_{12} = W(2a,0)W(0,0)$*, et je rappelle que* $W(0,0) = 0$ *: ainsi, le terme* $W(2a,0)$ *ne serait en réalité pas apparu dans l'équation.*
- *Toujours dans l'exemple, on peut observer que j'ai arbitrairement placé le terme intéressant à calculer dans la case* A_{12} *de ma matrice ; que les autres coefficients des deux premières lignes sont formés de termes qui sont dans le bloc calculé à l'étape précédente du double-and-add ; et que les autres coefficients de la matrice sont des termes initiaux.*
- *Ces termes initiaux sont apparus du fait de mon choix*

$$\forall 3 \leq i \leq m,\ v_i = m - i.$$

De façon triviale, on peut dire que les deux indices vraiment intéressants sont v_1 *et* v_2*, puisque ce sont avec eux que j'obtiens le terme à calculer* $W(2a,0)$*. Les autres* v_i *ne sont en somme que du remplissage, et je rappelle qu'il ne m'est pas possible de prendre pour deux indices* i *et* j *différents* $(v_i, w_i) = (v_j, w_j)$*, ce qui explique mon choix pour ces valeurs de* v_i *: j'ai pris les* $m-2$ *plus petits entiers.*
- *Une remarque sur la définition maintenant. On remarquera qu'un bloc est constitué de deux niveaux : le niveau* 0*, constitué des termes du type* $W(i,0)$*, et le niveau* 1 *avec tous les termes du type* $W(i,1)$*. Les termes* $W(i,0)$ *ne dépendent que de* D_1 *et des coefficients de la courbe* $\mathcal{C}$*, et vivent donc dans le corps de base* $\mathbb{F}_q$*, tandis que les* $W(i,1)$ *dépendent de* D_1 *et de* D_2*, et seront donc des éléments de* $\mathbb{F}_{q^k}$*. Ainsi, le calcul d'un bloc se déroulera en deux temps : d'abord on calculera les termes de niveau* 0*, et ces opérations seront peu couteuses puisque dans* $\mathbb{F}_q$ *; puis on réglera le compte du niveau* 1*, et pour cela il faudra payer un peu plus, puisqu'on évoluera dans* $\mathbb{F}_{q^k}$*.*
- *Enfin, on remarquera qu'avec* $g = 1$ *et* 2*, ma définition de bloc coïncide bien avec celles de [45] et [50].*

Définition 2.2.5 (*Paramétrages*)**.** Voici les différents choix des paramètres v_i et w_i qu'on va utiliser dans notre algorithme.

Pour le niveau 0, c'est-à-dire pour le calcul des termes $W(2a+c,0)$, $4-2m \leq c \leq 2m-4$, on utilise le choix suivant :

— pour tout $1 \leq i \leq m$, $w_i = 0$,
— $v_1 = l + l' + 1$,
— $v_2 = l - 1$,
— pour tout $3 \leq i \leq m$, $v_i = m - i$,

où

$$l = a + \left\lceil \frac{c}{2} \right\rceil \text{ et } l' = \begin{cases} 0 \text{ si } c \text{ est pair.} \\ -1 \text{ sinon.} \end{cases}$$

Un tel choix sera désormais appelé **paramétrage** $(\mathbf{l}, \mathbf{l'}, \mathbf{0})$, et vérifie

$$a - (m-2) \leq l \leq a + (m-2).$$

Pour le niveau 1 maintenant, on calcule $W(2a+c,1)$ $(-(m-3) \leq c \leq m-2)$ avec le choix suivant :

— $w_1 = 1$ et tous les autres w_i valent 0 ;
— $v_1 = a$;
— $v_2 = a + c$;
— for $3 \leq i \leq m$, $v_i = m - i$.

Un tel choix est appelé **paramétrage** $(\mathbf{a}, \mathbf{c}, \mathbf{1})$.

Regardons maintenant précisément les formules que l'on obtient avec ces différents paramétrages.
Je note $\hat{A}_{ij}$ la matrice carrée de taille $m-2$ obtenue en retirant les lignes et colonnes numéro i et j de A. De même, $\hat{A}_{ijkl}$ est la matrice de taille $m-4$ obtenue en retirant les 4 lignes et 4 colonnes numéro i, j, k et l de A. Le pfaffien de A se développe alors de la façon suivante :

$$\begin{aligned} \mathrm{Pf}(A) &= A_{12}\mathrm{Pf}(\hat{A}_{12}) + \sum_{i=3}^{m} (-1)^i A_{1i}\mathrm{Pf}(\hat{A}_{1i}) \\ &= A_{12}\mathrm{Pf}(\hat{A}_{12}) + \sum_{i=3}^{m} \sum_{\substack{j \neq i \\ 3 \leq j \leq m}} (-1)^{i+j} A_{1i}A_{2j}\mathrm{Pf}(\hat{A}_{12ij}). \end{aligned}$$

Or, on remarquera que dans tous les paramétrages $(l, l', 0)$ et $(a, c, 1)$, les lignes 3 à m de A sont identiques (seules les 2 premières varient). Ainsi, les coefficients $\mathrm{Pf}(\hat{A}_{12})$ et $\mathrm{Pf}(\hat{A}_{12ij})$ de nos équations sont constants tout le long de l'algorithme : il faudra les précalculer.

Finalement, les formules obtenues sont les suivantes :

Définition 2.2.6 (*Formules*)**.**

— Pour $-m+2 \leq b \leq m-2$,

$$W(2a+2b,0) = \sum_{\substack{j\neq i \\ 3\leq i,j\leq m}} \frac{(-1)^{i+j+1}\mathrm{Pf}\left(\hat{A}_{12ij}\right)}{\mathrm{Pf}\left(\hat{A}_{12}\right)W(2,0)} A_{1i}A_{2j}$$

avec $A_{1i} = W(a+b+m+1-i,0)W(a+b+1+i-m,0)$ et
$A_{2j} = W(a+b+m-1-j,0)W(a+b-1+j-m,0)$;
— pour $-m+3 \leq b \leq m-2$,

$$W(2a+2b+1,0) = \sum_{\substack{j\neq i \\ 3\leq i,j\leq m}} \frac{(-1)^{i+j+1}\mathrm{Pf}\left(\hat{A}_{12ij}\right)}{\mathrm{Pf}\left(\hat{A}_{12}\right)W(1,0)} A_{1i}A_{2j}$$

avec $A_{1i} = W(a+b+m+1-i,0)W(a+b+1+i-m,0)$ et
$A_{2j} = W(a+b+m-j,0)W(a+b+j-m,0)$;
— pour $-m+3 \leq b \leq m-2$,

$$W(2a+b,1) = \sum_{\substack{j\neq i \\ 3\leq i,j\leq m}} \frac{(-1)^{i+j+1}\mathrm{Pf}\left(\hat{A}_{12ij}\right)}{\mathrm{Pf}\left(\hat{A}_{12}\right)W(-b,1)} A_{1i}A_{2j}$$

avec $A_{1i} = W(a+m-i,1)W(a+i-m,1)$ et
$A_{2j} = W(a+b+m-j,0)W(a+b+j-m,0)$.

A partir de ces formules, on obtient notre algorithme de type double-and-add, ce qu'énonce le théorème suivant :

Théorème 2.2.7. *Soit a un entier. On suppose qu'aucun des termes suivants n'est nul :*
— *$W(1,0)$ et $W(2,0)$;*
— *$W(b,1)$, pour $-m+2 \leq b \leq m-3$;*
— *$Pf\left(\hat{A}_{12}\right)$.*

Dans ces conditions, si on dispose des termes initiaux de W et du bloc centré en a, alors on est en mesure de calculer le bloc centré en $2a$ et celui centré en $2a+1$.

Démonstration. Les conditions du théorème permettent de s'assurer que les dénominateurs des formules énoncées ne s'annulent pas. Une fois ces précautions prises, il ne reste alors plus qu'à vérifier que les termes apparaissant dans les membres de droite des différentes formules sont bien tous ou bien initiaux, ou bien dans le bloc centré en a.

□

Remarque 2.2.8.

— *Comme signalé dans la preuve du lemme 2.2.2, en pratique on aura* $W(1,0) = W(0,1) = 1$.
— *Pour* $g = 1$ *et* $m = 4$, *on a tous les* $Pf\left(\hat{A}_{12ij}\right)$ *égaux à* 1, *ce qui nous redonne les équations de [45].*

Enfin, remarquons que parmi tous ces coefficients A_{1j} et A_{2j} dont on a besoin, certains sont redondants :
— les A_{1j} du paramétrage $(l+1,-1,0)$ sont les mêmes que ceux du paramétrage $(l,0,0)$,
— les A_{2j} du paramétrage $(l,-1,0)$ sont les mêmes que ceux du paramétrage $(l,0,0)$,
— les A_{2j} du paramétrage $(l+2,-1,0)$ sont les A_{1j} du paramétrage $(l,0,0)$,
— les A_{1j} du paramétrage $(k,c,1)$ sont indépendants de c,
— les A_{2j} du paramétrage $(a,c,1)$ sont les mêmes que ceux du paramétrage $(a+c+1,0,0)$.

Ainsi, les seuls coefficients à calculer sont :
— les A_{1j}, $3 \leq j \leq m$, pour les $2m-4$ paramétrages $(l,0,0)$;
— les A_{2j}, $3 \leq j \leq m$, pour les paramétrages $(a-m+2,0,0)$ et $(a-m+3,0,0)$, ou $(a-m+3,0,0)$ et $(a-m+4,0,0)$;
— les A_{1j}, $3 \leq j \leq m$, du paramétrage $(a,-m+3,1)$.

2.2.2 Le cas $g \equiv 0,3 \pmod 4$

La stratégie est la même que précédemment : étant donné un bloc centré en a, on désire calculer le bloc centré en $2a$ ou celui centré en $2a+1$, de façon à finalement sortir $W(r+1,1)$.
Les différences avec le cas $g \equiv 1,2 \pmod 4$ sont les suivantes :
— la matrice A est maintenant symétrique ;
— on n'a plus besoin de prendre m pair, donc je peux prendre $m = 2^g + 1$;
— malheureusement, on ne dispose plus du pfaffien, mais seulement du déterminant. Ceci va alourdir les équations qu'on obtiendra.

La nouvelle définition de termes initiaux et de bloc est la suivante :

Définition 2.2.9. Les termes initiaux sont les
— $2m-3$ termes $\{W(i,0) \mid 0 \leq i \leq 2m-4\}$,
— $2m-2$ termes $\{W(i,1) \mid 0 \leq i \leq 2m-3\}$.

Pour a un entier, le bloc centré en a est constitué des
— $4m-6$ valeurs $\{W(a+i,0) \mid -2m+4 \leq i \leq 2m-3\}$,

— $2m-2$ valeurs $\{W(a+i,1) \mid 0 \leq i \leq 2m-3\}$.

On donne maintenant la définition des paramétrages choisis :

Définition 2.2.10 (*Paramétrages*)**.**

Pour calculer $W(2a+2b,\epsilon)$, on prend
— pour tout $1 \leq i \leq m$, $w_i = \epsilon/2$,
— $v_1 = a+b$,
— pour $2 \leq i \leq m$, $v_i = m-i$.
Ce réglage sera noté $(\mathbf{b}, \mathbf{0}, \boldsymbol{\epsilon})$, pour $\epsilon = 0, 1$, et $-m+2 \leq b \leq m-1$ si $\epsilon = 0$, ou $0 \leq b \leq m-1$ si $\epsilon = 1$.

Pour calculer $W(2a+2b+1,\epsilon)$, où $\epsilon = 0, 1$, et $-m+2 \leq b \leq m-2$ si $\epsilon = 0$, ou $0 \leq b \leq m-2$ si $\epsilon = 1$, on prend le réglage $(\mathbf{b}, \mathbf{1}, \boldsymbol{\epsilon})$, à savoir :
— pour tout $1 \leq i \leq m$, $w_i = \epsilon/2$,
— $v_1 = a+b+1/2$,
— pour $2 \leq i \leq m$, $v_i = m-i+1/2$.

Comme pour $g \equiv 1, 2 \pmod 4$, il faut maintenant donner les formules précises qui découlent de ces réglages.
Je note $\hat{A}_{i,j}$ la matrice carrée de taille $m-1$ obtenue en ôtant la i-ième ligne et la j-ième colonne de A. De même, j'obtiens $\hat{A}_{ij,kl}$ en retirant les lignes numéro i et j et les colonnes k et l de A.
Je développe alors le déterminant de A de la façon suivante :

$$0 = A_{11}\det\left(\hat{A}_{1,1}\right) + \sum_{1 \leq i,j \leq m} (-1)^{i+j} A_{1i} A_{1j} \det\left(\hat{A}_{1i,1j}\right).$$

Les coefficients $\det\left(\hat{A}_{1,1}\right)$ et $\det\left(\hat{A}_{1i,1j}\right)$ sont précalculés.
Malheureusement, et contrairement au cas $g \equiv 1, 2 \pmod 4$, ces déterminants diffèrent selon le réglage utilisé. Je les noterai donc respectivement par les lettres B, C, D et E pour les réglages $(b,0,0)$, $(b,1,0)$, $(b,0,1)$ and $(b,1,1)$.
Les formules obtenues sont alors les suivantes :

Définition 2.2.11 (*Formules*)**.**

— Pour $-m+2 \leq b \leq m-1$,

$$W(2a+2b,0) = \sum_{2 \leq i,j \leq m} \frac{(-1)^{i+j+1}\det\left(\hat{B}_{1j,1i}\right)}{\det\left(\hat{B}_{1,1}\right) W(0,0)} A_{1i} A_{1j}$$

avec $A_{1i} = W(a+b+m-i,0)W(a+b+i-m,0)$;

— pour $-m+2 \leq b \leq m-2$,

$$W(2a+2b+1,0) = \sum_{2\leq i,j\leq m} \frac{(-1)^{i+j+1}\det\left(\hat{C}_{1j,1i}\right)}{\det\left(\hat{C}_{1,1}\right)W(0,0)} A_{1i}A_{1j}$$

avec $A_{1i} = W(a+b+m+1-i,0)W(a+b+i-m,0)$;
— pour $0 \leq l \leq m-1$,

$$W(2k+2l,1) = \sum_{2\leq i,j\leq m} \frac{(-1)^{i+j+1}\det\left(\hat{D}_{1j,1i}\right)}{\det\left(\hat{D}_{1,1}\right)W(0,0)} A_{1i}A_{1j}$$

avec $A_{1i} = W(a+b+m-i,1)W(a+b+i-m,0)$;
— pour $0 \leq l \leq m-2$,

$$W(2k+2l+1,1) = \sum_{2\leq i,j\leq m} \frac{(-1)^{i+j+1}\det\left(\hat{E}_{1j,1i}\right)}{\det\left(\hat{E}_{1,1}\right)W(0,0)} A_{1i}A_{1j}$$

avec $A_{1i} = W(a+b+m+1-i,1)W(a+b+i-m,0)$.

Alors, avec le même raisonnement que dans le cas $g \equiv 1,2 \pmod 4$, on obtient à partir de ces formules notre algorithme en double-and-add :

Théorème 2.2.12. *Si aucune des valeurs* $det\left(\hat{B}_{1,1}\right)$, $det\left(\hat{C}_{1,1}\right)$, $det\left(\hat{D}_{1,1}\right)$ *et* $det\left(\hat{E}_{1,1}\right)$ *ne s'annule, alors, à partir des termes initiaux et du bloc centré en* a*, on est en mesure de calculer le bloc centré en* $2a$ *et celui centré en* $2a+1$*.*

Hélas, une autre différence pénible avec le cas $g \equiv 1,2 \pmod 4$ apparaît : il n'y a aucune redondance parmi les coefficients A_{1i} et A_{2j}, et il faudra tous les calculer.

2.2.3 Un exemple en genre 3

En genre 3, notre courbe a pour équation

$$\mathcal{C} \ : \ y^2 = x^7 + \lambda_6 x^6 + \cdots + \lambda_0.$$

La représentation de Mumford d'un diviseur $D \in \mathcal{J}\backslash\Theta$ est

$$D = [U,V] \text{ avec } U = x^3 + U_2x^2 + U_1x + U_0,\ V = V_2x^2 + V_1x + V_0.$$

On se donne une courbe $\mathcal{C}$ et deux diviseurs D_1 et D_2 dont on veut calculer le couplage. Comme expliqué plus haut, on commence par la phase d'initialisation,

c'est-à-dire par le calcul des $\{W(i,0) \mid i = 0, \ldots, 14\}$ et $\{W(i,1) \mid i = 0, \ldots, 15\}$.

Pour cela, on prend $W(1,0) = W(0,1) = 1$, et on obtient les autres avec les polynômes $\mathcal{F}_3$ et $\mathcal{G}_3$ (voir proposition 1.3.34) :

$$\forall a, b, i \in \mathbb{Z}, \begin{cases} \dfrac{W(a+b,i)W(a-b,i)}{W(a,i)^2 W(b,0)^2} = \mathcal{F}_3([a]D_1 + [i]D_2, [b]D_1), \\ \dfrac{W(2a,0)}{W(a,0)^4} = \mathcal{G}_3([a]D_1). \end{cases}$$

Les valeurs de $\mathcal{F}_3$ et $\mathcal{G}_3$ se trouvent par exemple dans [10] et [49] :

$$\begin{aligned} \mathcal{F}_3(u,v) &= (\wp_{31}(v) - \wp_{31}(u))(\wp_{22}(v) - \wp_{22}(u)) - (\wp_{31}(v) - \wp_{31}(u))^2 \\ &\quad + (\wp_{32}(v) - \wp_{32}(u))(\wp_{21}(v) - \wp_{21}(u)) + (\wp_{33}(v) - \wp_{33}(u))(\wp_{11}(v) - \wp_{11}(u)), \end{aligned}$$

$$\begin{aligned} \mathcal{G}_3(u) &= \wp_{113}(u)\wp_{223}(u) + \wp_{133}(u)\wp_{122}(u) - 2\wp_{133}(u)\wp_{113}(u) \\ &\quad - \wp_{123}(u)^2 - \wp_{233}(u)\wp_{112}(u) + \wp_{133}(u)\wp_{113}(u) + \wp_{333}(u)\wp_{111}(u). \end{aligned}$$

Je rappelle que les $\wp_{i3}$ et $\wp_{i33}$, $1 \leq i \leq 3$ d'un diviseur D se lisent directement sur ses coordonnées de Mumford. Les autres évaluations des $\wp$ fonctions s'obtiennent à partir des formules se trouvant dans l'appendice C de [13]. Plus précisément, on utilise les trois premières formules pour calculer successivement $\wp_{22}$, $\wp_{12}$ et $\wp_{11}$. Puis on utilise les formules de poids (-18) de ce même document pour obtenir les sept produits $\wp_{ijk}\wp_{lmn}$ intervenant dans le calcul de $\mathcal{G}_3$.

Par exemple, considérons la courbe $\mathcal{C}$ d'équation

$$y^2 = x^7 + 3x^6 + 2x^5 + 10x^4 + 9x^3 + 3x^2 + 11$$

définie sur $\mathbb{F}_{29}$. Avec $r = 41$, le degré de plongement est 40, donc l'exponentiation finale est

$$e = \frac{29^{40} - 1}{41} = 764075121631975351615803381072559018207546756758889888800.$$

Soit $a \in \overline{\mathbb{F}}_{29}$ une racine de

$$X^{40} + X^5 + 4 \in \mathbb{F}_{29}[X],$$

alors $\mathbb{F}_{29^{40}} \simeq \mathbb{F}_{29}(a)$.

Soit $D_1 \in \mathcal{J}(\mathbb{F}_{29})[41]$ donné par sa représentation de Mumford :

$$[x^3 + 2x^2 + 9x + 24, 23x^2 + 24x + 4].$$

Soit $D_2 \in \mathcal{J}(\mathbb{F}_{29^{40}})$ donné par :

$$\begin{aligned}U =&x^3 + (27a^5 + 27a^2 + 28a + 16)x^2 + (4a^7 + 2a^6 + 21a^5 + 2a^3 + 21a^2 + 5a + 10)x + \\ &25a^8 + 26a^7 + 24a^6 + 18a^5 + 24a^3 + 18a^2 + a + 8\end{aligned}$$

$$\begin{aligned}V =&(26a^{39} + 16a^{38} + 14a^{37} + 7a^{36} + 27a^{35} + 19a^{34} + 7a^{33} + 19a^{32} + 15a^{31} + a^{30} + \\ &21a^{29} + 2a^{28} + 5a^{27} + 22a^{26} + 27a^{25} + 21a^{24} + 4a^{23} + 5a^{22} + 6a^{21} + 27a^{20} + 6a^{19} + \\ &27a^{18} + 13a^{17} + 12a^{16} + 15a^{15} + 10a^{14} + 23a^{13} + 23a^{12} + 25a^{11} + 2a^{10} + 4a^9 + 14a^8 + \\ &a^{26} + 21a^{25} + 26a^{24} + 28a^{23} + 10a^{22} + 14a^{21} + 3a^{20} + 23a^{19} + 14a^{18} + 26a^{17} + 7a^{16} + \\ &23a^7 + 3a^6 + 28a^5 + 2a^4 + 26a^3 + 12a^2 + 27a + 4)x^2 + (a^{39} + 15a^{38} + 6a^{37} + 13a^{36} + \\ &20a^{35} + 5a^{34} + 23a^{33} + 28a^{32} + 8a^{31} + 20a^{30} + 16a^{29} + a^{28} + 12a^{27} + 13a^{15} + 28a^{14} + \\ &15a^{13} + 25a^{12} + a^{11} + 19a^{10} + 11a^9 + 17a^8 + 21a^7 + 11a^6 + 12a^5 + 16a^4 + 8a^2 + 21a + \\ &22)x + 15a^{39} + 14a^{38} + 23a^{37} + 25a^{36} + 27a^{35} + 14a^{34} + 13a^{33} + 10a^{31} + 3a^{30} + \\ &14a^{29} + 13a^{28} + 27a^{27} + 14a^{26} + 21a^{25} + 13a^{24} + 8a^{23} + 25a^{22} + 27a^{21} + 23a^{19} + \\ &24a^{18} + 11a^{17} + 6a^{16} + a^{15} + 3a^{14} + 18a^{13} + a^{12} + 21a^{11} + 2a^{10} + 26a^9 + \\ &26a^8 + 11a^7 + 4a^6 + 18a^5 + 9a^4 + 28a^3 + 5a^2 + 4a + 6.\end{aligned}$$

On trouve alors que

$$(W_{D1,D2}(42,1))^e = 4a^{39} + 14a^{38} + 21a^{37} + \cdots + 5a^2 + 4a + 16.$$

On peut vérifier que

$$\begin{array}{rccc}\tau : & \mathcal{J}(\mathbb{F}_q)[r] \times \mathcal{J}(\mathbb{F}_{q^k})/r\mathcal{J}(\mathbb{F}_{q^k}) & \to & \mu_r \\ & (D_1, D_2) & \mapsto & (W_{D_1,D_2}(r+1,1))^e\end{array}$$

est bien bilinéaire :

— on prend $D_3 \in_R \mathcal{J}(\mathbb{F}_{q^k})$ et vérifie que

$$(W_{D_1,D_2+rD_3}(r+1,1))^e = (W_{D_1,D_2}(r+1,1))^e ;$$

— on prend deux entiers aléatoires m and n et vérifie que

$$(W_{mD_1,nD_2}(r+1,1))^e = (W_{D_1,D_2}(r+1,1))^{emn} .$$

2.2.4 Analyse théorique de coûts

Comptons le nombre d'opérations effectuées à chaque itération de notre double-and-add.
Il faudra distinguer les opérations faisant intervenir uniquement les coordonnées de $P \in \mathcal{J}(\mathbb{F}_q)$ et ceux faisant également intervenir $Q \in \mathcal{J}(\mathbb{F}_{q^k})$. Ainsi, je note $\mathbf{M}$ et $\mathbf{S}$ les coûts respectifs d'une multiplication et d'une mise au carré dans $\mathbb{F}_{q^k}$, M

et S les coûts de ces opérations dans $\mathbb{F}_q$, et M_k le coût d'une multiplication mixte, c'est-à-dire entre un terme dans $\mathbb{F}_{q^k}$ et un autre dans $\mathbb{F}_q$. Le coût des additions sera négligé.

Je rappelle qu'à chaque itération il faudra calculer tous les termes d'un bloc. Ce bloc est constitué de termes de la forme $W(a,0) \in \mathbb{F}_q$ et de termes de la forme $W(a,1) \in \mathbb{F}_{q^k}$. Ces termes se calculent avec les formules données dans les sous sections 2.2.1 et 2.2.1.
Ces formules sont toutes de la forme

$$W(a,b) = \sum K_{ij} A_{1i} A_{2j}$$

avec

- $W(a,b)$ le terme à calculer ;
- les coefficients $K_{ij} \in \mathbb{F}_q$ si $b = 0$ et $\mathbb{F}_{q^k}$ si $b = 1$;
- dans le cas où $g \equiv 1, 2 \pmod 4$, ces K_{ij} sont précalculés avant de rentrer dans la boucle du double-and-add ; dans l'autre cas, il faudra les recalculer à chaque fois ;
- enfin, les éléments A_{1i} et A_{2j} de la matrice sont dans $\mathbb{F}_q$ si $b = 0$; certains d'entre eux sont dans $\mathbb{F}_{q^k}$ sinon.

L'analyse tiendra ainsi compte du coût de calcul des termes K_{ij}, A_{1i} et A_{2j} des équations, puis du coût des opérations à effectuer entre ces termes. Je rappelle que dans le cas où $g \equiv 1, 2 \pmod 4$ il y aura moins de termes A_{1i} et A_{2j} à calculer que dans le cas où $g \equiv 0, 3 \pmod 4$.

On obtient finalement le nombre suivants d'opérations :

— $8 \cdot 2^{3g} - 9 \cdot 2^{2g} + 5 \cdot 2^g - 2$ M,
— $2^{g+1} + 2$ S,
— $2^g(2^g - 1)(2^{g+1} - 1) + 2$ M_k,
— $2^{g+1} + 2^g - 4$ $\mathbf{M}$,
— 1 $\mathbf{S}$

si $g \equiv 1, 2 \pmod 4$; et

— $(4m^3 - 8m^2 + 4m - 4)$ M,
— $2(m-1)^2$ M_k,
— $(2m^3 - 6m^2 + 8m - 4)$ $\mathbf{M}$,
— $2(m-1)^2$ $\mathbf{S}$

si $g \equiv 0, 3 \pmod 4$.

2.3 Résultats en genre 1

Ici, je compare les coûts des algorithmes basés respectivement sur les fonctions thêta ([30]) et sur les nets ([45]) pour le genre 1.

Dans le premier cas, je rappelle que les bases de fonctions de niveau 2 introduites dans la partie 1.3.1.1 sont les suivantes :

— $[u_P : v_P] = \left[\theta \begin{bmatrix} 0 \\ 0 \end{bmatrix}(P,\Omega)^2 : \theta \begin{bmatrix} 1/2 \\ 0 \end{bmatrix}(P,\Omega)^2\right]$,

— $[u'_P : v'_P] = \left[\theta \begin{bmatrix} 0 \\ 0 \end{bmatrix}(2P,2\Omega) : \theta \begin{bmatrix} 1/2 \\ 0 \end{bmatrix}(2P,2\Omega)\right]$,

— $[\tilde{u}_P : \tilde{v}_P] = \left[\theta \begin{bmatrix} 0 \\ 0 \end{bmatrix}(P,\Omega/2) : \theta \begin{bmatrix} 0 \\ 1/2 \end{bmatrix}(P,\Omega/2)\right]$.

La stratégie a été rappelée dans la section 2.1 : c'est un double-and-add, et, à chaque itération, à partir des coordonnées de jD_1, $(j+1)D_1$ et $jD_1 + D_2$, on va chercher les coordonnées de $2jD_1$, $(2j+1)D_1$ et $2jD_1 + D_2$ ou celles des points $(2j+1)D_1$, $(2j+2)D_1$ et $(2j+1)D_1 + D_2$, selon la décomposition binaire de r. Ainsi, à chaque itération, on utilise toujours les formules données dans la proposition 1.3.18 pour effectuer deux additions et un doublement. En étudiant ces formules, on remarque que les coûts de l'arithmétique des coordonnées (u', v') sont les mêmes que ceux pour les coordonnées $(\tilde{u}, \tilde{v})$.

Avec les notations de la section précédente, nous avons donc les coûts suivants pour une itération :

Coordonnées	(u, v)	(u', v') ou $(\tilde{u}, \tilde{v})$
Coûts	$2\mathbf{S} + 1\mathbf{M} + 3M_k + 6S + 6M$	$2\mathbf{S} + 1\mathbf{M} + 2M_k + 10S + 7M$

Du côté elliptic net, dans [45], le coût d'une étape est de $1\mathbf{S} + 2\mathbf{M} + 6M_k + 6S + 26M$.

Pour comparer ces coûts, plusieurs approximations sont faites. Ainsi, en pratique, on considère généralement que $1M_k \simeq kM$.
De plus, en cryptographie, le corps $\mathbb{F}_{q^k}$ est construit à partir de $\mathbb{F}_q$ comme une tour d'extensions. On prendra volontiers k comme un entier de la forme $2^i 3^j$, et ainsi $\mathbb{F}_{q^k}$ s'obtient grâce à une succession d'extensions quadratiques et cubiques. Grâce aux méthodes de Karatsuba et Toom-Cook, on a donc $\mathbf{M} \simeq 3^i 6^j M$ et $\mathbf{S} \simeq 2^i M 5^j S$ (voir par exemple [12] pour plus de détails). Si finalement on admet l'approximation $M \simeq S$, on obtient le tableau suivant pour les premières valeurs de k :

k	Coordonnées (u, v)	Coordonnées (u', v') ou $(\tilde{u}, \tilde{v})$	Méthode Elliptic Nets
4	41M	42M	78M
6	65M	64M	108M
8	79M	76M	142M
12	143M	136M	214M
16	173M	162M	306M
18	274M	261M	406M
24	299M	280M	486M
36	644M	613M	996M
48	721M	678M	1210M

Si on compare le coût d'utilisation des coordonnées (u, v) avec celui pour les deux autres jeux de coordonnées, on remarque qu'un petit nombre d'opérations dans le corps de base ($4S$ et $1M$) est échangé contre une multiplication mixte. Cela explique que pour les grandes valeurs de k, les coordonnées (u', v') et $(\tilde{u}, \tilde{v})$ sont plus efficaces que les coordonnées (u, v). Pour les petites valeurs de k par contre la différence est négligeable.
Concernant l'algorithme des elliptic nets, le nombre prohibitif de multiplications mixtes le rend non compétitif par rapport à l'algorithme basé sur les formules d'addition des fonctions thêta.

Enfin, comparons les performances de ces algorithmes avec celles des méthodes traditionnelles de calcul de couplage, basées sur l'algorithme de Miller. Selon [28], le coût d'une étape est de $1\mathbf{S} + 1\mathbf{M} + 1M_k + 11M$. Ainsi que noté dans [31], ces résultats rendent les algorithmes étudiés dans ce chapitre très peu compétitifs en genre 1. Ainsi, le coût de l'utilisation de l'algorithme de type Miller donné dans [28] varie selon les valeurs de k entre 68 et 86% celui de l'utilisation des coordonnées (u', v').

Chapitre 3

Unification de ces deux méthodes

Observant les théorèmes 2.1.1 et 2.2.1, on remarque une grande similitude entre la formule de couplages fournie par les nets, et celle donnée par les fonctions thêta. L'objet de ce chapitre est d'expliquer cette ressemblance.
Plus exactement, je montre ici que les deux théorèmes 2.1.1 et 2.2.1 sont en fait des "sous cas" des propriétés suivantes des fonctions de niveau l (i.e. des bases des espaces de Riemann $\mathcal{L}(l\mathcal{O})$) :

(**i**) si l est pair, alors tout ensemble de coordonnées de niveau l vérifie une loi d'addition, à partir de laquelle on peut en extraire un algorithme tel que celui de [30] ;

(**ii**) si l est pair, cette loi d'addition a pour conséquence que ces coordonnées forment ce que j'appelle des generalized hyperelliptic nets ; en particulier, si $l = 2$, ce sont des hyperelliptic nets au sens classique de [50] (cette définition est rappelée dans le théorème 1.4.14).

Dans ce chapitre, et comme je l'ai déjà fait plusieurs fois, je m'intéresserai d'abord au cas elliptique. Il sera ensuite aisé de généraliser le résultat aux genres supérieurs.

3.1 Réécriture des elliptic nets

Je rappelle qu'un des théorèmes fondamentaux de Stange dans [45] est le théorème 1.4.7, établissant le diviseur de sa fonction Ψ, à partir de laquelle sont bâtis ses elliptic nets. Ainsi, ce théorème lui-a-t'il permis de reconstruire les fonctions de Miller, puisque ceux-ci sont eux mêmes définis par leur diviseur.

Son théorème 1.4.7, elle l'a établi connaissant le diviseur de sigma : cette fonction admet un zéro simple en $\mathcal{O}$, et rien d'autre. Il est aisé de faire le même travail en utilisant une autre fonction thêta. Ainsi, dans le théorème suivant, nous allons étendre la construction des fonctions Ψ de Stange :

Théorème 3.1.1. *Soit $\mathcal{E}$ une courbe elliptique, pour le moment définie sur $\mathbb{C}$. Soient $a, b \in \{0, 1/2\}$ des caractéristiques semi-entières.*
Pour $\mathbf{v} = (v_1, v_2) \in \mathbb{Z}^2$,

$$\Psi_{\mathbf{v}}(z_1, z_2) = \frac{\theta \begin{bmatrix} a \\ b \end{bmatrix} (v_1 z_1 + v_2 z_2)}{\theta \begin{bmatrix} a \\ b \end{bmatrix} (z_1)^{v_1^2 - v_1 v_2} \theta \begin{bmatrix} a \\ b \end{bmatrix} (z_1 + z_2)^{v_1 v_2} \theta \begin{bmatrix} a \\ b \end{bmatrix} (z_2)^{v_2^2 - v_1 v_2}}$$

est une fonction bien définie sur $\mathcal{E} \times \mathcal{E}$.

Démonstration. Pour le premier point, il s'agit de montrer que pour tout couple $(N_1, N_2) \in \mathbb{N}^2$, on a

$$\Psi_{\mathbf{v}}(z_1 + N_1, z_2 + N_2) = \Psi_{\mathbf{v}}(z_1, z_2) \text{ et } \Psi_{\mathbf{v}}(z_1 + N_1\tau, z_2 + N_2\tau) = \Psi_{\mathbf{v}}(z_1, z_2).$$

Etudions la première égalité. En se servant du fait que pour tout entier n,

$$\theta \begin{bmatrix} a \\ b \end{bmatrix} (z + n, \tau) = \exp(2\pi i a n)\ \theta \begin{bmatrix} a \\ b \end{bmatrix} (z, \tau),$$

on a

$$\Psi_{\mathbf{v}}(z_1 + N_1, z_2 + N_2) = \exp(2\pi i a A) \Psi_{\mathbf{v}}(z_1, z_2),$$

avec

$$\begin{aligned} A =& v_1 N_1 + v_2 N_2 - (v_1^2 - v_1 v_2) N_1 - v_1 v_2 (N_1 + N_2) - (v_2^2 - v_1 v_2) N_2 \\ =& (v_1 - v_1^2) N_1 + (v_2 - v_2^2) N_2. \end{aligned}$$

On remarque alors que $A \equiv 0 \pmod 2$, ce qui nous permet bien de conclure que

$$\Psi_{\mathbf{v}}(z_1 + N_1, z_2 + N_2) = \Psi_{\mathbf{v}}(z_1, z_2).$$

On passe maintenant à la seconde égalité. La formule dont on doit se servir est la suivante : pour tout entier n,

$$\theta \begin{bmatrix} a \\ b \end{bmatrix} (z + n\tau, \tau) = \exp\left(-2\pi i b n - \pi i n^2 \tau - 2\pi i n z\right)\ \theta \begin{bmatrix} a \\ b \end{bmatrix} (z, \tau).$$

Ainsi, on a

$$\Psi_{\mathbf{v}}(z_1 + N_1\tau, z_2 + N_2\tau) = \exp(-2\pi i b A - \pi i \tau B - 2\pi i C) \Psi_{\mathbf{v}}(z_1, z_2),$$

où A est l'entier calculé plus haut (on a donc toujours $\exp(-2\pi i b A) = 0$), alors que B et C sont deux quantités qui restent à calculer, et que nous allons voir

nuls.
En effet, on a

$$\begin{aligned} B =&(v_1N_1 + v_2N_2)^2 - (v_1^2 - v_1v_2)N_1^2 - v_1v_2(N_1 + N_2)^2 - (v_2^2 - v_1v_2)N_2^2 \\ =&0. \end{aligned}$$

et

$$\begin{aligned} C =&(v_1N_1 + v_2N_2)(v_1z_1 + v_2z_2) - (v_1^2 - v_1v_2)N_1z_1 - v_1v_2(N_1 + N_2)(z_1 + z_2) - (v_2^2 - v_1v_2)N_2z_2 \\ =&0. \end{aligned}$$

Ainsi, on a bien

$$\Psi_{\mathbf{v}}(z_1 + N_1\tau, z_2 + N_2\tau) = \Psi_{\mathbf{v}}(z_1, z_2),$$

et donc $\Psi_{\mathbf{v}}$ est bien une fonction bien définie sur $\mathcal{E} \times \mathcal{E}$.

□

Il reste alors à faire le lien entre ces nouvelles fonctions Ψ et les elliptic nets :

Proposition 3.1.2. *Soient $a, b \in \{0, 1/2\}$, et $P, Q \in \mathcal{E}$. La fonction*

$$\begin{aligned} W : \mathbb{Z}^2 &\to \mathbb{C} \\ (p, q) &\mapsto \Psi_{p,q}(P, Q) \end{aligned}$$

définie dans le théorème précédent 3.1.1 est un elliptic net au sens de Stange.

Démonstration. Pour $1 \leq i \leq 4$, soient $u_{i,1}$ et $u_{i,2} \in \mathbb{Z}$. Il faut voir que

$$\det\left(W(u_{i,1} + u_{j,1}, u_{i,2} + u_{j,2},)W(u_{i,1} - u_{j,1}, u_{i,2} - u_{j,2},)\right) = 0.$$

En utilisant les formules de Riemann, (qui se trouvent par exemple dans [34] p.22), cela peut se faire via un logiciel de calcul formel.

□

Ainsi, on a vu que les fonctions thêta de caractéristiques semi entières fournissent des elliptic nets. On passe maintenant aux autres fonctions de niveau l, pour l pair, en commençant par le cas $l = 2$.

3.2 Fonctions de niveau 2 et elliptic nets

Dans cette section et la suivante, je travaille avec $\mathcal{E}/\mathbb{K}$ une courbe elliptique définie sur un corps quelconque $\mathbb{K}$. La variété de Kummer de $\mathcal{E}$ est notée $\mathcal{K}$ (je

rappelle que $\mathcal{K} \cong \mathcal{E}/\{\pm 1\}$).

Les deux théorèmes suivants nous donnent deux méthodes possibles en double-and-add pour obtenir des quantités du type $\kappa_1([r]P+Q)$ pour un grand premier r. La première est d'utiliser la loi d'addition vérifiée par $\{\kappa_1, \kappa_2\}$:

Théorème 3.2.1. *On reprend $\kappa = [\kappa_1 : \kappa_2] : \mathcal{E} \to \mathbb{P}_1$ un plongement projectif de la variété de Kummer de $\mathcal{E}$, avec $\{\kappa_1, \kappa_2\}$ une base de $\mathcal{L}(2\mathcal{O})$.*

Alors il existe des applications bilinéaires B_{11}, B_{12} et B_{22} telles que, pour tous points P et $Q \in \mathcal{E}$, l'on ait

- $\kappa_1(P+Q)\kappa_1(P-Q) = B_{11}\left(\overrightarrow{\kappa(P)_2}, \overrightarrow{\kappa(Q)_2}\right)$,
- $\kappa_1(P+Q)\kappa_2(P-Q) + \kappa_1(P-Q)\kappa_2(P+Q) = B_{12}\left(\overrightarrow{\kappa(P)_2}, \overrightarrow{\kappa(Q)_2}\right)$,
- $\kappa_2(P+Q)\kappa_2(P-Q) = B_{22}\left(\overrightarrow{\kappa(P)_2}, \overrightarrow{\kappa(Q)_2}\right)$,

où $\overrightarrow{\kappa(P)_2}$ et $\overrightarrow{\kappa(Q)_2}$ désignent les vecteurs colonnes suivants :

$$\overrightarrow{\kappa(P)_2} = \begin{pmatrix} \kappa_1(P)^2 \\ \kappa_1(P)\kappa_2(P) \\ \kappa_2(P)^2 \end{pmatrix}, \overrightarrow{\kappa(Q)_2} = \begin{pmatrix} \kappa_1(Q)^2 \\ \kappa_1(Q)\kappa_2(Q) \\ \kappa_2(Q)^2 \end{pmatrix},$$

Démonstration. Si on prouve l'existence de telles B_i pour une base particulière $\{v_1, v_2\}$ de $\mathcal{L}(2\mathcal{O})$, alors le résultat sera vrai pour toutes les autres bases. Il se trouve justement que la proposition 1.3.18 nous donne trois bases différentes satisfaisant cette propriété.

□

La seconde est d'utiliser l'algorithme des elliptic nets :

Théorème 3.2.2. *Dans le même contexte, κ_1 et κ_2 vérifient la récurrence des elliptic nets :*

$$\det\left(\kappa_1(x_i+x_j)\kappa_1(x_i-x_j)\right)_{1\leq i,j\leq 4} = \det\left(\kappa_2(x_i+x_j)\kappa_2(x_i-x_j)\right)_{1\leq i,j\leq 4} = 0.$$

Démonstration. Notons :

— A_{11} et A_{22} les matrices associées aux applications bilinéaires B_{11} et B_{22} établies dans le théorème précédent,

— pour $1 \leq i \leq 4$, $\overrightarrow{\kappa_i} = \overrightarrow{\kappa(x_i)_2}$.

On a donc, pour tout $1 \leq i, j \leq 4$,

$$\kappa_1(x_i+x_j)\kappa_1(x_i-x_j) = {}^t\overrightarrow{\kappa_i} A_{11} \overrightarrow{\kappa_j}.$$
$$\kappa_2(x_i+x_j)\kappa_2(x_i-x_j) = {}^t\overrightarrow{\kappa_i} A_{22} \overrightarrow{\kappa_j}.$$

Les deux matrices $({}^t\overrightarrow{\kappa_i} A_{11} \overrightarrow{\kappa_j})_{1\leq i,j\leq 4}$ et $({}^t\overrightarrow{\kappa_i} A_{22} \overrightarrow{\kappa_j})_{1\leq i,j\leq 4}$ sont de rang ne dépassant pas 3, et donc de déterminant nul.

□

3.3 Fonctions de niveau l (pair) et elliptic nets

D'un point de vue pratique, le cas $l = 2$ est le plus efficace. Toutefois, il n'est pas compliqué de généraliser les théorèmes 3.2.1 et 3.2.2 aux niveaux plus élevés. Ce sont les deux théorèmes suivants :

Théorème 3.3.1. *Soit $m = l(l+1)/2$. On fixe deux indices $1 \leq i, j \leq l$. Il existe alors une application bilinéaire $B_{i,j}$ telle que pour tous P et $Q \in \mathcal{E}$ l'on ait*

$$\kappa_i(P+Q)\kappa_j(P-Q) + \kappa_j(P+Q)\kappa_i(P-Q) = B_{i,j}\left(\overrightarrow{\kappa(P)_2}, \overrightarrow{\kappa(Q)_2}\right),$$

où $\overrightarrow{\kappa(P)_2}$ et $\overrightarrow{\kappa(Q)_2}$ sont les vecteurs colonnes $m \times 1$:

$$\overrightarrow{\kappa(P)_2} = \begin{pmatrix} \kappa_1(P)^2 \\ \vdots \\ \kappa_a(P)\kappa_b(P) \\ \vdots \\ \kappa_l(P)^2 \end{pmatrix}, \overrightarrow{\kappa(Q)_2} = \begin{pmatrix} \kappa_1(Q)^2 \\ \vdots \\ \kappa_a(Q)\kappa_b(Q) \\ \vdots \\ \kappa_l(Q)^2 \end{pmatrix},$$

Démonstration. Comme précédemment, il suffit d'établir le résultat pour un choix particulier de bases et il sera vrai pour toutes les autres. Et, comme précédemment, les fonctions thêta nous donnent les relations voulues : le résultat se trouve par exemple dans le théorème 1 de [30]. □

Théorème 3.3.2. *Avec les notations précédentes, on a pour tout $n > m$*

$$\det\left(\kappa_i(x_a + x_b)\kappa_i(x_a - x_b)\right)_{1 \leq a,b \leq n} = 0.$$

Démonstration. C'est exactement la même preuve que pour le théorème 3.2.2. □

Remarque 3.3.3.

- *Si l est impair, on peut toujours calculer les fonctions de Miller avec les fonctions de niveau l. Par contre, on ne dispose plus du théorème 3.3.1 établissant les formules d'addition pour mes fonctions de niveau l : on est donc privé de tout moyen pour aller chercher la quantité $\kappa_1(rP+Q)$.*
- *Pour $l = 2$ on a $m = 3$ et donc, avec $n > m$, la matrice du théorème 3.3.2 est de taille au moins 4 : on retombe bien sur la définition des elliptic nets. Si $l > 2$ par contre, on obtient un résultat moins fort, qui nécessite de prendre des matrices de plus grande taille, et donnent donc des formules plus compliquées à utiliser.*

3.4 Fonctions de niveau l (toujours pair) et hyperelliptic nets

Jusqu'ici dans ce chapitre, j'ai travaillé en genre 1 : c'est la situation la plus courante en cryptographie, et c'est la plus simple à comprendre. En réalité, les résultats s'étendent de façon immédiate au cas hyperelliptique. J'énonce donc ici les équivalents des théorèmes 3.3.1 et 3.3.2.

Dans cette section, je travaille donc sur une courbe $\mathcal{C}$ de genre g. Je me donne un entier $l \geq 2$ pair, et $\{\kappa_1, \ldots, \kappa_{l^2}\}$ une base de $\mathcal{L}(l\Theta)$.

Théorème 3.4.1. *On pose $m = l^g(l^g+1)/2$. On fixe deux indices $1 \leq i, j \leq l^g$. Il existe une application bilinéaire $B_{i,j}$ telle que pour tous D_1 et $D_2 \in \mathcal{J}$ on ait*

$$\kappa_i(D_1+D_2)\kappa_j(D_1-D_2) + \kappa_j(D_1+D_2)\kappa_i(D_1-D_2) = B_{i,j}\left(\overrightarrow{\kappa(D_1)_2}, \overrightarrow{\kappa(D_2)_2}\right),$$

avec $\overrightarrow{\kappa(D_1)_2}$ et $\overrightarrow{\kappa(D_2)_2}$ les vecteurs $m \times 1$ suivants :

$$\overrightarrow{\kappa(D_1)_2} = \begin{pmatrix} \kappa_1(D_1)^2 \\ \vdots \\ \kappa_a(D_1)\kappa_b(D_1) \\ \vdots \\ \kappa_{l^2}(D_1)^2 \end{pmatrix}, \overrightarrow{\kappa(D_2)_2} = \begin{pmatrix} \kappa_1(D_2)^2 \\ \vdots \\ \kappa_a(D_2)\kappa_b(D_2) \\ \vdots \\ \kappa_{l^2}(D_2)^2 \end{pmatrix},$$

Démonstration. Les formules d'addition de Riemann montrent que les fonctions thêta admettent bien de telles relations bilinéaires. Ainsi, par exemple, le théorème 1 de [30] montre que le théorème est vrai pour la base

$$\left\{\theta\begin{bmatrix} 0 \\ b \end{bmatrix}\left(z, l^{-1}\Omega\right) \mid b \in \left(\frac{1}{l}\mathbb{Z}^g/\mathbb{Z}^g\right)\right\}.$$

Comme on a démontré le résultat pour une base donnée, il est vrai pour toutes les bases. □

Théorème 3.4.2. *Avec les notations du théorème précédent, κ_i vérifie ce que j'appelle la récurrence des hyperelliptic nets généralisés : pour tout $n > m$,*

$$\det\left(\kappa_i(x_a+x_b)\kappa_i(x_a-x_b)\right)_{1\leq a,b\leq n} = 0.$$

Démonstration. Reprenons la démonstration du théorème 3.2.2, et notons :

— A_{ij} les matrices associées aux applications bilinéaires B_{ij} établies dans le théorème précédent,

— pour $1 \leq a \leq n$, $\overrightarrow{\kappa_a}$ le vecteur $m \times 1$ $\overrightarrow{\kappa(D_a)_2}$.

Autrement dit, pour tout couple (a, b) avec $1 \leq a, b \leq l^g$, et pour tout entier $1 \leq i \leq l$, la situation est la suivante :

$$\kappa_i(D_a + D_b)\kappa_i(D_a - D_b) = {}^t\overrightarrow{\kappa_a} A_{ii} \overrightarrow{\kappa_b}.$$

Or, les matrices $({}^t\overrightarrow{\kappa_a} A_{ii} \overrightarrow{\kappa_b})_{1 \leq a,b \leq n}$, pour tout $1 \leq i \leq l$, sont de rang ne dépassant pas m, et on a pris soin dans les hypothèses du théorème de prendre ces matrices de taille $n > m$: leur déterminant est nul.

□

3.5 Conclusion

Dans ce chapitre, on a commencé par voir que la construction des hyperelliptic nets avec la fonction sigma pouvaient se généraliser aux autres fonctions thêta de caractéristiques semi entières. Plus encore, toutes les fonctions de niveau 2 nous fournissent des hyperelliptic nets, tandis que les fonctions de niveau pair supérieur aboutissent à une relation plus faible, appelée ici relation des hyperelliptic nets généralisée.
Surtout, ce chapitre permet de faire le lien entre l'algorithme des fonctions thêta pour le calcul de couplages, et celui des nets : la formule de récurrence des nets est une conséquence directe des formules d'addition qui sont à la base de l'algorithme des fonctions thêta.
Ainsi, pour un même outil (un ensemble de coordonnées de niveau pair), il existe deux méthodes pour le calcul de couplages. Nous avons vu dans le chapitre précédent que, pour les fonctions thêta, les formules d'addition donnaient un algorithme bien plus efficace que la relation des nets. Une question ouverte serait de trouver une famille de coordonnées pour laquelle les formules d'addition sont optimales.

Chapitre 4

Polynômes de sommation

4.1 Introduction

La sécurité de l'utilisation des courbes hyperelliptiques en cryptographie repose sur la difficulté du problème du logarithme discret. Les algorithmes pour traiter ce problème sur un groupe $\mathbb{G}$ se classent en deux catégories :

1. les algorithmes génériques, c'est-à-dire qui sont valables sur tout groupe ; si $\mathbb{G}$ est un groupe cyclique d'ordre premier n, ces algorithmes ont un coût en $O(\sqrt{n})$, aussi sont-ils également appelés algorithmes en racine carré ;
2. et puis il y a les algorithmes sous-exponentiels, qui utilisent la structure du groupe étudié pour aller plus vite.

Ainsi, pour les courbes elliptiques définies sur une extension $\mathbb{F}_{q^n}/\mathbb{F}_q$, Gaudry combina dans [22] les polynômes de sommation de Semaev ([43]) et la descente de Weil pour produire un algorithme de calcul d'index dont la complexité est sous-exponentielle.
Dans le cas des courbes hyperelliptiques, l'idée générale de l'algorithme de Gaudry reste valable. Malheureusement, jusqu'ici aucun équivalent des polynômes de Semaev ne fût trouvé. Ainsi, Nagao ([36]) passa par les espaces de Riemann-Roch pour exhiber des systèmes polynomiaux à résoudre pour la phase de recherche de relations dans le calcul d'index.

Dans ce chapitre, je vais commencer par rappeler la construction des polynômes de sommation par Semaev et leur utilisation par Gaudry pour le calcul d'index. Dans le but d'étendre ces polynômes aux courbes hyperelliptiques, je vais avoir besoin de donner une construction alternative en genre 1, utilisant la fonction sigma de Weierstrass. Je montrerai ensuite l'impact des nouveaux polynômes de sommation hyperelliptique sur le problème du logarithme discret. Enfin, je discuterai de divers choix possibles de polynômes alternatifs.

4.2 Rappels sur les travaux de Gaudry et Semaev

4.2.1 Cadre général : le calcul d'index

Je commence par rappeler le principe général du calcul d'index, avant de considérer spécifiquement le groupe généré par un point $P \in \mathcal{E}(\mathbb{F}_{q^k})$.

Soit $\mathbb{G} = \langle P \rangle$ un groupe général. La loi de groupe est notée additivement et le neutre désigné $\mathcal{O}$. Soit $Q \in \mathbb{G}$, on veut trouver un entier n tel que

$$Q = nP.$$

L'algorithme s'initialise par la construction d'un sous ensemble de $\mathbb{G}$ appelé base de factorisation $\mathcal{F} = \{F_1, \ldots, F_k\}$. Ce sera plus explicite par la suite, mais $\mathcal{F}$ est choisi de façon à ce que

— pour tout élément $R \in \mathbb{G}$ il soit facile de décider si R peut se décomposer

$$R = m_1 F_1 + \cdots + m_k F_k, \text{ avec } \forall i,\ m_i \in \mathbb{Z} \tag{4.1}$$

(un tel élément est dit $\mathcal{F}$-friable, ou juste friable) ;

— et, lorsque c'est le cas, il soit facile d'exhiber une telle décomposition. En pratique, cette décomposition s'obtient par la résolution d'un système polynomial.

L'algorithme de calcul d'index en lui même se déroule en deux étapes :

(**1**) **la recherche de relations** :
On génère aléatoirement des entiers α et $\beta \in \mathbb{N}$, et on construit $R = \alpha P + \beta Q$, jusqu'à obtenir un point R qui soit friable, et on calcule alors sa décomposition. On répète l'opération jusqu'à obtenir $k+1$ relations

$$\forall i,\ R_i = \alpha_i P + \beta_i Q = \sum_{j=1}^{k} m_{i,j} F_j,$$

où k est le cardinal de $\mathcal{F}$. On stocke ces relations dans une matrice $M = (m_{ij})$, qui est donc de taille $(k+1) \times k$.

(**2**) **la phase d'algèbre linéaire** :
À part cas exceptionnel (auquel cas on recommence l'étape 1), il y a un vecteur non nul (γ_i) dans le noyau de M. Un tel vecteur nous donne

$$\sum_i \gamma_i (\alpha_i P + \beta_i Q) = \mathcal{O}.$$

Encore une fois, sauf en cas de malchance, on a $\sum_i \gamma_i\beta_i \neq 0$, ce qui nous permet de déduire :

$$Q = -\left(\frac{\sum_i \gamma_i\alpha_i}{\sum_i \gamma_i\beta_i}\right) P,$$

où la quantité $-\left(\frac{\sum_i \gamma_i\alpha_i}{\sum_i \gamma_i\beta_i}\right)$ est calculée modulo le cardinal de $\mathbb{G}$.

Le coût de la construction de $\mathcal{F}$ et de la phase 1 dépend du groupe $\mathbb{G}$ et de sa représentation. Plus précisément, l'analyse de l'algorithme prend en compte

— le coût de l'arithmétique dans $\mathbb{G}$;
— la probabilité pour un élément R aléatoire d'être friable ;
— le coût du test de friabilité ;
— le coût du calcul de la décomposition d'un élément friable.

Au contraire, la phase d'algèbre linéaire ne dépend que de k. Ainsi, quand k augmente, les relations de la phase 1 s'obtiennent plus rapidement, tandis que la phase 2 devient plus lente. L'idée est donc de trouver k pour équilibrer les coûts des deux phases.

4.2.2 L'algorithme dans le cas des courbes elliptiques et hyperelliptiques

Spécifier l'algorithme de calcul d'index pour un groupe $\mathbb{G}$ donné, c'est préciser la base de factorisation $\mathcal{F}$, ainsi que la phase de recherche de relations.

Dans le cas où $\mathbb{G} =< P >\subset \mathcal{E}(\mathbb{F}_{q^n})$ est généré par un point d'une courbe elliptique, le choix de $\mathcal{F}$ fait par Gaudry est

$$\mathcal{F} = \{R \in \mathcal{E}(\mathbb{F}_{q^n})|\ x(R) \in \mathbb{F}_q\}.$$

L'algorithme s'étend aux courbes hyperelliptiques. Dans ce cas, la base de factorisation est

$$\mathcal{F} = \{D \sim (P) - (\infty)|\ x(P) \in \mathbb{F}_q\}.$$

Il reste maintenant à expliquer comment tester une relation.
Pour le cas hyperelliptique, la solution donnée par Gaudry dans [22] est de calculer le polynôme u de la représentation de Mumford (voir chapitre 1) du diviseur $R = \alpha P + \beta Q$, et de tester s'il se scinde totalement dans $\mathbb{F}_q$.
Dans le cas elliptique, la méthode la plus rapide est d'utiliser les polynômes de Semaev que j'expose à présent.

4.2.3 Polynômes de Semaev

Ici, je rapporte les principaux résultats de [43] : la définition des polynômes de sommation et leurs propriétés. On se donne $\mathcal{E}$ une courbe elliptique définie sur un corps $\mathbb{K}$ de caractéristique différente de 2, dont l'équation de Weierstrass est $y^2 = x^3 + Ax + B$.

Définition 4.2.1 ([43])**.** Pour tout entier $n \geq 2$, le polynôme de sommation $S_n \in \mathbb{K}[X_1, \ldots, X_n]$ est défini par la récurrence :

$$\begin{aligned} S_2(X_1, X_2) =& X_1 - X_2, \\ S_3(X_1, X_2, X_3) =& (X_1 - X_2)^2 X_3^2 - 2((X_1 + X_2)(X_1X_2 + A) + 2B)X_3 \\ &+ ((X_1X_2 - A)^2 - 4B(X_1 + X_2)), \end{aligned}$$

$\forall n \geq 4, 1 \leq k \leq n-3$,

$$S_n(X_1, \ldots, X_n) = \mathrm{Res}_X(S_{n-k}(X_1, \ldots, X_{n-k-1}, X), S_{k+2}(X_{n-k}, \ldots, X_n, X)).$$

L'intérêt de ces polynômes réside dans le théorème suivant :

Théorème 4.2.2 ([43])**.** *Soit S_n le n-ième polynôme de sommation de $\mathcal{E}/\mathbb{K}$. Soit $x_1, \ldots, x_n$ dans $\overline{\mathbb{K}}$.*
Alors $S_n(x_1, \ldots, x_n) = 0$ si et seulement si il existe $y_1, \ldots, y_n$ dans $\overline{\mathbb{K}}$ tel que pour tout i, $P_i = (x_i, y_i)$ soit un point de $\mathcal{E}$ et

$$P_1 + \cdots + P_n = \mathcal{O}.$$

Remarque 4.2.3. *On peut réécrire ce théorème de la façon suivante :*
ayant les x_i tels que $S_n(x_1, \ldots, x_n) = 0$, le théorème me dit qu'il existe pour chaque i deux choix de y_i tels que (x_i, y_i) soit un point de $\mathcal{E}$ (si y_i est un de ces choix, l'autre est évidemment $-y_i$). Une fois les bons choix effectués, j'obtiens des points $P_1, \ldots, P_n$ tels que

$$P_1 \pm \cdots \pm P_n = \mathcal{O}.$$

On revient à notre algorithme de calcul d'index. Soit $R = (x_R, y_R)$ un point de $\mathcal{E}(\mathbb{F}_{q^n})$ que l'on veut décomposer comme somme de n points de $\mathcal{F}$. On veut donc résoudre

$$S_{n+1}(x_R, X_1, \ldots, X_n) = 0, \ \forall i, X_i \in \mathbb{F}_q.$$

On décompose alors x_R et chacun des coefficients de S_{n+1} dans la base $\{1, t, \ldots, t^{n-1}\}$ de $\mathbb{F}_{q^n}/\mathbb{F}_q$ choisie, et on obtient donc une équation de la forme

$$\sum_{i=0}^{k-1} S_{n+1}^{(i)}(X_1, \ldots, X_n)t^i = 0,$$

où chaque $S_{n+1}^{(i)}$ est dans $\mathbb{F}_q[X_1, \ldots, X_n]$, soit donc un système de n équations à n indéterminées dans $\mathbb{F}_q$.

4.3 Nouvelle construction des polynômes de sommation en genre 1

Il est maintenant temps pour moi d'exposer mes contributions à ce problème de logarithme discret.
Comme énoncé en introduction dans cette partie, je me suis attaché à étendre la notion de polynôme de sommation aux courbes hyperelliptiques.
Dans le cas elliptique, Nous avons vu que les polynômes de Semaev sont définis par récurrence via un calcul de résultant. Ceci ne sera plus le cas dans le cas hyperelliptique, aussi ai-je été obligé de réécrire ces polynômes en genre 1 avec d'autres formules, en utilisant la fonction sigma de Weierstrass.
La définition et les propriétés élémentaires de cette fonction ont été rappelées au chapitre 1. Je me contenterai donc ici de seulement rapporter les deux théorèmes sur cette fonction que j'utiliserais spécifiquement.

4.3.1 Deux théorèmes fondamentaux

Les théorèmes cités ici sont anciens : ainsi, le premier est dû à Weierstrass ([53], [54]), tandis que le second a été énoncé en premier par Frobenius et Stickelberger dans [17].

Théorème 4.3.1 (**Loi d'addition**).
Soit $u_1, u_2 \in \mathcal{E}$.
En posant $x_1 = x(u_1)$ *et* $x_2 = x(u_2)$, *on a*

$$\frac{\sigma(u_1 - u_2)\sigma(u_1 + u_2)}{\sigma^2(u_1)\sigma^2(u_2)} = \begin{vmatrix} 1 & x_1 \\ 1 & x_2 \end{vmatrix}.$$

Je rappelle que σ admet un zéro d'ordre 1 uniquement en $\mathcal{O}$.
Ce théorème est donc très facile à lire : deux points u_1 et u_2 vérifient $u_1 \pm u_2 = \mathcal{O}$ si et seulement si $x_1 = x_2$.
De plus, comme la fonction x admet comme seul pôle le point $\mathcal{O}$ (et c'est un pôle d'ordre 2), un tel test est possible si et seulement si on a bien vérifié que $u_1 \neq \mathcal{O}$ et $u_2 \neq \mathcal{O}$.

Le théorème suivant est une généralisation du précédent pour $n \geq 2$ points :

Théorème 4.3.2 (Frobenius-Stickelberger).
Soit $n \geq 2$ *un entier, et* $u_1, \ldots, u_n \in \mathcal{E}$, *avec* $u_i = (x_i, y_i)$. *On a l'égalité*

$$(-1)^{(n-1)(n-2)/2}\frac{\sigma(u_1+\cdots+u_n)\prod_{i<j}\sigma(u_i-u_j)}{\left(\prod_i\sigma(u_i)\right)^n}=\begin{vmatrix}1 & x_1 & y_1 & x_1^2 & y_1x_1 & x_1^3 & \dots\\ 1 & x_2 & y_2 & x_2^2 & y_2x_2 & x_2^3 & \dots\\ & & & \vdots & & & \\ 1 & x_n & y_n & x_n^2 & y_nx_n & x_n^3 & \dots\end{vmatrix},$$

où la matrice est de taille $n\times n$.
Plus précisément, sa i-ième ligne est l'évaluation des fonctions

$$1, x, \dots, x^a, y, yx, \dots, yx^b$$

sur le point u_i, avec $a=\lfloor\frac{n}{2}\rfloor$ et $b=\lfloor\frac{n-3}{2}\rfloor$.

Comme pour 4.3.1, si je ne donne pas la démonstration précise de ce théorème classique, il est facile de le comprendre :

$$\begin{aligned}\sigma(u_1+\cdots+u_n)=0 &\Leftrightarrow (u_1)+\cdots+(u_n)-n(\infty)\sim\mathcal{O}\\ &\Leftrightarrow \exists f\in\mathcal{L}(n\infty)\text{ t.q. }\forall i,\ f(u_i)=0.\end{aligned}$$

Or, une base de l'espace de Riemann Roch $\mathcal{L}(n\infty)$ est justement $\{1, x, \dots, x^a, y, yx, \dots, yx^b\}$ [1] :

$\sigma(u_1+\cdots+u_n)=0 \Leftrightarrow$ il existe des coefficients A_i, B_i t.q.

$$\forall j,\ \sum_{i=0}^{a}A_ix_j^i+\sum_{i=0}^{b}B_iy_jx_j^i=0$$

$\Leftrightarrow$ le déterminant du membre droit du théorème s'annule.

Pour recoller les deux morceaux, il suffit d'écrire

$$f(x,y)=\sum_{i=0}^{a}A_ix^i+\sum_{i=0}^{b}B_iyx^i.$$

On peut d'ailleurs rapidement faire le lien entre mon travail dans ce chapitre et celui de Nagao dans [36] : basiquement, je teste des relations de décomposition en testant des annulations de déterminant, quand lui préfère chercher l'existence des coefficients A_i et B_i pour construire la fonction f.

Remarque 4.3.3.

1. on vérifiera aisément que le cardinal de cette famille est bien $a+b+2=n$, soit la dimension de $\mathcal{L}(n\infty)$.

- *En particulier, le déterminant de cette matrice est un polynôme en $2n$ variables (les x_i et les y_i), de degré 1 en chaque y_i et $a = \lfloor \frac{n}{2} \rfloor$ en chaque x_i.*
- *Ce théorème ne nous donne pas directement les polynômes de Semaev. En effet, si on reste loin des cas dégénérés ($u_i = \mathcal{O}$ ou $u_i \pm u_j = \mathcal{O}$) le déterminant s'annule si et seulement si $u_1 + \cdots + u_n = \mathcal{O}$, ce qui n'est pas exactement la propriété des polynômes de Semaev. Plus gênant, le déterminant qui apparaît nous donne un polynôme en $2n$ variables au lieu de n : il faudra trouver un moyen de se débarrasser des y_i.*
- *C'est un raisonnement qu'on a déjà vu dans le chapitre 2 : à priori, la fonction σ est définie sur $\mathbb{C}$. Mais les déterminants apparaissant dans les deux théorèmes nous donnent des tests polynomiaux ($u_1 + \cdots + u_n = \mathcal{O}$ si et seulement si le déterminant s'annule) valables sur tout corps $\mathbb{K}$ sur lequel on souhaiterait travailler, en particulier les corps finis en cryptographie.*

4.3.2 Construction

Ici apparaissent les premiers résultats originaux de cette partie. Je commence par quelques notations :

Notations :

— pour $n \geq 2$, je note $\Delta_n(u_1, \ldots, u_n)$ le déterminant apparaissant dans le théorème 4.3.2 :

$$\Delta_n(u_1, \ldots, u_n) = \begin{vmatrix} 1 & x_1 & y_1 & x_1^2 & y_1x_1 & x_1^3 & \ldots \\ 1 & x_2 & y_2 & x_2^2 & y_2x_2 & x_2^3 & \ldots \\ & & & \vdots & & & \\ 1 & x_n & y_n & x_n^2 & y_nx_n & x_n^3 & \ldots \end{vmatrix};$$

— je note $\mathbb{E}_n \subset \{\pm 1\}^n$ l'ensemble

$$\mathbb{E}_n = \{\epsilon = (\epsilon_1, \ldots, \epsilon_n) | \epsilon_1 = 1\} \subset \{\pm 1\}^n.$$

En particulier, une expression du type $\epsilon \cdot u$ signifiera $u_1 \pm u_2 \cdots \pm u_n$.

Avant d'exprimer les polynômes de Semaev avec la fonction sigma, je donne un lemme de réduction :

Lemme 4.3.4.

(i) La fonction $\Delta_n(u_1, u_2, \ldots, u_n) \times \Delta_n(u_1, -u_2, \ldots, u_n)$ est un polynôme en les $2n$ variables x_i, y_i.

(ii) Tous ses monômes sont de degré pair en y_2, en particulier on peut utiliser la relation

$$y_2^2 = x_2^3 + Ax_2 + B$$

pour obtenir un polynôme indépendant de y_2.

(iii) En faisant ainsi, on obtient un polynôme divisible par

$$\prod_{i\neq 2}(x_i - x_2) = \prod_{i\neq 2}\Delta_2(u_2, u_i).$$

Démonstration. On étudie la fonction

$$f(u_1,\dots,u_n) = \begin{vmatrix} 1 & x_1 & y_1 & x_1^2 & y_1x_1 & x_1^3 & \dots \\ 1 & x_2 & y_2 & x_2^2 & y_2x_2 & x_2^3 & \dots \\ & & & \vdots & & & \\ 1 & x_n & y_n & x_n^2 & y_nx_n & x_n^3 & \dots \end{vmatrix} \cdot \begin{vmatrix} 1 & x_1 & y_1 & x_1^2 & y_1x_1 & x_1^3 & \dots \\ 1 & x_2 & -y_2 & x_2^2 & -y_2x_2 & x_2^3 & \dots \\ & & & \vdots & & & \\ 1 & x_n & y_n & x_n^2 & y_nx_n & x_n^3 & \dots \end{vmatrix}.$$

(i) Le premier point est clair.

(ii) On a

$$f(u_1, -u_2, \dots, u_n) = f(u_1, u_2, \dots, u_n).$$

Or, $-u_2 = (x_2, -y_2)$: la fonction f est donc paire en y_2.

(iii) Avec l'observation que

$$x_2 = x_i \Leftrightarrow u_2 = \pm u_i \text{ dans } \mathcal{E},$$

il ne reste qu'à démontrer que

$$f(u_1, u_2, u_3, \dots, u_2, \dots, u_n) = f(u_1, u_2, u_3, \dots, -u_2, \dots, u_n) = 0,$$

avec u_2 (respectivement $-u_2$) remplaçant u_i.
Mais si $u_i = u_2$, alors dans la première matrice de la définition de f, les seconde et i-ième lignes sont égales, donc son déterminant (et donc f) s'annule. De même, si $u_i = -u_2$, alors c'est le second déterminant qui s'annule.

□

Ainsi, si je multiplie chaque identité de Frobenius-Stickelberger (théorème 4.3.2) que j'obtiens pour chaque choix de signes $\epsilon \in \mathbb{E}_n$, je peux utiliser le lemme de réduction pour simplifier le polynôme obtenu en une expression indépendante des y_i. Comme je le démontrerai un peu plus loin, cette expression est alors égale au polynôme S_n.

Théorème 4.3.5.
(i) Pour $u_1, \dots, u_n \in \mathcal{E}$,

$$\frac{\prod_{\epsilon\in\mathbb{E}_n}\sigma(\epsilon\cdot u)}{\left(\prod_{i=1}^n \sigma(u_i)\right)^{2^{n-1}}} = \frac{\prod_\epsilon \Delta_n(\epsilon_1 u_1, \dots, \epsilon_n u_n)}{\left(\prod_{i<j}\Delta_2(u_i,u_j)\right)^{2^{n-2}}}.$$

(ii) Comme dans le lemme 4.3.4, on peut utiliser les relations

$$y_i^2 = x_i^3 + Ax_i + B,\ 1 \leq i \leq n,$$

pour obtenir un polynôme en $x_1, \ldots, x_n$ (c'est-à-dire indépendant des y_i).

Démonstration.

(i) On commence par multiplier les différentes relations de Frobenius-Stickelberger obtenus pour chaque choix de $\epsilon \in \mathbb{E}_n$.
Du côté droit de l'égalité, on a le produit de tous les $\Delta_n(\epsilon_1 u_1, \ldots, \epsilon_n u_n)$.
Du côté gauche, on a :

$$\frac{\left(\prod_\epsilon \sigma(\epsilon \cdot u)\right)\left(\prod_\epsilon \prod_{i<j} \sigma(\epsilon_i u_i - \epsilon_j u_j)\right)}{\left(\prod_i \sigma(u_i)^n\right)^{2^{n-1}}}.$$

On rappelle que σ est une fonction impaire. Donc

$$\prod_\epsilon \prod_{i<j} \sigma(\epsilon_i u_i - \epsilon_j u_j) = \prod_{i<j} \left(\sigma(u_i - u_j)\sigma(u_i + u_j)\right)^{2^{n-2}}.$$

Finalement, il ne reste qu'à écrire que

$$\begin{aligned}\left(\prod_{i<j} \Delta_2(u_i, u_j)\right)^{2^{n-2}} &= \prod_{i<j} \left(\frac{\sigma(u_i + u_j)\sigma(u_i - u_j)}{\sigma(u_i)^2\sigma(u_j)^2}\right)^{2^{n-2}} \\ &= \frac{\prod_{i<j} \left(\sigma(u_i - u_j)\sigma(u_i + u_j)\right)^{2^{n-2}}}{\prod_i \sigma(u_i)^{2^{n-1}(n-1)}}.\end{aligned}$$

(ii) A priori, cette quantité est un élément de $\mathbb{K}(x_1, \ldots, x_n, y_1 \ldots, y_n)$.
Le fait qu'on obtienne un élément de $\mathbb{K}(x_1, \ldots, x_n)$ (i.e. indépendant des y_i) est une conséquence du second point du lemme.
Le fait que ce soit un élément de $\mathbb{K}[x_1, \ldots, x_n]$ (i.e. un polynôme) est une conséquence du troisième point du lemme.

□

Définition 4.3.6. Pour $n \geq 2$, la fonction définie dans le théorème précédent est noté $\sigma^{(n)}$:

$$\sigma^{(n)}(u_1, \ldots, u_n) = \frac{\prod_\epsilon \Delta_n(\epsilon_1 u_1, \ldots, \epsilon_n u_n)}{\left(\prod_{i<j} \Delta_2(u_i, u_j)\right)^{2^{n-2}}}.$$

Ainsi qu'on l'a vu, $\sigma^{(n)}$ est indépendant des y_i, aussi noterai-je $\sigma^{(n)}(x_1, \ldots, x_n)$.

Exemple 4.3.7. *Pour $n = 3$ on a*

$$\sigma^{(3)}(u_1,u_2,u_3) = \frac{\sigma(u_1+u_2+u_3)\sigma(u_1-u_2+u_3)\sigma(u_1+u_2-u_3)\sigma(u_1-u_2-u_3)}{\sigma^4(u_1)\sigma^4(u_2)\sigma^4(u_3)}$$
$$= \frac{\begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix} \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & -y_2 \\ 1 & x_3 & y_3 \end{vmatrix} \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & -y_3 \end{vmatrix} \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & -y_2 \\ 1 & x_3 & -y_3 \end{vmatrix}}{\begin{vmatrix} 1 & x_1 \\ 1 & x_2 \end{vmatrix}^2 \begin{vmatrix} 1 & x_1 \\ 1 & x_3 \end{vmatrix}^2 \begin{vmatrix} 1 & x_2 \\ 1 & x_3 \end{vmatrix}^2}$$

Il faut maintenant montrer que $\sigma^{(n)}$ est en fait le n-ième polynôme de Semaev S_n. La démonstration se déroule ainsi :

Proposition 4.3.8.

(i) $\sigma^{(2)}(u_1,u_2) = S_2(x_1,x_2)$ *et* $\sigma^{(3)}(u_1,u_2,u_3) = S_3(x_1,x_2,x_3)$.

(ii) Pour tout n, $\sigma^{(n)}$ est symétrique : pour tout $i<j$,

$$\sigma^{(n)}(x_1,\dots,x_i,\dots,x_j,\dots,x_n) = \sigma^{(n)}(x_1,\dots,x_j,\dots,x_i,\dots,x_n).$$

(iii) Grâce au point (ii), on peut regarder $\sigma^{(n)}$ comme un polynôme en n'importe quel x_i : on choisit $i=1$. Son degré est alors 2^{n-2}, et son coefficient dominant $\left(\sigma^{(n-1)}(x_2,\dots,x_n)\right)^2$.

(iv) On fixe $x_1,\dots,x_{n-1} \in \mathbb{K}$ tels qu'il existe des $y_i \in \overline{\mathbb{K}}$ avec $u_i = (x_i,y_i) \in \mathcal{E}$. Soit $f \in \overline{\mathbb{K}}[X]$ le polynôme défini par

$$f(X) = \sigma^{(n)}(x_1,\dots,x_{n-1},X).$$

Alors les racines de f sont

$$\{x\,(\epsilon_1 u_1 + \cdots + \epsilon_{n-1}u_{n-1})\}.$$

(v) Pour tout $n \geq 4$ et $1 \leq k \leq n-3$,

$$\sigma^{(n)}(X_1,\dots,X_n) = \mathrm{Res}_X(\sigma^{(n-k)}(X_1,\dots,X_{n-k-1},X), \sigma^{(k+2)}(X_{n-k},\dots,X_n,X)).$$

Démonstration. Les deux premiers points sont faciles à vérifier.

(iii) Dans

$$\sigma^{(n)}(u_1,\dots,u_n) = \frac{\prod_\epsilon \Delta_n(\epsilon_1 u_1,\dots,\epsilon_n u_n)}{\left(\prod_{i<j}\Delta_2(u_i,u_j)\right)^{2^{n-2}}},$$

le dénominateur est de degré $(n-1)2^{n-2}$ en x_1. Pour le numérateur :

— si $n=2k$ est pair, le terme de plus haut degré dans Δ_n est x_1^k ;

— si $n=2k+1$ est impair, le terme de plus haut degré dans Δ_n est $y_1 x_1^{k-1}$ (je rappelle qu'après avoir multiplié les différents Δ_n, je remplacerai y_1^2 par $x_1^3 + Ax_1 + B$).

Finalement, le numérateur est de degré $n2^{n-2}$ en x_1, et on obtient ainsi le degré annoncé de $\sigma^{(n)}$.

Pour le coefficient dominant, on développe Δ_n selon la première ligne :

$$\Delta_n(x_1,\dots,x_n) = \begin{cases} \Delta_n(x_2,\dots,x_n)x_1^k + \dots & \text{si n=2k est pair.} \\ \Delta_n(x_2,\dots,x_n)y_1x_1^{k-1} + \dots & \text{si n=2k+1 est impair.} \end{cases}$$

Ainsi, on peut développer le numérateur de $\sigma^{(n)}(u_1,\dots,u_n)$ comme :

$$\begin{aligned}\prod_{\epsilon\in\mathbb{E}_n} \Delta_n(\epsilon_1u_1,\dots,\epsilon_nu_n) &= x_1^{2^{n-2}} \cdot \prod_{\epsilon\in\mathbb{E}_n} \Delta_{n-1}(\epsilon_2u_2,\dots,\epsilon_nu_n) + \dots \\ &= x_1^{2^{n-2}} \cdot \left(\prod_{\epsilon\in\mathbb{E}_{n-1}} \Delta_{n-1}(u_2,\epsilon_3u_3,\dots,\epsilon_nu_n)\right)^2 + \dots\end{aligned}$$

La seconde égalité étant vraie parce que

$$\Delta_{n-1}(u_2,\dots,u_n) = (-1)^{b+1}\Delta_{n-1}(-u_2,\dots,-u_n).$$

(iv) On a

$$\begin{aligned}\sigma^{(n)}(x_1,\dots,x_{n-1},X) = 0 &\Leftrightarrow \exists\epsilon\in\mathbb{E}_n \text{ t.q. } \sigma(u_1+\epsilon_2u_2\dots,\epsilon_nu) = 0 \\ &\Leftrightarrow \exists\epsilon\in\mathbb{E}_n \text{ t.q. } u = \epsilon_1u_1\dots,\epsilon_{n-1}u_{n-1}.\end{aligned}$$

(v) Soient $u_1,\dots,u_n \in \mathcal{E}$ fixés. On note les coordonnées de ces points (X_i,Y_i). On veut alors montrer que

$$\frac{\prod_{\epsilon\in\mathbb{E}_n}\sigma(\epsilon\cdot u)}{\left(\prod_{i=1}^n\sigma(u_i)\right)^{2^{n-1}}} = \mathrm{Res}_u(\sigma^{(n-k)}(u_1,\dots,u_{n-k-1},u),\sigma^{(k+2)}(u_{n-k},\dots,u_n,u)).$$

Le membre de droite de l'égalité est noté *Res*. Pour le calculer, on utilise la formule suivante du résultant :

$$\begin{aligned}&\mathrm{Res}_X(\sigma^{(n-k)}(X_1,\dots,X_{n-k-1},X),\sigma^{(k+2)}(X_{n-k},\dots,X_n,X)) \\ =&a^m\prod_i\sigma^{(n-k)}(X_1,\dots,X_{n-k-1},\alpha_i),\end{aligned}$$

où
— $a = lc_X(\sigma^{(k+2)}(X_{n-k},\dots,X_n,X))$;
— les α_i sont ses racines ;
— $m = deg_X(\sigma^{(n-k)}(X_1,\dots,X_{n-k-1},X))$.

En utilisant le point (*iii*) de la proposition (réécrit avec la symétrie de $\sigma^{(k+1)}$ montrée dans le point (*ii*)) :

$$\begin{aligned}a^m =&[\sigma^{(k+1)}(u_{n-k},\dots,u_n)]^{2^{n-k-1}} \\ =&\frac{\prod\sigma(\epsilon_{n-k}u_{n-k}+\dots+\epsilon_nu_n)^{2^{n-k-1}}}{\left(\prod_{n-k}^n\sigma(u_i)\right)^{2^{n-k-1}2^k}}.\end{aligned}$$

Enfin on utilise le point (iv) de la proposition :

$$\{\alpha_i\} = \{\epsilon_{n-k}u_{n-k} + \cdots + \epsilon_n u_n\}.$$

Notons que c'est un ensemble de cardinal 2^k.

En rassemblant les morceaux :

$$\begin{aligned} Res =& a^m \prod_i \sigma^{(n-k)}(X_1, \ldots, X_{n-k-1}, \alpha_i) \\ =& a^m \cdot \frac{\prod_{\epsilon\in\mathbb{E}_n} \sigma(\epsilon \cdot u)}{\left(\left(\prod_{i=1}^{n-k-1} \sigma(u_i)\right)^{2^{n-k-1}}\right)^{2^k} \prod \sigma(\epsilon_{n-k}u_{n-k} + \cdots + \epsilon_n u_n)^{2^{n-k-1}}} \\ =& \frac{\prod_{\epsilon\in\mathbb{E}_n} \sigma(\epsilon \cdot u)}{\left(\prod_1^{n-k-1} \sigma(u_i)\right)^{2^{n-1}} \left(\prod_{n-k}^{n} \sigma(u_i)\right)^{2^{n-1}}}. \end{aligned}$$

□

Finalement, on obtient le résultat attendu :

Corollaire 4.3.9. *Pour tout $n \geq 2$, $\sigma^{(n)}(X_1, \ldots, X_n) = S_n(X_1, \ldots, X_n)$.*

Remarque 4.3.10. *Il est bien plus rapide de calculer les polynômes de Semaev en tant que résultant qu'en tant que produit des Δ_n.*
Le but de l'introduction de cette nouvelle formule n'est pas d'accélérer le calcul des polynômes de Semaev en genre 1, mais plutôt d'étendre ces polynômes aux genres plus grands.

Remarque 4.3.11. *Désormais, je dénommerai les polynômes de sommation $\sigma^{(n)}$ au lieu de S_n. Les nouveaux polynômes que je vais maintenant construire pour les courbes hyperelliptiques de genre g seront nommés $\sigma^{(n,g)}$. Ainsi, implicitement, la notation $\sigma^{(n)}$ signifiera que je travaille avec $g = 1$.*

4.4 Extension aux courbes hyperelliptiques

La construction de mes polynômes de sommation pour les courbes hyperelliptiques suit le même schéma qu'en genre 1 : je commence par donner les équivalents de théorèmes 4.3.1 et 4.3.2 pour les grands genres. Ces théorèmes proviennent de [39]. Puis je construis les polynômes $\sigma^{(n,g)}$ en multipliant les $\Delta_{n,g}$ qui m'intéressent. Je montre que les fonctions ainsi obtenus vérifient les propriétés voulues, c'est-à-dire que ce sont bien des polynômes en les x_i (indépendants des y_i) qui me donnent des tests pour la recherche de relations dans le calcul d'index.

4.4.1 Enoncé des deux théorèmes pour tout genre

La définition et les propriétés de σ ont été donnés dans le chapitre 1.
Il me faut encore quelques rappels avant de pouvoir donner les énoncés des deux théorèmes dans le cas des courbes hyperelliptiques.

Définition 4.4.1 ([39])**.** Soit $\mathcal{C}$ une courbe hyperelliptique de genre g, $\mathcal{J}$ sa jacobienne et σ sa fonction sigma. On rappelle que σ est par définition une fonction sur $\mathbb{C}^g$. On aura besoin des deux dérivées suivantes de σ :

$$\sigma_\sharp = \left(\prod_{\substack{2\leq i\leq g \\ i\equiv 0 \pmod 2}} \frac{\partial}{\partial u_i} \right) \sigma$$

$$\sigma_\flat = \left(\prod_{\substack{2\leq i\leq g \\ i\equiv 1 \pmod 2}} \frac{\partial}{\partial u_i} \right) \sigma$$

Exemple 4.4.2. *En particulier, en genre* 1, $\sigma_\sharp = \sigma_\flat = \sigma$.
En genre 2, *on a* $\sigma_\sharp = \dfrac{\partial}{\partial u_2}\sigma = \sigma_2$ *et* $\sigma_\flat = \sigma$.

Je rappelle que la difficulté d'utiliser les fonctions σ provient du fait que ce ne sont pas à proprement parler des fonctions sur $\mathcal{J}$: si on veut définir des fonctions sur $\mathcal{J}$, il faudra manipuler des quotients de σ fonctions. Par contre, le lieu d'annulation de ces fonctions sont des sous ensembles de $\mathcal{J}$ bien définis. Il en va de même pour $\sigma_\sharp$ et $\sigma_\flat$.

Lemme 4.4.3 ([39])**.**
- $\{\sigma = 0\} = \Theta$;
- $\{\sigma_\sharp = 0\} \cap \Theta^{[1]} = \{\mathcal{O}\}$;
- $\{\sigma_\flat = 0\} \cap \Theta^{[2]} = \Theta^{[1]}$.

(pour les rappels des définitions de Θ *et* $\Theta^{[i]}$*, je renvois au chapitre 1, définition 1.1.17)*

Ainsi munis de ces deux nouveaux outils, je reproduis ici les énoncés des théorèmes d'addition et de Frobenius-Stickelberger donnés par Onishi dans [39] :

Théorème 4.4.4 (Loi d'addition)**.** *[39]*
Soit $u_1, u_2 \in \Theta^{[1]}$. *On les écrit* $u_i \sim (P_i) - (\infty)$ *pour certains points* $P_1, P_2 \in \mathcal{C}$.
Avec $x_i = x(P_i)$, *on a*

$$(-1)^{g+1} \frac{\sigma_\flat(u_1 - u_2)\sigma_\flat(u_1 + u_2)}{\sigma_\sharp^2(u_1)\sigma_\sharp^2(u_2)} = \begin{vmatrix} 1 & x_1 \\ 1 & x_2 \end{vmatrix}$$

Théorème 4.4.5 (Frobenius-Stickelberger). *[39]*
Pour $n \geq g$ un entier, soient $u_1, \ldots, u_n \in \Theta^{[1]}$, avec $u_i \sim (P_i) - (\infty)$ et $P_i = (x_i, y_i)$. On a

$$c_n \frac{\sigma(u_1 + \cdots + u_n) \prod_{i<j} \sigma_\flat(u_i - u_j)}{\left(\prod_i \sigma_\sharp(u_i)\right)^n} = \begin{vmatrix} 1 & x_1 & x_1^2 & \ldots & x_1^g & y_1 & x_1^{g+1} & x_1 y_1 & x_1^{g+2} & \ldots \\ \vdots & \vdots & & & \vdots & & & \vdots & \vdots & \\ 1 & x_n & x_n^2 & \ldots & x_n^g & y_n & x_n^{g+1} & x_n y_n & x_n^{g+2} & \ldots \end{vmatrix} = \Delta_{n,g},$$

avec la matrice de taille $n \times n$, et c_n des signes donnés dans la table suivante :

		n *mod* 4			
		1	2	3	0
g *mod* 4	1	1	1	-1	-1
	2	-1	-1	-1	-1
	3	-1	1	1	-1
	0	-1	1	-1	1

Plus précisément, la i-ième ligne de la matrice est l'évaluation des fonctions

$$1, x, \ldots, x^a, y, yx, \ldots, yx^b$$

sur les points (P_i), avec $a = \lfloor \frac{n+g-1}{2} \rfloor$ et $b = \lfloor \frac{n-g-2}{2} \rfloor$.

Remarque 4.4.6.

- *Si je prends $g = 1$ dans ces deux théorèmes, je retrouve les résultats énoncés dans 4.3.1 et 4.3.2 .*
- *Comme précédemment, $\Delta_{n,g}$ contrôle quand on a une relation*

$$(P_1) + \cdots + (P_n) - n(\infty) \in \Theta.$$

- *En pratique, dans 4.4.5, la plus petite valeur de n utile est $n = g + 2$. En effet, si on a $\Delta_{n,g}(P_1, \ldots, P_n) = 0,$, alors on peut écrire*

$$(P_1) + \cdots + (P_n) - n(\infty) \sim (Q_1) + \cdots + (Q_k) - (k)(\infty)$$

pour certains points Q_i et $k \leq g - 1$.
Mais si $n = g$ ou $g + 1$, on obtient alors deux diviseurs réduits équivalents, donc égaux. En particulier, les seules solutions sont triviales : $k = n$ et $\{P_i\} = \{Q_i\}$.

- *Comme pour le cas $g = 1$, je peux expliquer ce théorème :*

$$\begin{aligned}\sigma(u_1 + \cdots + u_n) = 0 \Leftrightarrow & (u_1) + \cdots + (u_n) - n(\infty) \in \Theta \\ \Leftrightarrow & \text{ il existe des points } u_{n+1}, \ldots, u_{n+g-1} \text{ t.q.} \\ & (u_1) + \cdots + (u_n) + \\ & (u_{n+1}) + \cdots + (u_{n+g-1}) - (n+g-1)(\infty) \sim \mathcal{O} \\ \Leftrightarrow & \exists f \in \mathcal{L}((n+g-1)\infty) \text{ t.q. } \forall 1 \leq i \leq n,\ f(u_i) = 0.\end{aligned}$$

Et justement, $\{1, x, \ldots, x^a, y, yx, \ldots, yx^b\}$ est une base de cet espace de Riemann Roch $\mathcal{L}((n+g-1)\infty)$.

4.4.2 Polynômes de sommation hyperelliptiques

J'utilise maintenant les deux théorèmes 4.4.4 et 4.4.5 pour construire les polynômes de sommation hyperelliptiques.

Théorème 4.4.7.

- *Pour $n \geq g$ et $u_1, \ldots, u_n \in \Theta^{[1]}$,*

$$\sigma^{(n,g)}(u_1, \ldots, u_n) = \frac{\prod_{\epsilon \in \mathbb{E}_n} \sigma(\epsilon \cdot u)}{\left(\prod_{i=1}^n \sigma_\sharp(u_i)\right)^{2^{n-1}}} = \frac{\prod_\epsilon \Delta_{n,g}(\epsilon_1 u_1, \ldots, \epsilon_n u_n)}{\left(\prod_{i<j} \Delta_2(u_i, u_j)\right)^{2^{n-2}}}$$

est une fonction sur $\left(\Theta^{[1]}\right)^n$.

- *On a les propriétés suivantes de $\sigma^{(n,g)}$:*
  - *$\sigma^{(n,g)} \in \mathbb{K}[x_1, \ldots, x_n]$;*
  - *$\sigma^{(n,g)}$ vérifie*

$$\sigma^{(n,g)}(u_1, \ldots, u_n) = 0 \Leftrightarrow \exists \epsilon \in \mathbb{E}_n \text{ t.q. } \epsilon_1 u_1 + \cdots + \epsilon_n u_n \in \Theta;$$

  - *$\sigma^{(n,g)}$ est de degré $2^{n-2}g$ en chaque x_i.*

Démonstration. Ce théorème étant l'extension du travail effectué dans la section 4.3.2, on va suivre le même schéma de réflexion pour le démontrer.

Commençons par démontrer l'égalité énoncée dans le premier point. Comme on l'a fait en genre 1, cela se fait en multipliant les différentes relations de Frobenius-Stickelberger obtenues pour chaque choix de $\epsilon \in \mathbb{E}_n$.
Du côté droit de l'égalité, on a le produit de tous les déterminants $\Delta_{n,g}(\epsilon_1 u_1, \ldots, \epsilon_n u_n)$.
Du côté gauche, on a :

$$\frac{\left(\prod_\epsilon \sigma(\epsilon \cdot u)\right)\left(\prod_\epsilon \prod_{i<j} \sigma_\flat(\epsilon_i u_i - \epsilon_j u_j)\right)}{\left(\prod_i \sigma_\sharp(u_i)^n\right)^{2^{n-1}}}.$$

Selon la valeur de g modulo 4, σ est une fonction paire ou impaire. Quoiqu'il en soit, on a toujours

$$\prod_{\epsilon}\prod_{i<j}\sigma_\flat(\epsilon_i u_i-\epsilon_j u_j)=\prod_{i<j}\left(\sigma_\flat(u_i-u_j)\sigma_\flat(u_i+u_j)\right)^{2^{n-2}}.$$

Finalement, il ne reste qu'à écrire que

$$\begin{aligned}\left(\prod_{i<j}\Delta_2(u_i,u_j)\right)^{2^{n-2}}&=\prod_{i<j}\left(\frac{\sigma_\flat(u_i+u_j)\sigma_\flat(u_i-u_j)}{\sigma_\sharp(u_i)^2\sigma_\sharp(u_j)^2}\right)^{2^{n-2}}\\&=\frac{\prod_{i<j}\left(\sigma_\flat(u_i-u_j)\sigma_\flat(u_i+u_j)\right)^{2^{n-2}}}{\prod_i\sigma_\sharp(u_i)^{2^{n-1}(n-1)}}.\end{aligned}$$

Il faut ensuite montrer que cette quantité est un polynôme en les x_i. Dans le cas elliptique, on avait démontré ce résultat grâce au lemme de réduction 4.3.4. Il faudra refaire ici le même travail.

On considère donc le produit suivant de deux déterminants :

$$f(u_1,\dots,u_n)=\begin{vmatrix}1&x_1&y_1&x_1^2&y_1x_1&x_1^3&\dots\\1&x_2&y_2&x_2^2&y_2x_2&x_2^3&\dots\\&&&\vdots&&&\\1&x_n&y_n&x_n^2&y_nx_n&x_n^3&\dots\end{vmatrix}\cdot\begin{vmatrix}1&x_1&y_1&x_1^2&y_1x_1&x_1^3&\dots\\1&x_2&-y_2&x_2^2&-y_2x_2&x_2^3&\dots\\&&&\vdots&&&\\1&x_n&y_n&x_n^2&y_nx_n&x_n^3&\dots\end{vmatrix}.$$

C'est à priori un polynôme en les x_i et y_i, et comme énoncé dans le lemme 4.3.4, c'est une fonction paire en y_2 : on peut donc utiliser l'équation de Weierstrass de la courbe pour éliminer y_2 dans l'expression de f. On continue de suivre le raisonnement du lemme : si pour un indice $i\neq 2$ on a $x_i=\pm x_2$, alors $f(x_1,\dots,x_n)=0$, ce qui nous permet de voir que f est un polynôme divisible par

$$\prod_{i\neq 2}(x_i-x_2)=\prod_{i\neq 2}\Delta_2(u_2,u_i).$$

On revient à la quantité

$$\sigma^{(n,g)}(u_1,\dots,u_n)=\frac{\prod_{\epsilon\in\mathbb{E}_n}\sigma(\epsilon\cdot u)}{\left(\prod_{i=1}^n\sigma_\sharp(u_i)\right)^{2^{n-1}}}=\frac{\prod_\epsilon\Delta_{n,g}(\epsilon_1u_1,\dots,\epsilon_nu_n)}{\left(\prod_{i<j}\Delta_2(u_i,u_j)\right)^{2^{n-2}}}$$

étudiée dans ce théorème : muni du lemme de réduction, on voit que c'est bien un polynôme en $x_1,\dots,x_n$.

Le fait que $\sigma^{(n,g)}$ permette effectivement de détecter les relations désirées :

$$\sigma^{(n,g)}(u_1,\dots,u_n)=0\Leftrightarrow\exists\epsilon\in\mathbb{E}_n\text{ t.q. }\epsilon_1u_1+\dots+\epsilon_nu_n\in\Theta$$

provient directement des propriétés de la fonction σ.

Enfin, il ne reste plus qu'à démontrer le degré. Cela se fait exactement comme dans la proposition 4.3.8.

□

Remarque 4.4.8. *Comme pour le genre* 1, σ *est défini sur* $\mathbb{C}^g$, *mais les fonctions* $\sigma^{(n,g)}$ *sont des éléments de* $\mathbb{K}[x_1, \ldots, x_n]$ *si* $\mathcal{C}$ *est définie sur le corps* $\mathbb{K}$.

D'un point de vue calculatoire, j'ai déjà relevé le fait que la formule du résultant donnée par Semaev est plus efficace que ma formule de $\sigma^{(n)}$ comme produit de fonctions Δ_n. Malheureusement, la formule du résultant est fausse pour $g > 1$. En effet, je rappelle que la preuve de cette formule est le point (v) de la proposition 4.3.8 : j'utilisais la formule

$$\begin{aligned}&\mathrm{Res}_X(\sigma^{(n-k)}(X_1, \ldots, X_{n-k-1}, X), \sigma^{(k+2)}(X_{n-k}, \ldots, X_n, X))\\ =&a^m \prod_i \sigma^{(n-k)}(X_1, \ldots, X_{n-k-1}, \alpha_i),\end{aligned}$$

où

- $a = lc_X(\sigma^{(k+2)}(X_{n-k}, \ldots, X_n, X)) = \sigma^{(k+1)}(X_{n-k}, \ldots, X_n)$;
- les α_i sont ses racines ;
- $m = deg_X(\sigma^{(n-k)}(X_1, \ldots, X_{n-k-1}, X)) = 2^{n-k-2}$;

et la définition de $\sigma^{(n)}$:

$$\sigma^{(n)}(u_1, \ldots, u_n) = \frac{\prod_{\epsilon \in \mathbb{E}_n} \sigma(\epsilon \cdot u)}{\left(\prod_{i=1}^n \sigma_\flat(u_i)\right)^{2^{n-1}}}.$$

Maintenant, si j'essaye d'imiter cette preuve pour $g > 1$, je fais face à trois problèmes.
Tout d'abord, je n'ai pas de jolie description des α_i comme pour le genre 1 (point (iv) de 4.3.8) ; ainsi, les racines de $X \mapsto \sigma^{(n,g)}(X_1, \ldots, X_{n-1}, X)$ sont maintenant :

$$\{x \,(P_1, \epsilon_2 P_2, \ldots, \epsilon_n P_n + Q_1 + \cdots + Q_{g-1}) \,| Q_i \in \mathcal{C}\};$$

dans le cas elliptique, le miracle $g - 1 = 0$ permettait d'avoir au numérateur de

$$\prod_i \sigma^{(n-k)}(X_1, \ldots, X_{n-k-1}, \alpha_i) = \prod_i \frac{\prod_\epsilon \sigma(P_1 + \epsilon_2 P_2 + \cdots + \epsilon_{n-k-1} P_{n-k-1} + \epsilon_n \alpha_i)}{\left(\prod_1^{n-k-1} \sigma(P_j)\right) \sigma(\alpha_i)}$$

la quantité

$$\prod_\epsilon \sigma(\epsilon \cdot P).$$

Le second problème est que le degré de $\sigma^{(n,g)}$ n'est plus 2^{n-2}, mais $2^{n-2}g$.
Enfin, troisième problème : si $g > 2$, $\sigma_\flat \neq \sigma$.

4.4.3 Utilisation pour le calcul d'index

Je rappelle que la base de factorisation ici est

$$\mathcal{F} = \{u = (P) - (\infty) \in \Theta^{[1]} | x(P) \in \mathbb{F}_q\}.$$

On a un point $R = aD_1 + bD_2$, pour des entiers a et b choisis aléatoirement, et on veut décider de la décomposition de R dans $\mathcal{F}$.

Soient $P_1, \ldots, P_g$ les points intervenant dans la représentation réduite de R, avec $P_i = (x_i, y_i)$. On considère le $n+g$-ième polynôme de sommation $\sigma^{(n+g,g)}$, et on veut résoudre le système

$$\begin{cases} \sigma^{(n+g,g)}(x_1, \ldots, x_g, X_{g+1}, \ldots, X_{g+n}) = 0, \\ X_{g+1}, \ldots, X_{g+n} \in \mathbb{F}_q. \end{cases}$$

Ici, les x_i sont donnés par l'énoncé et mes inconnues sont les n variables X_i. On utilise alors une base de $\mathbb{F}_{q^n}/\mathbb{F}_q$ pour obtenir un système de n polynômes sur $\mathbb{F}_q$ à n variables. On le résout par un calcul de base de Gröbner.

Comme Gaudry l'a remarqué pour $g = 1$, les nouveaux polynômes $\sigma^{(n+g,g)}$ sont symétriques. On peut donc les exprimer avec les polynômes symétriques élémentaires $e_1, \ldots, e_n$ en $X_{g+1}, \ldots, X_{g+n}$: on obtient ainsi des polynômes de degré **total** $2^{n+g-2}g$.

Malheureusement, et contrairement à la situation rencontrée pour $g = 1$, résoudre ce système n'est pas suffisant pour trouver une décomposition de R. En effet, trouver une solution $(X_{g+1}, \ldots, X_{g+n})$ signifie que l'on a des points $P_{g+1}, \ldots, P_{g+n} \in \mathcal{F}$ ($P_i = (X_i, Y_i)$ pour un certain Y_i) tel qu'il existe certains points (inconnus) $Q_1, \ldots, Q_{g-1}$ avec :

$$P_1 \pm P_2 \pm \cdots \pm P_{g+n} \sim Q_1 + \cdots + Q_{g-1}.$$

A ce point, il reste à traiter les problèmes suivants :

(i) Les relations que l'on veut en réalité sont

$$P_1 + \cdots + P_g \pm P_{g+1} \cdots \pm P_{g+n} \in \Theta,$$

i.e. les $g-1$ premiers signes sont imposés comme étant positifs. Cela arrive avec une probabilité 2^{1-g}.

(ii) Même si les signes sont bons, il reste à calculer ces points Q_i : on les obtient en calculant tous les diviseurs

$$P_1 + \cdots + P_g \pm P_{g+1} \cdots \pm P_{g+n}.$$

(iii) Enfin, il faut vérifier si, pour l'un des choix de signes effectués en (ii), l'on a bien $x(Q_i) \in \mathbb{F}_q$ pour tout i.

J'illustre maintenant la recherche de relations en genre 2.

4.4.4 Un exemple en genre 2

Je rappelle qu'en genre 2, $\Theta = \{(P) - (\infty)\} = \Theta^{[1]}$, c'est-à-dire $\mathcal{L} = \Theta^{[1]} \subset \mathcal{F}$. La situation est la suivante : on a un diviseur $R = (P_1) + (P_2) - 2(\infty)$ dont on veut tester la friabilité.

Dans cet exemple, on travaille sur $\mathbb{F}_{q^2}$ avec $q = 1048583$ et $\mathbb{F}_{q^2} \simeq \mathbb{F}_q(a)$, a vérifiant $a^2 + a + 1 = 0$.
La courbe choisie est définie par $y^2 = x^5 + (956 + 16a)x + 560$.
Comme l'extension est de degré 2, on doit utiliser le $g + 2 = 4$-ième polynôme de sommation $\sigma^{(4,2)}$. Si le point que l'on souhaite décomposer est $R \sim (P_1) + (P_2) - 2(\infty)$, avec

$$P_1 = (822466a+1019211, 208059a+779837) \text{ et } P_2 = (964315a+809425, 207076a+760154),$$

dans $\sigma^{(4,2)}(x_1, x_2, x_3, x_4)$, on doit remplacer les inconnues x_1 et x_2 par les valeurs connues $x(P_1)$ et $x(P_2)$ respectivement. On obtient alors un polynôme symétrique en x_3 et x_4. Si on l'exprime en fonction des polynômes symétriques élémentaires $e_1 = x_3 + x_4$ et $e_2 = x_3x_4$, on obtient quelque chose qui commence comme

$$e_1^8 - (86278a + 469901)e_1^7e_2 + (413036a + 304842)e_1^7 + \cdots = 0.$$

Remarquons que le degré total de ce polynôme est bien $2^{n-2}g = 8$.
Comme on veut des solutions dans $\mathbb{F}_q$, cette équation est équivalente au système

$$\begin{cases} e_1^8 - 469901e_1^7e_2 + \cdots = 0, \\ -86278e_1^7e_2 + \cdots = 0. \end{cases}$$

Après calcul de la base de Gröbner de ce système, on obtient les quatre couples de solutions

$$\{(48993, 5391), (-890, 86741), (-72562, -313), (145643, 17928)\},$$

et finalement, les solutions dans $\mathbb{F}_q$ sont

$$(x_3, x_4) \in \{(97763, 999813), (47880, 999813), (97763, 47880), (640319, 335702)\}.$$

On calcule les points P_3 et P_4 correspondants, et on voit que les trois premiers couples nous donnent effectivement une décomposition de R :

$$R \sim (47880, 467662a+247545)+(999813, 350820a+258681)+(97763, 7563a-232022),$$

alors que le dernier couple nous donne une décomposition de $-(P_1)+(P_2)-2(\infty)$, ce dont on n'a pas besoin.

4.4.5 Complexité

On retourne au cas g quelconque pour analyser le coût de la recherche de relations avec ces polynômes de sommation hyperelliptiques.

On considère donc une courbe $\mathcal{C}$ de genre g définie sur un corps fini $\mathbb{F}_{q^n}$, et on veut décomposer un élément

$$R \sim (P_1) + \cdots + (P_g) - g(\infty) \in \mathcal{J}$$

en somme de n points de $\mathcal{F}$. On doit ainsi utiliser le polynôme $\sigma^{(n+g,g)} \in \mathbb{F}_{q^n}[X_1, \ldots, X_{g+n}]$. Comme expliqué dans l'exemple précédent en genre 2, on remplace les g premières variables de $\sigma^{(n+g,g)}$ par les $x_i = x(P_i)$ fournis par R, et on obtient alors, après symétrisation, un système de n équations polynomiales sur $\mathbb{F}_q$ en n variables de degré total $2^{n+g-2}g$.

Dans sa thèse ([52], section 7.2.2), Vitse étudie le coût d'un test de décomposition d'un point R en utilisant les polynômes de sommation elliptiques. En suivant son analyse, j'obtiens le résultat suivant :

Proposition 4.4.9. *En notant $\mathfrak{c}(g, n, q)$ le coût d'un test de décomposition d'un point $R \in \mathcal{J}$ de la jacobienne d'une courbe de genre g en la somme de n points de la base de factorisation $\mathcal{F}$, on a*

$$\mathfrak{c}(g, n, q) = \tilde{O}(2^{3n(n+g-2)} g^{3n} n)$$

quand $q \to \infty$.

Démonstration. En fait, $\mathfrak{c}(g, n, q)$ correspond au coût de la résolution du système polynomial sur $\mathbb{F}_q$ obtenu à partir de la relation

$$\sigma^{(n+g,g)}(x_1, \ldots, x_g, X_{g+1}, \ldots, X_{g+n}) = 0.$$

Cette résolution se fait par un calcul de bases de Gröbner. La stratégie usuelle est en deux temps : d'abord utiliser l'algorithme $F4$ de Faugère pour calculer une base pour l'ordre degrevlex, puis appliquer l'algorithme de changement d'ordre FGLM pour en déduire une base pour l'ordre lexicographique. Ces algorithmes ont été proposés respectivement dans [15] et [16]. Pour estimer $\mathfrak{c}(g, n, q)$, il faut regarder la complexité de ces deux étapes.

Sans rentrer dans les détails, la complexité du calcul d'une base de Gröbner d'un idéal $I = \langle f_1, \ldots, f_N \rangle$ de dimension zéro d'un anneau de polynômes $\mathbb{K}[X_1, \ldots, X_N]$ avec l'algorithme $F4$ est majorée par

$$\tilde{O}\left(\binom{N + d_{reg}}{N}^{\omega}\right),$$

où le degré de régularité d_{reg} est majoré par la borne de Macaulay

$$d_{reg} \leq \sum_i (\deg(f_i) - 1) + 1,$$

et ω est l'exposant intervenant dans la complexité du produit matriciel.
Ici, on a $N = n$ et la borne de Macaulay donne

$$d_{reg} \leq n2^{n+g-2}g - n + 1.$$

Après utilisation de la formule de Stirling, on trouve finalement une complexité en

$$\tilde{O}\left(\left(2^{n(n+g-2)}g^n e^n n^{-1/2}\right)^{\omega}\right).$$

On passe maintenant à la complexité de l'algorithme de changement d'ordre $FGLM$. Pour un idéal de degré D de polynômes à N variables, elle est de $O(ND^3)$. Le degré D peut s'estimer par la borne de Bézout : dans notre cas, l'idéal est généré par n polynômes, tous de même degré $2^{n+g-2}g$, ce qui nous donne la majoration suivante

$$D \leq 2^{n(n+g-2)}g^n.$$

La coût de cette étape est donc

$$O(2^{3n(n+g-2)}g^{3n}n).$$

Comme le coût de la seconde étape domine celui de la première, c'est lui qui donne l'estimation de $\mathfrak{c}(g, n, q)$. □

Maintenant qu'on a obtenu le coût $\mathfrak{c}(g, n, q)$ d'une utilisation des polynômes hyperelliptiques de sommation, il s'agit d'estimer le coût global de la méthode de calcul d'indice. Pour cela, il est nécessaire d'estimer :

- le cardinal de la base de factorisation (puisque le but est d'obtenir plus de relations qu'il n'y a d'éléments dans cette base) ;
- la probabilité qu'un diviseur choisi aléatoirement se décompose dans la base.

Pour estimer ces quantités, nous continuons de suivre la thèse de Vitse ([52], section 7.2.3) : la base de factorisation contient environ $q/2$ points, quand q est

suffisamment grand, tandis qu'une analyse heuristique donne une probabilité de décomposition de l'ordre de :

$$\frac{\sharp \mathcal{C}^{ng} / \mathfrak{S}_{ng}}{\sharp \mathcal{F}} = \frac{1}{(ng)!}.$$

Il faut maintenant faire attention au fait que les polynômes de sommation hyperelliptiques ne testent pas la nullité d'une somme, mais seulement sa présence dans le diviseur Θ.
Pour plus de clarté, reprenons notre exemple de la section 4.4.3 : partant d'un diviseur $R = P_1 + \cdots + P_g$ que l'on veut décomposer, l'utilisation de $\sigma^{(n+g,g)}$ permet d'obtenir les abscisses $x_{g+1}, \ldots, x_{g+n}$ de n points $P_{g+1}, \ldots, P_{g+n}$ nous donnant la relation

$$P_1 + \cdots + P_{g+n} \in \Theta,$$

c'est-à-dire

$$P_1 + \cdots + P_{g+n} \sim Q_1 + \cdots + Q_{g-1},$$

pour certains points Q_i inconnus de la courbe $\mathcal{C}$.
Cela aboutit à une relation à la double condition que tous les x_i soient dans $\mathbb{F}_q$, puis que tous les $x(Q_i)$ soient également dans $\mathbb{F}_q$. Selon l'analyse heuristique citée plus haut, la première condition est vérifiée avec une probabilité $1/(g+n)!$.
Quant à la seconde, les $g-1$ quantités $x(Q_i)$ sont à priori dans $\mathbb{F}_{q^n}$, et on leur demande à être dans $\mathbb{F}_q$: ceci arrive avec une probabilité en $q^{(1-n)(1-g)}$.

Ainsi, pour obtenir de l'ordre de $q/2$ relations, il faudra résoudre en moyenne $(g+n)!(g-1)!q/2$ systèmes polynomiaux provenant des polynômes de sommation, et, finalement, le coût global de la méthode est de l'ordre de

$$\tilde{O}\left(q^{(1+(n-1)(g-1))} 2^{3n(n+g-2)} g^{3n} n(g+n)!\right)$$

quand $q \to \infty$.

4.5 Constructions alternatives

Faisons le point : en réécrivant la théorie de Semaev, j'ai pu étendre ses polynômes de sommation aux courbes hyperelliptiques. Ces nouveaux polynômes méritent effectivement le nom de polynômes de sommation hyperelliptiques, en référence aux polynômes de sommation de Semaev, car :

- comme dans le genre 1, ils permettent de trouver des relations pour l'algorithme de calcul d'index ;
- comme dans le genre 1, ils ne dépendent que de la coordonnée x des points de la courbe ;

— si on remplace g par 1 dans mes $\sigma^{(n,g)}$, on retrouve bien les polynômes de Semaev.

Dans cette section, je m'intéresse maintenant à l'étude de polynômes de sommation alternatifs. En particulier, j'expliquerai pourquoi ces polynômes sont moins performants que ceux que j'ai précédemment exposés.

Dans un premier temps, je m'attaque à deux défauts de mes polynômes : ils sont chers à calculer, car ils nécessitent la multiplication de 2^{n-1} déterminants de taille n ; et ils ne détectent une relation qu'avec une probabilité 2^{1-g}. En multipliant moins de déterminants, pourrait on obtenir des polynômes de sommation plus efficaces ?

Dans un second temps, je donne mes résultats sur une question soulevée par Guénaël Renault lors d'un exposé à Rennes que je réécris de la façon suivante : pour tester des égalités de type

$$R + P_1 + \cdots + P_n = \mathcal{O}$$

sur une courbe $\mathcal{E}$, peut on utiliser une coordonnée c et un automorphisme Ψ de $\mathcal{E}$ tels que

$$\pi_c(\mathcal{E}) \sim \mathcal{E}/\Psi$$

pour obtenir des polynômes de sommation dépendant uniquement des $c(P_i)$?
Ainsi, dans le cas des polynômes de Semaev, l'automorphisme Ψ est simplement $[-1] : (x, y) \mapsto (x, -y)$ et ma coordonnée c est x.

4.5.1 Peut-on utiliser moins de fonctions $\Delta_{n,g}$?

Considérons par exemple le polynôme $\sigma^{(3,2)}$. Pour le construire, on doit multiplier au numérateur

$$\Delta_{3,2}(P_1, P_2, P_3) \times \Delta_{3,2}(P_1, -P_2, P_3) \times \Delta_{3,2}(P_1, P_2, -P_3) \times \Delta_{3,2}(P_1, -P_2, -P_3).$$

Mais, parmi ces déterminants, seulement la moitié nous sont utiles :

$$\Delta_{3,2}(P_1, P_2, P_3) \times \Delta_{3,2}(P_1, P_2, -P_3) = 0 \Leftrightarrow P_1 + P_2 \pm P_3 \in \Theta,$$

tandis que les deux autres nous donnent de mauvaises relations :

$$\Delta_{3,2}(P_1, -P_2, P_3) \times \Delta_{3,2}(P_1, -P_2, -P_3) = 0 \Leftrightarrow P_1 - P_2 \pm P_3 \in \Theta.$$

Remarque 4.5.1.

— *Cet exemple est un peu artificiel : j'ai déjà expliqué que pour $g = 2$, le premier polynôme de sommation intéressant était $\sigma^{(4,2)}$. Pour des n plus grands, ce sont exactement les mêmes idées qui entrent en jeu. La seule différence est que de tels exemples sont plus longs à écrire.*
— *Pour $g = 1$, il ne se passe rien du tout : les nouveaux polynômes que je vais construire $\tilde{\sigma}^{(n,g)}$ sont alors égaux à $\sigma^{(n,g)}$.*

Je rappelle que pour $g = 1$, on avait démontré le lemme de réduction 4.3.4. Pour le cas général, ce lemme est démontré dans la preuve du théorème 4.4.7. En utilisant ce lemme, j'obtiens la construction suivante de polynômes $\tilde{\sigma}^{(n,g)}$ alternatifs :

Théorème 4.5.2. *Soit $n \in \mathbb{N}$, et soient $P_1, \dots, P_{g+n} \in \mathcal{C}$. On veut décomposer le diviseur $R = (P_1) + \cdots + (P_g) - g(\infty)$. On doit considérer séparément les cas $n = 2$ et $n > 2$:*

- *si $n = 2$, on considère la fonction $\tilde{\sigma}^{(g+2,g)}$ construite ainsi :*

$$\frac{\Delta_{g+2,g}(R, P_{g+1}, P_{g+2})\Delta_{g+2,g}(R, P_{g+1}, -P_{g+2})\Delta_{g+2,g}(R, -P_{g+1}, P_{g+2})\Delta_{g+2,g}(R, -P_{g+1}, -P_{g+2}}{\left(\Delta_2(P_{g+1}, P_{g+2}) \prod_{\substack{i \neq g+1 \\ i \neq g+2}} \Delta_2(P_{g+1}, P_i)\Delta_2(P_{g+2}, P_i)\right)^2}$$

  C'est un polynôme en x_{g+1} et x_{g+2}. Il est symétrique et est de degré $2g$ en chaque variable.
- *si $n > 2$, la fonction est*

$$\tilde{\sigma}^{(n+g)}(R, P_{g+1}, \dots, P_{g+n}) = \frac{\prod_{\epsilon \in \{\pm 1\}^n} \Delta_{n+g,g}(P_1, \dots, P_g, \epsilon_1 P_{g+1} + \cdots + \epsilon_n P_{g+n})}{\left(\prod_{i \neq g+1} \Delta_2(P_{g+1}, P_i)\right)^{2^{n-1}}}.$$

  C'est un polynôme en les x_i $(g + 1 \leq i \leq g + n)$.
  Il n'est pas symétrique.
  Il est de degré $2^{n-1}(2g + n - 2)$ en chaque x_i pour $i \leq g + 2$, et de degré $2^{n-1}g$ en x_{g+1}.

Démonstration. A priori, les deux quantités évoquées dans ce théorème sont des fonctions rationnelles en les x_i et les y_i. Il faut d'abord voir que ce sont en fait des fonctions rationnelles ne dépendant que des x_i (c'est le second point du lemme), puis que ce sont en fait des polynômes (c'est le troisième point du lemme).

□

Remarque 4.5.3. *On ne peut pas utiliser moins de déterminants que les 2^n impliqués dans la construction de $\tilde{\sigma}^{(g+n,g)}$. En effet, sans eux, on ne peut plus utiliser correctement le lemme de réduction 4.3.4 afin de se débarrasser des variables y_i.*

Ainsi, sauf pour $n = 2$, le défaut de symétrisation des nouveaux $\tilde{\sigma}^{(g+n,g)}$ les rend inutilisables. En effet, après symétrisation, les polynômes $\sigma^{(g+n,g)}$ sont de degré total $2^{n+g-2}g$, contre $2^{n-1}((2n-1)g + (n-2)(n-1))$ pour les $\tilde{\sigma}^{(g+n,g)}$.

Pour $n = 2$ par contre, $\tilde{\sigma}^{(g+2,g)}$ est plus efficace que $\sigma^{(g+2,g)}$: en effet, ils sont tous les deux symétriques, mais le premier est de degré $2g$ alors que le second est de degré $2^g g$.

4.5.2 Cas elliptique : peut on utiliser un autre automorphisme que $[-1]$?

Comme énoncé dans l'introduction de cette section, l'idée est d'utiliser une coordonnée c et un automorphisme ξ de $\mathcal{E}$ tels que

$$\pi_c(\mathcal{E}) \simeq \mathcal{E}/\xi.$$

À partir de c et ξ, je veux obtenir des polynômes de sommation $\hat{\sigma}^{(n)}$ alternatifs :

Définition 4.5.4. J'appelle *polynômes de sommation relatifs à c* des polynômes $\hat{\sigma}^{(n)} \in \mathbb{K}[C_1, \ldots, C_n]$ tels que

$$\begin{aligned}\hat{\sigma}^{(n)}(c_1, \ldots, c_n) = 0 \Leftrightarrow \exists\ & P_i \in \mathcal{E} \text{ tq } c(P_i) = c_i, \\ & \exists\ i_2, \ldots, i_n \in \mathbb{N}, \\ & P_1 + [\xi^{i_2}]P_2 + \cdots + [\xi^{i_n}]P_n = \mathcal{O}.\end{aligned}$$

Dans le cadre du calcul d'index, la base de factorisation à considérer devient naturellement

$$\{P \in \mathcal{E}(\mathbb{F}_{q^k}) \mid c(P) \in \mathbb{F}_q\}.$$

Remarque 4.5.5. *Cette section est basée sur les résultats de [14]. Plus précisément, dans cet article, les auteurs ont construit les polynômes que j'appelle $\hat{\sigma}^{(2)}$ et $\hat{\sigma}^{(3)}$ de mes théorèmes 4.5.7, 4.5.11, 4.5.12 et 4.5.4 (il s'agit des propositions 5.1, 5.4 et 6.2 de [14]). La construction de $\hat{\sigma}^{(n)}$ pour tout n n'était pas le but de leurs travaux, et fait donc l'objet d'un travail original de ma part.*

4.5.2.1 Quels automorphismes ?

Je me place en caractéristique $\neq 2, 3$. Je rappelle (voir [42] par exemple) que

$$\mathrm{Aut}(\mathcal{E}) \simeq \begin{cases} \mu_2 \text{ si } j \neq 0, 1728 \\ \mu_4 \text{ si } j = 1728 \\ \mu_6 \text{ si } j = 0, \end{cases}$$

avec $[\xi](x, y) = (\xi^2 x, \xi^3 y)$.

Ainsi, dans le cas $j \neq 0, 1728$, nous n'avons rien de plus que l'automorphisme $[-1]$ utilisé par Semaev, avec la coordonnée $c = x$.

4.5.2.2 Le cas $j = 1728$

Le modèle de Weierstrass de $\mathcal{E}$ est $y^2 = x^3 + Ax$.
Si on fixe i racine carrée de -1, les deux nouveaux automorphismes qu'on obtient sont $(x, y) \mapsto (-x, \pm iy)$. Autrement dit, on a $c = x^2$.

On considère le polynôme suivant :

$$f(x_1, \ldots, x_n) = \prod \sigma^{(n)}(x_1, \pm x_2, \ldots, \pm x_n).$$

Ce polynôme vérifie presque la définition 4.5.4 : $f(x_1, \ldots, x_n) = 0$ signifie qu'il existe des points $P_i = (x_i, y_i) \in \mathcal{E}$ tels que

$$P_1 \pm [\xi]^{i_2} P_2 \pm \cdots \pm [\xi]^{i_n} P_n = \mathcal{O},$$

où $[\xi](x, y) = (-x, iy)$.

Remarque 4.5.6. *La racine carrée de -1, i, ne devra pas être confondue avec les entiers $0 \leq i_k \leq 3$.*

Il ne manque donc plus qu'une chose à montrer pour que f rentre bien dans le cadre de la définition : le fait que f soit un élément de $\mathbb{K}[x_1^2, \ldots, x_n^2]$, c'est-à-dire que, comme fonction de $x_1, \ldots, x_n$, f soit paire. Cela se fait en trois étapes :

- d'abord on montre que f est paire en tant que fonction en x_2 ;
- en remarquant que f est symétrique en $x_2, \ldots, x_n$, on obtient alors gratuitement la parité en $x_3, \ldots, x_n$;
- enfin on montre séparément la parité en x_1.

Pour la parité en x_2, il n'y qu'à écrire :

$$\begin{aligned} f(x_1, -x_2, \ldots, x_n) &= \prod_{\epsilon} \sigma^{(n)}(x_1, -\epsilon_2 x_2, \epsilon_3 x_3 \ldots, \epsilon_n x_n) \\ &= \prod_{\epsilon'} \sigma^{(n)}(x_1, \epsilon'_2 x_2, \epsilon'_3 x_3 \ldots, \epsilon'_n x_n) \\ &= f(x_1, x_2, \ldots, x_n), \end{aligned}$$

où $\epsilon = (\epsilon_2, \ldots, \epsilon_n)$ et $\epsilon' = (-\epsilon_2, \ldots, \epsilon_n)$.

Pour x_1, on va devoir développer l'expression de f :

$$\begin{aligned} f(x_1, x_2, \ldots, x_n) &= \prod \sigma^{(n)}(x_1, \pm x_2, \ldots, \pm x_n) \\ &= \prod \sigma^{(n)}(P_1, [\xi]^{i_2} P_2, \ldots, [\xi]^{i_n} P_n) \\ &= \prod_{i_k \in \{0,1\}} \left(\frac{\prod_{\epsilon_k \in \{\pm 1\}} \Delta_n(P_1, \epsilon_2 [\xi]^{i_2} P_2, \ldots, \epsilon_n [\xi]^{i_n} P_n)}{\prod_{j>1} \left(\Delta_2(P_1, [\xi]^{i_j} P_j)\right)^{4^{n-2}} \prod_{2 \le k < l} \left(\Delta_2([\xi]^{i_k} P_k, [\xi]^{i_l} P_l)\right)^{2^{2n-5}}} \right). \end{aligned}$$

Avant d'aller plus loin, justifions les puissances au dénominateur.
Dans le développement de $\sigma^{(n)}$

$$\sigma^{(n)}(u_1, \ldots, u_n) = \frac{\prod_\epsilon \Delta_n(\epsilon_1 u_1, \ldots, \epsilon_n u_n)}{\left(\prod_{i<j} \Delta_2(u_i, u_j)\right)^{2^{n-2}}},$$

on a une puissance 2^{n-2} qui apparaît.
On fait le produit sur tous les $(i_2, \ldots, i_n) \in \{0,1\}^{n-1}$ possibles. Ainsi, pour le premier produit, il y a 2^{n-2} $(n-1)$-uplets pour chaque $i_j \in \{0,1\}$ fixé, tandis que pour le second produit, il y a 2^{n-3} $(n-2)$-uplets pour chaque couple $(i_k, i_l) \in \{0,1\}^2$ fixé.

Regardons justement ce dénominateur de plus près.
Le premier produit vaut

$$\begin{aligned} \prod_{j>1} \prod_{i_j \in \{0,1\}} \left(\Delta_2(P_1, [\xi]^{i_j} P_j)\right)^{4^{n-2}} &= \prod_j \left(\begin{vmatrix} 1 & x_1 \\ 1 & x_j \end{vmatrix} \cdot \begin{vmatrix} 1 & x_1 \\ 1 & -x_j \end{vmatrix} \right)^{4^{n-2}} \\ &= \left(\prod_j (x_j^2 - x_1^2) \right)^{4^{n-2}}. \end{aligned}$$

En particulier, c'est bien un polynôme en $c_1, \ldots, c_n$. On traite de même la seconde partie du dénominateur :

$$\begin{aligned} &\prod_{k<l} \prod_{i_j \in \{0,1\}} \left(\Delta_2([\xi]^{i_k} P_k, [\xi]^{i_l} P_l)\right)^{2^{2n-5}} \\ &= \prod_{k<l} \left(\begin{vmatrix} 1 & x_k \\ 1 & x_l \end{vmatrix} \cdot \begin{vmatrix} 1 & x_k \\ 1 & -x_l \end{vmatrix} \cdot \begin{vmatrix} 1 & -x_k \\ 1 & x_l \end{vmatrix} \cdot \begin{vmatrix} 1 & -x_k \\ 1 & -x_l \end{vmatrix} \right)^{2^{2n-5}} \\ &= \left(\prod_{k<l} (x_k^2 - x_l^2)^2 \right)^{2^{2n-5}} \\ &= \left(\prod_{k<l} x_k^2 - x_l^2 \right)^{4^{n-2}}. \end{aligned}$$

Encore une fois, c'est bien un polynôme en $c_1, \ldots, c_n$.
Ainsi, pour pouvoir affirmer que f est bien le n-ième polynôme de sommation relatif à c, il ne nous reste plus qu'à montrer que le produit des Δ_n au numérateur est une fonction paire en x_1.
On considère un des termes $\Delta_n(P_1, \epsilon_2[\xi]^{i_2}P_2, \ldots, \epsilon_n[\xi]^{i_n}P_n)$ de ce produit :

$$\begin{aligned}\Delta_n(P_1, \epsilon_2[\xi]^{i_2}P_2, \ldots, \epsilon_n[\xi]^{i_n}P_n) &= \begin{vmatrix} 1 & x_1 & \ldots & y_1 x_1^b \\ 1 & (-1)^{i_2}x_2 & \ldots & \epsilon_2 i^{i_2} y_2((-1)^{i_2}x_2)^b \\ & & \vdots & \end{vmatrix} \\ &= (-1)^m \Delta_n([\xi]P_1, \epsilon_2[\xi]^{i_2+1}P_2, \ldots, \epsilon_n[\xi]^{i_n+1}P_n),\end{aligned}$$

car on multiplie $m = \lceil \frac{a}{2} \rceil + \lceil \frac{b}{2} \rceil$ colonnes par -1 pour obtenir cette égalité.
Ainsi, en prenant tous les termes $\Delta_n(P_1, \epsilon_2[\xi]^{i_2}P_2, \ldots, \epsilon_n[\xi]^{i_n}P_n)$, on trouve bien que le numérateur est une fonction en c_1. En conclusion, nous avons :

Théorème 4.5.7. *La fonction*

$$\prod \sigma^{(n)}(x_1, \pm x_2, \ldots, \pm x_n)$$

est le n-ième polynôme de sommation relatifs à la variable $c = x^2$. On le nomme donc ici $\hat{\sigma}^{(n)}$.

Nous avons la propriété suivante :

Proposition 4.5.8. *$\hat{\sigma}^{(n)}$ est symétrique en $(c_2, \ldots, c_n)$. Après symétrisation, il est de degré total 4^{n-2}.*

Démonstration. On a déjà vu la symétrie.
Pour ce qui est du degré, il suffit de voir que $\deg_{x_i}(\sigma^{(n)}) = 2^{n-2}$. On multiplie 2^{n-1} tels polynômes pour construire $\hat{\sigma}^{(n)}$, on a donc un polynôme de degré

$$2^{2n-3} = 2\left(4^{n-2}\right)$$

en x_i, soit 4^{n-2} en $c_i = x_i^2$.

□

Exemple 4.5.9. *Je rappelle que je travaille sur la courbe $y^2 = x^3 + Ax$. Pour $n = 3$, j'utilise $e_1 = c_2 + c_3$ et $e_2 = c_2c_3$, j'obtiens*

$$\hat{\sigma}^{(3)} = c_1^4 e_1^4 - 4(c_1^3 A^2 + c_1^3 e_2)e_1^3 + \ldots$$

Pour $n = 4$, j'ai deux méthodes pour calculer $\hat{\sigma}^{(4)}$: je peux me servir de $\sigma^{(4)}$ comme je m'étais servi de $\sigma^{(3)}$ pour calculer $\hat{\sigma}^{(3)}$:

$$\hat{\sigma}^{(4)} = \prod \sigma^{(4)}(x_1, \pm x_2, \pm x_3, \pm x_4);$$

ou bien je peux utiliser la formule du résultant, comme l'indique le théorème 4.5.16 énoncé plus loin :

$$\hat{\sigma}^{(4)}(c_1, c_2, c_3, c_4) = Res_c(\hat{\sigma}^{(3)}(c_1, c_2, c)\hat{\sigma}^{(3)}(c_3, c_4, c)).$$

Pour "symétriser" ce polynôme j'utilise les polynômes symétriques élémentaires que je note :

$$e_1 = c_2 + c_3 + c_4,\ e_2 = c_2c_3 + c_2c_4 + c_3c_4 \text{ et } e_3 = c_2c_3c_4.$$

Après symétrisation, on vérifie que ces deux formules donnent bien le même résultat :

$$\hat{\sigma}^{(4)} = A^{32}e_1^{16} + (-544c_1^2A^{28}e_3 - 16c_1A^{32} - 16c_1A^{30}e_2 - 448c_1A^{29}e_3 - 16A^{30}e_3)e_1^{15} + \dots$$

C'est bien un polynôme de degré total $4^{4-2} = 16$ comme annoncé. Pour la petite histoire, il a 10351 monômes.

4.5.2.3 Le cas $j = 0$

Le modèle de Weierstrass de $\mathcal{E}$ que l'on considère est maintenant $y^2 = x^3 + B$. On travaille avec l'automorphisme

$$[\xi](x, y) = (Jx, y),$$

où $J^2 + J + 1 = 0$.

Remarque 4.5.10. *La racine cubique de l'unité J ne doit pas être confondue avec le j-invariant (ni avec la jacobienne $\mathcal{J}$ d'une quelconque courbe).*

Avec exactement le même raisonnement que dans le cas $j = 1728$, on a le résultat suivant :

Théorème 4.5.11. *La quantité*

$$\prod_{0 \le i_k \le 2} \sigma^{(n)}(P_1, [\xi]^{i_2}P_2, \dots, [\xi]^{i_n}P_n) = \frac{\prod_{\epsilon_k \in \{\pm 1\}, i_k \in \{0,1,2\}} \Delta_n(P_1, \epsilon_2[\xi]^{i_2}P_2, \dots, \epsilon_n[\xi]^{i_n}P_n)}{\left(\prod_{i<j}(y_i^2 - y_j^2)\right)^{6^{n-2}}}$$

est le n-ième polynôme de sommation relatif à $c = y^2$.
C'est donc un polynôme en $y_1^2, \dots, y_n^2$, symétrique en $c_2, \dots, c_n$. Après symétrisation, il est de degré 6^{n-2}.

Démonstration. Le raisonnement étant le même que pour le théorème 4.5.7, je ne donne ici que le squelette de la preuve.

Dans l'égalité énoncée, le numérateur est clair, il n'y a que le dénominateur à justifier :

$$\prod_{j>1}\Big(\Delta_2(P_1,[\xi]^{i_j}P_j)\Big)^{6^{n-2}}\prod_{2\leq k<l}\Big(\Delta_2([\xi]^{i_k}P_k,[\xi]^{i_l}P_l)\Big)^{6^{n-3}3}$$

avec ici les indices $i_k \in \{0,1,2\}$.
Comme pour 4.5.7, on coupe cette quantité en deux parties que l'on traite séparément. La première moitié vaut :

$$\begin{aligned}\prod_{k>1}\prod_{i_k\in\{0,1,2\}}\Big(\Delta_2(P_1,[\xi]^{i_k}P_k)\Big)^{6^{n-2}} &= \prod_k\left(\begin{vmatrix}1 & x_1\\ 1 & x_k\end{vmatrix}\cdot\begin{vmatrix}1 & x_1\\ 1 & Jx_k\end{vmatrix}\cdot\begin{vmatrix}1 & x_1\\ 1 & J^2x_k\end{vmatrix}\right)^{6^{n-2}}\\ &= \left(\prod_j(x_j^3-x_1^3)\right)^{6^{n-2}}\\ &= \left(\prod_j(y_j^2-y_1^2)\right)^{6^{n-2}}\end{aligned}$$

car on rappelle que $y^2 = x^3 + B$. On effectue ensuite un travail similaire pour la seconde moitié du dénominateur et on a notre égalité.

Ensuite, avec exactement les mêmes arguments que pour 4.5.7, on voit que cette quantité est bien symétrique en $P_2,\dots,P_n$, et qu'elle permet bien de tester les égalités voulues :

$$P_1 + [\xi^{i_2}]P_2 + \cdots + [\xi^{i_n}]P_n = \mathcal{O},$$

pour l'automorphisme $[\xi] : (x,y) \mapsto (Jx,-y)$.

Enfin il reste à voir que c'est bien un polynôme en $c_i = y_i^2$. Si on note provisoirement

$$f(P_1,\dots,P_n) = \prod_{0\leq i_k\leq 5}\sigma^{(n)}(P_1,[\xi]^{i_2}P_2,\dots,[\xi]^{i_n}P_n),$$

ceci revient à dire que

$$f(P_1,\dots,P_n) = f([\xi]P_1,\dots,P_n) = \cdots = f([\xi]^5P_1,\dots,P_n),$$

d'une part, et

$$f(P_1,P_2\dots,P_n) = f(P_1,[\xi]P_2,\dots,P_n) = \cdots = f(P_1,[\xi]^5P_2,\dots,P_n)$$

d'autre part : la symétrie en $(P_2,\dots,P_n)$ permettra de conclure.

Ceci se traite exactement comme pour 4.5.7 : la dépendance en c_1 se résout en considérant le produit

$$\prod_{\substack{\epsilon_k\in\{\pm 1\}\\ 0\le i_k\le 5}} \Delta_n(P_1, \epsilon_2[\xi]^{i_2}P_2, \dots, \epsilon_n[\xi]^{i_n}P_n)$$

qui apparaît au numérateur du développement de f, tandis que la dépendance en c_2 se voit directement dans l'expression

$$f(P_1,\dots,P_n) = \prod_{0\le i_k\le 5} \sigma^{(n)}(P_1, [\xi]^{i_2}P_2, \dots, [\xi]^{i_n}P_n).$$

□

Mais il existe encore une autre construction possible de polynômes de sommation, cette fois-ci relatifs à la variable $c = y$. Elle est inspirée de [14].

Théorème 4.5.12.

- *Pour $n = 2$, on a l'égalité*

$$\frac{\sigma(P+Q)\sigma(P+[\xi]Q)\sigma(P+[\xi^2]Q)}{\sigma(P)^3\sigma(Q)^3} = y_P - y_Q.$$

  Ceci est le second polynôme de sommation relatif à $c = y$. Pour comparer avec le second point, on pourra remarquer qu'il est de degré $3^{2-2} = 1$ en y_Q.

- *Soit $n > 2$. Soient $P_1, \dots, P_n \in \mathcal{E}$ de coordonnées (x_i, y_i). Pour $2 \le k \le n$, soit $0 \le i_k \le 2$ un entier. On a alors l'égalité*

$$\frac{\prod_{0\le i_k\le 2}\sigma(P_1 + [\xi]^{i_2}P_2 + \cdots + [\xi]^{i_n}P_n)}{\left(\prod_{i=1}^n \sigma(u_i)\right)^{3^{n-1}}} = \frac{\prod_{0\le i_k\le 2}\Delta_n(P_1, [\xi]^{i_2}P_2, \cdots + [\xi]^{i_n}P_n)}{\left(\prod_{i<j}(y_i - y_j)\right)^{3^{n-2}}}.$$

  Cette quantité est le polynôme de sommation relatif à y. Elle est symétrique en $(y_2, \dots, y_n)$. Après symétrisation, c'est un polynôme de degré total 3^{n-2}.

Dans ce théorème, il y a à chaque fois deux choses à démontrer : l'égalité annoncée d'une part, et le fait qu'on obtienne bien le polynôme de sommation relatif à y d'autre part.
Le premier point du théorème, c'est-à-dire le cas $n = 2$, repose sur le lemme suivant, tiré de [14] :

Lemme 4.5.13 ([14]). *Pour tout point $(x, y) \in \mathcal{E}$, on a*

$$\sigma([\xi](x, y)) = \sigma(Jx, y) = J\sigma(x, y).$$

De cette égalité, on en déduit que :

$$\sigma([\xi]P)^3 = \sigma(P)\sigma([\xi]P)\sigma([\xi^2]P) = \sigma(P)^3,$$

et d'autre part que :

$$\frac{\sigma([1-\xi]P)}{\sigma(P)^3} = (1-J)x_P.$$

Muni de ce lemme, montrons le premier point :

Démonstration du premier point.
On utilise la formule de Frobenius-Stickelberger 4.3.2 pour $n = 3$:

$$-\frac{\sigma(u_1+u_2+u_n)\prod_{i<j}\sigma(u_i-u_j)}{\sigma(u_1)^3\sigma(u_2)^3\sigma(u_3)^3} = \Delta_3(u_1, u_2, u_3)$$

avec $u_1 = P$, $u_2 = Q$ et $u_3 = [\xi]Q$.
Il ne reste plus qu'à simplifier en utilisant le lemme pour obtenir l'égalité annoncée.

Le fait qu'on ait là le second polynôme pour y est juste la traduction du fait suivant :
deux points P et Q de $\mathcal{E}$ ont même y si et seulement si ils sont dans la même orbite sous $[\xi]$. □

On passe maintenant au calcul du polynôme de sommation pour les plus grands n.

Démonstration du second point.
Encore une fois, on utilise le théorème de Frobenius-Stickelberger :

$$(-1)^{(n-1)(n-2)/2}\frac{\sigma(u_1+\cdots+u_n)\prod_{i<j}\sigma(u_i-u_j)}{(\prod_i\sigma(u_i))^n} = \Delta_n(u_1, \ldots, u_n)$$

Ainsi, si l'on veut le produit de tous les $\Delta_n(P_1, [\xi]^{i_2}P_2, \cdots + [\xi]^{i_n}P_n)$:

$$\prod \Delta_n(P_1, [\xi]^{i_2}P_2, \ldots, [\xi]^{i_n}P_n) = \prod_{i_2,\ldots,i_n} \sigma(P_1+[\xi]^{i_2}P_2\cdots+[\xi]^{i_n}P_n)\cdot \prod_{i_2,\ldots,i_n}\left(\frac{\prod_{k<l}\sigma([\xi]^{i_k}P_k - [\xi]^{i_l}P_l)}{\prod_k\sigma([\xi]^{i_k}P_k)^n}\right).$$

Dans le membre de droite, le premier terme

$$\prod_{i_2,\dots,i_n} \sigma(P_1 + [\xi]^{i_2} P_2 \cdots + [\xi]^{i_n} P_n)$$

est satisfaisant : c'est ce que l'on veut pour tester des égalités comme dans la définition 4.5.4. Essayons de simplifier le second terme

$$\prod_{i_2,\dots,i_n} \left(\frac{\prod_{k<l} \sigma([\xi]^{i_k} P_k - [\xi]^{i_l} P_l)}{\prod_k \sigma([\xi]^{i_k} P_k)^n} \right).$$

Le produit se porte sur 3^{n-1} termes, et le dénominateur se simplifie donc via le lemme en

$$\prod_i \sigma(P_i)^{n3^{n-1}}.$$

Pour le numérateur, on fixe deux indices $k < l$. Pour un couple (i_k, i_l) choisi, il existe $n-3$ uplets dans $\{0,1,2\}^{n-1}$ convenables. Le produit vaut donc :

$$\begin{aligned}\prod_{k<l} &(\sigma(P_k - P_l)\sigma(P_k - [\xi]P_l)\sigma(P_k - [\xi]^2 P_l) \\ &\sigma([\xi]P_k - P_l)\sigma([\xi]P_k - [\xi]P_l)\sigma([\xi]P_k - [\xi]^2 P_l) \\ &\sigma([\xi]^2 P_k - P_l)\sigma([\xi]^2 P_{[}\xi]k - P_l)\sigma([\xi]^2 P_k - [\xi]^2 P_l))^{3^{n-3}}.\end{aligned}$$

En utilisant le lemme, on voit que les trois lignes sont égales. Ainsi, en rassemblant les morceaux :

$$\begin{aligned}\prod \Delta_n(P_1, [\xi]^{i_2} P_2, \dots, [\xi]^{i_n} P_n) = &\prod_{i_2,\dots,i_n} \sigma(P_1 + [\xi]^{i_2} P_2 \cdots + [\xi]^{i_n} P_n) \\ &\cdot \prod_{k<l} \left(\frac{\sigma(P_k - P_l)\sigma(P - [\xi]P_l)\sigma(P_k - [\xi^2]P_l)}{\sigma(P_k)^3 \sigma(P_l)^3} \right)^{3^{n-2}} \\ &\cdot \left(\frac{1}{\prod_i \sigma(P_i)} \right)^{3^{n-1}}.\end{aligned}$$

En utilisant le premier point pour simplifier la seconde ligne, on obtient le résultat voulu. □

Remarque 4.5.14. *La différence entre les deux théorèmes précédents (4.5.11 et 4.5.12) réside dans le fait que dans le premier, $c = y^2$ alors que dans le second $c = y$.*
Autrement dit, considérons l'automorphisme

$$[\xi] : (x, y) \mapsto (Jx, y).$$

Dans le premier on teste les relations

$$P_1 \pm [\xi^{i_2}]P_2 \pm \cdots \pm [\xi^{i_n}]P_n = \mathcal{O},$$

où les entiers i_k sont compris entre 0 et 2. Autrement dit, l'automorphisme associé à $c = y^2$ est $-[\xi]$ qui est d'ordre 6.
Dans le second théorème, on teste les relations

$$P_1 + [\xi^{i_2}]P_2 + \cdots + [\xi^{i_n}]P_n = \mathcal{O},$$

l'automorphisme considéré est $[\xi]$ qui est d'ordre 3.

On y reviendra plus loin, mais on peut d'ors et déjà remarquer que cela signifie que le théorème 4.5.11 nous donne des polynômes de plus haut degré (6^{n-2} contre 3^{n-2}) qui trouveront plus de relations, car les orbites du premier automorphisme sont de taille 6, contre 3 pour le second.

Exemple 4.5.15. *Nous avons deux théorèmes donnant deux familles alternatives de polynômes de sommation différentes, selon le choix de c. Regardons ce qu'ils nous donnent pour $n = 3$.*
Dans les deux cas, je note $e_1 = c_2 + c_3$ et $e_2 = c_2c_3$.

Avec le théorème 4.5.11, on a $c = y^2$, et j'obtiens le polynôme de degré $6^{3-2} = 6$

$$\hat{\sigma}^{(3)} = e_1^6 - (540c_1^5B^2 + 6c_1^5e_2 + 1944c_1^4B^3 + 1458c_1^3B^4)e_1^5 + \dots$$

Avec le théorème 4.5.12, on a $c = y$, et j'obtiens le polynôme de degré $3^{3-2} = 3$

$$\hat{\sigma}^{(3)} = c_1^3e_1^3 - (18c_1^2B - 3c_1^2e_2 + 27B^2)e_1^2 + \dots$$

4.5.2.4 Théorème final avant de passer au cas hyperelliptique

Ici, on énonce le théorème annoncé dans l'exemple 4.5.9 : pour le calcul des polynômes alternatifs de sommation dans le cas elliptiques, on peut utiliser la formule du résultant.

Théorème 4.5.16. *Les trois polynômes de sommation présentés dans les théorèmes 4.5.7, 4.5.11 et 4.5.12, c'est-à-dire relatifs aux variables $c = x^2$ dans le cas $j = 1728$ et $c = y^2$ et y dans le cas $j = 0$, vérifient tous la formule du résultant :*

$$\hat{\sigma}^{(n)}(c_1, \dots, c_n) = Res_c(\hat{\sigma}^{(n-k)}(c_1, \dots, c_{n-k-1}, c), \hat{\sigma}^{(k+2)}(c_{n-k}, \dots, c_n, c))$$

pour tout $n \geq 4$ et $1 \leq k \leq n-3$.

Démonstration. Traitons uniquement le cas du théorème 4.5.7, les deux autres cas se traitant exactement de la même façon. Le cheminement de la preuve est le même que celle de la proposition 4.3.8 : il faut calculer le degré, le coefficient dominant et les racines du polynôme

$$f(C) = \hat{\sigma}^{(n)}(c_1, \ldots, c_{n-1}, C),$$

puis les rentrer dans la formule

$$\begin{aligned}&\mathrm{Res}_C(\hat{\sigma}^{(n-k)}(C_1, \ldots, C_{n-k-1}, C), \hat{\sigma}^{(k+2)}(C_{n-k}, \ldots, C_n, C))\\ =&a^m \prod_i \hat{\sigma}^{(n-k)}(C_1, \ldots, C_{n-k-1}, \alpha_i).\end{aligned}$$

Le degré de f a déjà été démontré plus haut : il vaut 4^{n-2}.
Pour le coefficient dominant, je reprend l'expression comme quotient de déterminants de f :

$$\frac{\prod_{\epsilon_k \in \{\pm 1\}, i_k \in \{0,1\}} \Delta_n(P_1, \epsilon_2[\xi]^{i_2} P_2, \ldots, \epsilon_n[\xi]^{i_n} P_n)}{\left(\prod_{i<j}(x_i^2 - x_j^2)\right)^{4^{n-2}}}.$$

En développant les déterminants du numérateur selon leur dernière ligne on trouve que le coefficient dominant de f vaut

$$\hat{\sigma}^{(n-1)}(c_1, \ldots, c_{n-1})^4.$$

En rentrant ces données dans la formule du résultant rappelée juste au-dessus, on achève la démonstration.

□

4.5.3 Le cas général

Ainsi que spécifié dans la remarque 4.5.5, le point de départ de cette section est [14]. Le choix des modèles de courbes (que j'ai légèrement modifiés) et des automorphismes à considérer ainsi que les différents lemmes en sont tirés. Ma contribution se trouve dans le théorème 4.5.20.

Suivant donc [14], je considère les courbes hyperelliptiques décrites par les équations suivantes :

$$\begin{aligned}y^2 =& x^{rs+1} + \mu_{r(s-1)+1} x^{r(s-1)+1} + \cdots + \mu_{r+1} x^{r+1} + \mu_1 x \text{ pour } rs = 2g,\\ y^2 =& x^{rs} + \mu_{r(s-1)} x^{r(s-1)} + \cdots + \mu_r x^r + \mu_0 \text{ pour } rs = 2g+1.\end{aligned}$$

Remarque 4.5.17. *Par rapport à [14], j'ai dû modifier les notations : ce qui chez eux était le couple (a, m) s'appelle maintenant (r, s), la lettre a étant déjà utilisée.*

Soit ξ une racine $2r$-ième de l'unité, l'automorphisme considéré est

$$[\xi](x, y) = \begin{cases} (\xi^2 x, \xi y) \text{ dans le premier cas ;} \\ (\xi^2 x, y) \text{ dans le second cas.} \end{cases}$$

Exemple 4.5.18. *Voyons qu'avec $g = 1$ on retrouve les situations décrites dans la sous section précédente.*

- *si $r = 1$, le morphisme n'est rien d'autre que $[-1]$;*
- *dans le premier modèle, si $r = 2$ et $s = 1$, la courbe est d'équation*

$$y^2 = x^3 + Ax$$

  et le morphisme est $(x, y) \mapsto (-x, iy)$ avec i racine carrée de -1.
- *dans le second modèle, si $r = 3$ et $s = 1$, on retrouve la courbe*

$$y^2 = x^3 + B$$

  et le morphisme $(x, y) \mapsto (Jx, y)$, avec J racine cubique de l'unité.

Exemple 4.5.19. *Regardons maintenant le cas général.*

- *si $r = 1$, encore une fois, le morphisme est $[-1]$;*
- *dans le premier modèle, si $r = 2$ et $s = g$, la courbe est d'équation*

$$y^2 = x^{2g+1} + \mu_{2g-1} x^{2g-1} + \cdots + \mu_1 x$$

  et le morphisme est $(x, y) \mapsto (-x, iy)$ avec i racine carrée de -1.
- *dans le second modèle, si je prends r maximal, i.e. $r = 2g + 1$., le modèle de la courbe est*

$$y^2 = x^{2g+1} + \mu_0.$$

Je vais maintenant généraliser les théorèmes 4.5.7 et 4.5.11 :

Théorème 4.5.20. *On suppose que $s = 1$, c'est-à-dire que r vaut $2g$ ou $2g+1$ selon le modèle de courbe considéré.*
Soient $P_1, \dots, P_n \in \mathcal{C}$ de coordonnées (x_i, y_i). On a l'égalité :

$$\prod_{0 \leq i_k \leq r-1} \sigma^{(n,g)}(P_1, [\xi]^{i_2} P_2, \dots, [\xi]^{i_n} P_n) = \frac{\prod_{\epsilon_k \in \{\pm 1\}, 0 \leq i_k \leq r-1} \Delta_{n,g}(P_1, \epsilon_2 [\xi]^{i_2} P_2, \dots, \epsilon_n [\xi]^{i_n} P_n)}{\left(\prod_{i<j}(c_i - c_j)\right)^{(2r)^{n-2}}},$$

où $[\xi]$ est décrit plus haut et d'ordre r, et
— *$c = x^r = x^{2g}$ si $rs = 2g$;*
— *$c = y^2$ si $rs = 2g+1$.*
Cette quantité est alors le n-ième polynôme de sommation relatif à la variable c, qui permet de tester des égalités :

$$P_1 \pm [\xi]^{i_2} \pm \dots \pm [\xi]^{i_n} P_n \in \Theta.$$

C'est un polynôme symétrique en $(c_2, \dots, c_n)$ et après symétrisation il est de degré $(2r)^{n-2} g$.

Démonstration. Ce théorème se démontre de la même manière que les théorèmes 4.5.7 et 4.5.11. Reprenons donc le même schéma de raisonnement.

Il s'agit d'abord de démontrer l'égalité annoncée. En utilisant la définition de $\sigma^{(n,g)}$ (voir théorème 4.4.7), le terme de gauche de l'égalité annoncée se développe comme

$$\prod_{0 \leq i_k \leq r-1} \left(\frac{\prod_{\epsilon_k \in \{\pm 1\}} \Delta_n(P_1, \epsilon_2 [\xi]^{i_2} P_2, \dots, \epsilon_n [\xi]^{i_n} P_n)}{\prod_{j>1} \left(\Delta_2(P_1, [\xi]^{i_j} P_j)\right)^{(2r)^{n-2}} \prod_{2 \leq k < l} \left(\Delta_2([\xi]^{i_k} P_k, [\xi]^{i_l} P_l)\right)^{(2r)^{n-3r}}} \right).$$

Ainsi, le numérateur est bien celui que l'on voulait obtenir, et il nous faut travailler sur le dénominateur. Comme pour les théorèmes 4.5.7 et 4.5.11, les deux moitiés de ce dénominateur se traitent séparément. La première moitié donne :

$$\begin{aligned}
\prod_{k>1} \prod_{0 \leq i_k \leq r-1} \left(\Delta_2(P_1, [\xi]^{i_k} P_k)\right)^{(2r)^{n-2}} &= \prod_k \prod_{i_k} \left(\begin{vmatrix} 1 & x_1 \\ 1 & (\xi)^{2i_k} x_k \end{vmatrix} \right)^{(2r)^{n-2}} \\
&= \left(\prod_k (x_k^r - x_1^r) \right)^{(2r)^{n-2}} \\
&= \left(\prod_k (c_j^2 - c_1^2) \right)^{(2r)^{n-2}}
\end{aligned}$$

On effectue ensuite un travail similaire pour la seconde moitié du dénominateur et on a notre égalité.

Maintenant que l'égalité est démontrée, il s'agit de voir que cette quantité nous donne bien le n-ième polynôme relatif à la variable c, c'est-à-dire que c'est un polynôme en les c_i qui permet de tester les égalités

$$P_1 \pm [\xi]^{i_2} \pm \cdots \pm [\xi]^{i_n} P_n \in \Theta.$$

Pour le moment, notons $f(P_1, \ldots, P_n)$ cette quantité.

Le fait que f permette de tester les égalités voulues provient simplement de la définition de $\sigma^{(n,g)}$.

A priori, grâce au membre de gauche de l'égalité définissant f, on sait qu'il s'agit d'un polynôme en $x_1, \ldots, x_n$. Il faut voir qu'en réalité c'est un polynôme en les c_i.
Pour cela, montrons dans un premier temps que, vis-à-vis de P_1 et de P_2, f ne dépend que des variables c_1 et c_2. Ceci revient à dire que

$$f(P_1, \ldots, P_n) = f([\xi]P_1, \ldots, P_n) = \cdots = f([\xi]^{2r-1}P_1, \ldots, P_n),$$

d'une part, et

$$f(P_1, P_2 \ldots, P_n) = f(P_1, [\xi]P_2, \ldots, P_n) = \cdots = f(P_1, [\xi]^{2r-1}P_2, \ldots, P_n)$$

d'autre part.

La dépendance en c_1 se résout en considérant le produit

$$\prod_{\substack{\epsilon_k \in \{\pm 1\} \\ 0 \leq i_k \leq 2r-1}} \Delta_n(P_1, \epsilon_2[\xi]^{i_2}P_2, \ldots, \epsilon_n[\xi]^{i_n}P_n)$$

qui apparaît au numérateur du développement de f, tandis que la dépendance en c_2 se voit directement dans l'expression

$$f(P_1, \ldots, P_n) = \prod_{0 \leq i_k \leq 2r-1} \sigma^{(n)}(P_1, [\xi]^{i_2}P_2, \ldots, [\xi]^{i_n}P_n).$$

Grâce au membre de droite de l'égalité, on voit que f est symétrique en $P_2, \ldots, P_n$. Ceci permet de conclure que f est bien un polynôme en les c_i.

Enfin il ne reste plus qu'à démontrer le degré. Fixons un indice i. On a $\deg_{x_i} \sigma^{(n,g)} = 2^{n-2}g$. Pour construire f, il faut, d'après le terme de gauche de l'égalité, multiplier r^{n-1} tels polynômes, d'où

$$\deg_{x_i} f = (2r)^{n-2} rg.$$

Pour avoir le degré en c_i, il suffit de diviser cette quantité par r.

□

Remarque 4.5.21.

- *Il ne semble pas y avoir d'équivalent au théorème 4.5.12 pour $g > 1$, ni de résultat pour $s \neq 1$.*
- *Les deux modèles de courbes qu'on considère sont $y^2 = x^{2g+1} + \mu_1 x$ et $y^2 = x^{2g+1} + \mu_0$.*
- *Si $g > 1$, la formule du résultant est fausse pour $\hat{\sigma}^{(n,g)}$, et ce pour les mêmes raisons qu'elle était fausse pour $\sigma^{(n,g)}$.*

4.5.4 Analyse et comparaisons

Dans cette section, nous devons discuter de l'efficacité de ces polynômes de sommation alternatifs, relatifs à d'autres choix de coordonnées c que le choix initial $c = x$ fait par Semaev. Il s'agit ainsi de reprendre l'analyse faite dans la proposition 4.4.9, et de l'adapter à ces nouveaux polynômes. Dans cette proposition, je rappelle qu'on avait noté $\mathfrak{c}(g, n, q)$ le coût de la résolution du système polynomial sur $\mathbb{F}_q$ obtenu à partir de la relation

$$\sigma^{(n+g,g)}(x_1, \ldots, x_g, X_{g+1}, \ldots, X_{g+n}) = 0.$$

Remarque 4.5.22. *Dans cette section, on prendra garde de ne pas confondre la lettre c qui désigne une variable et la lettre gothique $\mathfrak{c}$ qui désigne un coût.*

Proposition 4.5.23. *On est dans la situation et avec les notations du théorème 4.5.20. On considère donc le polynôme de sommation suivant :*

$$\tilde{\sigma}^{n+g,g}(c_2, \ldots, c_{n+g}) = \prod_{0 \leq i_k \leq r-1} \sigma^{(n,g)}(P_1, [\xi]^{i_2} P_2, \ldots, [\xi]^{i_{n+g}} P_{n+g}).$$

Le coût de la résolution du système polynomial sur $\mathbb{F}_q$ déduit de la relation

$$\tilde{\sigma}^{n,g}(c_2, \ldots, c_{n+g}) = 0$$

est noté $\mathfrak{c}'(g, n, q)$ et vaut $\tilde{O}((2r)^{3n(n+g-2)} g^{3n} n)$ quand $q \to \infty$.

Démonstration. On rappelle qu'un tel coût est directement lié au nombre de variables (ici n : ce sont $c_{g+1}, \ldots, c_{g+n}$) et au degré des polynômes du système à résoudre (ici $(2r)^{g+n-2} g$, soit $4^{g+n-2} g^{g+n-1}$ ou $(4g+2)^{g+n-2} g$ selon la valeur de r).

Suivons maintenant les mêmes étapes que dans la démonstration de la proposition 4.4.9. On commence donc par une utiliser la borne de Macaulay pour estimer le degré de régularité. Ici

$$d_{reg} \leq n(2r)^{n+g-2} g - n + 1.$$

Muni de ce premier résultat, on peut estimer le coût de l'utilisation de l'algorithme $F4$ pour calculer une base de Gröbner pour l'order degrevlex : ce coût est majoré par

$$\tilde{O}\left(\binom{n + d_{reg}}{n}^{\omega}\right).$$

Utilisant le fait que n soit négligeable devant $n + d_{reg}$ et la formule de Stirling, ce coût devient

$$\tilde{O}\left(\left((2r)^{n(n+g-2)} g^n e^n n^{-1/2}\right)^{\omega}\right).$$

Il s'agit maintenant d'étudier la complexité de l"algorithme de changement d'ordre $FGLM$. Pour cela, on commence par utiliser la borne de Bézout pour majorer le degré D de l'idéal polynomial que l'on souhaite étudier :

$$D \leq (2r)^{n(n+g-2)} g^n.$$

Le coût de cette étape est alors

$$O((2r)^{3n(n+g-2)} g^{3n} n).$$

Comme le coût de la seconde étape domine celui de la première, c'est lui qui donne l'estimation de $\mathfrak{c}'(g, n, q)$. □

Je rappelle qu'avec les polynômes de sommation classiques, le coût était de $\tilde{O}(2^{3n(n+g-2)} g^{3n} n)$: ainsi, l'utilisation des polynômes alternatifs aboutit à des systèmes dont la résolution est environ $r^{3n(n+g-2)}$ plus chère, avec r qui est de l'ordre de $2g$.
D'un autre côté, il faut tenir compte du fait que ces nouveaux polynômes détectent r fois plus de relations. En effet, avec les polynômes de sommation classiques, on recherchait les relations du type

$$D_1 \pm D_2 + \cdots \pm D_{g+n} \in \Theta,$$

tandis que les nouveaux polynômes recherchent plus de relations :

$$\hat{\sigma}^{(n)}(c_1, \ldots, c_{g+n}) = 0 \Leftrightarrow D_1 + [\xi^{i_2}] D_2 + \cdots + [\xi^{i_{g+n}}] D_{g+n} \in \Theta,$$

pour les indices i_k dans $\{0, \ldots, 2r\}$.

Ainsi, utiliser les polynômes alternatifs permet de devoir tester r fois moins de relations avant d'en trouver une, mais chaque relation devient $r^{3n(n+g-2)}$ fois plus chère à tester : très clairement, ces nouveaux polynômes ne sont pas du tout compétitifs par rapport aux polynômes de sommation classiques.

Exemple 4.5.24. *Regardons la situation en genre* 1 *sur les courbes de j invariant* 1728*, c'est-à-dire d'équation de Weierstrass* $y^2 = x^3 + Ax$*. Si on a un point* $R \in \mathcal{E}(\mathbb{F}_{q^n})$ *de la courbe à décomposer, on peut utiliser le polynôme de Semaev* $\sigma^{n+1}(x_R, x_1, \ldots, x_n)$*, qui s'annule sur les abscisses des points* P_i *vérifiant*

$$R \pm P_1 \pm \cdots \pm P_n = \mathcal{O};$$

ou bien on peut utiliser le polynôme

$$\hat{\sigma}^{n+1} = \prod \sigma^{(n)}(x_1, \pm x_2, \ldots, \pm x_n),$$

qui détecte les relations du type

$$R + [\xi^{i_1}]P_1 + [\xi^{i_2}]P_2 + \cdots + [\xi^{i_n}]P_n = \mathcal{O},$$

avec les i_k *dans* $\{0, \ldots, 3\}$*. En rappelant que*

$$[\xi^i]P = \begin{cases} P \text{ si } i = 0 \\ P' \text{ si } i = 1 \\ -P \text{ si } i = 2 \\ -P' \text{ si } i = 3 \end{cases}$$

avec $P' = (-x, iy)$ *quand* $P = (x, y)$*, on voit qu'effectivement* $\hat{\sigma}^{n+1}$ *détecte* 2 *fois plus de relations que* σ^{n+1}*.*
D'un autre côté, comme σ^{n+1} *est de degré* 2^{n-1} *tandis que* $\hat{\sigma}^{n+1}$ *de degré* 4^{n-1}*, utiliser* $\hat{\sigma}^{n+1}$ *entraîne un surcoût très important : chaque test de relation coûte* $2^{3n(n-1)}$ *fois plus.*

4.6 Conclusion

Le but premier de de chapitre était la généralisation des polynômes de Semaev aux courbes hyperelliptiques. Celle ci passe par une réécriture de ces polynômes de sommation comme produit de déterminants. En effet, la construction initiale de Semaev était basée sur une formule de résultants, qui n'est plus valide en genre supérieur à 1. Une fois ces polynômes de sommation hyperelliptiques construits, on a pu étudier leurs propriétés : symétrie, degré, impact sur le coût de la recherche de relations.

Suite à cela, j'ai également cherché à construire des polynômes de sommation alternatifs, non plus rattachés à l'automorphisme de degré 2 $(x, y) \mapsto (x, -y)$, mais à d'autres automorphismes de degré $2r$. L'idée sous jacente à cette recherche est la suivante : plus le degré de l'automorphisme est élevé, plus le nombre de relations détectés par l'égalité

$$\hat{\sigma}^{(n+g)} = 0$$

est important. Malheureusement, l'analyse montre que finalement le trop haut degré de ces nouveaux polynômes les rend inutilisables.

Bibliographie

[1] J. Adleman, J.DeMarrais, et M. Huang. A subexponential algorithm for discrete logarithms over the rational subgroup of the Jacobians of large genus hyperelliptic curves over finite fields. Leonard M. Adleman and Ming-Deh Huang, editors, Algorithmic Number Theory, volume 877 of Lecture Notes in Computer Science, pp 28–40, Berlin, 1994. Springer-Verlag.

[2] R. Avanzi. Aspects of Hyperelliptic Curves over Large Prime Fields in Software Implementations , Cryptographic Hardware and Embedded Systems - CHES 2004, Lecture Notes in Computer Science Volume 3156, 2004, pp 148-162

[3] R. Avanzi, H. Cohen, C. Doche, G. Frey, T. Lange, K. Nguyen and F. Vercauteren. Handbook of elliptic and hyperelliptic curve cryptography, Discrete mathematics and its applications, 2005.

[4] J. Balakrishnan, J. Belding, S. Chisholm, K. Eisenträger, K. Stange et E.Teske. Pairings on hyperelliptic curves, CoRR Vol. abs/0908.3731, 2009.

[5] I. Blake, G. Seroussi, N. Smart. Advances in elliptic curve cryptography. London Mathematical Society Lecture Note Series 317, Cambridge University press (2005), pp. 183-212.

[6] D. Boneh et M. Franklin, "Identity-based encryption from the Weil pairing, Advances in Cryptology – CRYPTO 2001, Lecture Notes in Computer Science, 2139 (2001), 213–229.

[7] D. Boneh, B. Lynn et H. Shacham, Short signatures from the Weil pairing, Advances in Cryptology – ASIACRYPT 2001, Lecture Notes in Computer Science, 2248 (2001), 514–532.

[8] E. Brown et B. T. Myers. Elliptic curves from Mordell to Diophantus and back (se trouve à l'adresse : http://www.math.vt.edu/people/brown/doc/dioellip.pdf).

[9] V. Buchstaber et V. Enolskii. Explicit Algebraic Description of Hyperelliptic Jacobians on the Basis of the Klein σ-Functions, Functional Analysis and Its Applications, Vol. 30, No. 1, 1996.

[10] V. Buchstaber, V. Enolskii, et D. Leykin, A Recursive Family of Differential Polynomials Generated by the Sylvester Identity and Addition Theorems for

Hyperelliptic Kleinian Functions, Functional Analysis and Its Applications, Vol. 31, No. 4, 1997.

[11] D. Cantor. Computing in the Jacobian of a hyperelliptic curve. Mathematics of computation 48 (1987), no. 177, 95-101.

[12] A. Devegili, C.hEigeartaigh, M. Scott, R. Dahab. Multiplication and squaring on pairing-friendly fields. Cryptology ePrint Archive, Report 2006/471

[13] J. Eilbeck, M. England, Y. Ônishi. Abelian functions associated with genus three algebraic curves , LMS J. Comput. Math., vol. 14 (2011), pp.291-326.

[14] J.C. Eilbeck, S. Matsutani, Y. Onishi. Addition formulae for abelian functions associated with specialized curves, Phil. Trans. R. Soc., A2011 369, février 2011.

[15] J-C Faugère. A new efficient algorithm for computing Gröbner bases (F4). Journal of Pure and Applied Algebra, 139(1–3) :61–88, June 1999.

[16] J-C Faugère, P. Gianni, D. Lazard et T. Mora. Efficient computation of zero-dimensional Gröbner bases by change of ordering. Journal of Symbolic Computation, 16(4) :329–344, 1993.

[17] F. Frobenius et L. Stickelberger, Zur Theorie der elliptischen Functionen, J. reine angew. Math. 83 (1877), 175–179.

[18] W. Fulton. Algebraic curves. Math. Lec. Note Series, W. A. Benjamin Inc, 1969

[19] S. Galbraith, F. Hess et F. Vercauteren. Hyperelliptic pairings, Pairing-Based Cryptography -Pairing 2007, Lecture Notes in Computer Science Volume 4575, 2007, pp 108-131 .

[20] R. Granger, F. Hess, R. Oyono, N. Thériault et F. Vercauteren, Ate pairing on hyperelliptic curves, Advances in Cryptology - Eurocrypt 2007, LNCS, vol. 4515, Springer-Verlag, 2007, pp. 419–436.

[21] P. Gaudry. Fast genus 2 arithmetic based on theta functions, J.Math.Cryptol.1 (2007), 243–265.

[22] P. Gaudry. Index calculus for abelian varieties of small dimension and the elliptic curve discrete logarithm problem, Journal of Symbolic Computation, vol. 44, no. 12, pp. 1690-1702, 2009.

[23] P. Gaudry, E. Thome, N. Theriault, C. Diem. A double large prime variation for small genus hyperelliptic index calculus, Mathematics of computation, vol. 76, pp. 475-492, 2007.

[24] P. Hewitt. A brief history of elliptic curves, notes de cours (se trouve à l'adresse : http://livetoad.org/Courses/Documents/132d/Notes/history_of_elliptic_curves.pdf)

[25] A. Joux. A one round protocol for tripartite Diffie-Hellman, Algorithmic Number Theory :4th International Symposium, ANTS-IV, Lecture Notes in Computer Science, 1838 (2000), 385–393.

[26] A. Koblitz, N. Koblitz, A. Menezes. Elliptic curve cryptography : The serpentine course of a paradigm shift, Journal of Number theory, vol. 131, issue 5 (2011), pp 781-814.

[27] N. Koblitz, Hyperelliptic cryptosystems, J. Cryptology, 1 (1989), pp. 139-150.

[28] N. Koblitz, A. Menezes, Pairing-based cryptography at high security levels, Proceedings of Cryptography and Coding 2005, volume 3796 of LNCS, pp. 13-36.

[29] T. Lange, Formulae for Arithmetic on Genus 2 Hyperelliptic Curves, Applicable Algebra in Engineering, Communication and Computing, vol. 15 (2003), pp. 295-328.

[30] D. Lubicz, D. Robert. Efficient pairing computation with theta functions. Algorithmic Number Theory, 9th international symposium, ANTS-IX, Nancy, France, July 2010, Proceedings.

[31] D. Lubicz, D. Robert. A generalisation of Miller's algorithm and applications to pairing computations on abelian varieties. Journal of Symbolic Computation, vol. 67, March–April 2015, pp. 68-92.

[32] A. Menezes, T. Okamoto et S. Vanstone, Reducing elliptic curve logarithms to logarithms in a finite field, IEEE Transactions on Information Theory, 39 (1993), pp. 1639-1646.

[33] V. Miller. Short programs for functions on curves, IBM Thomas J. Watson Research Center (se trouve à l'adresse : http://crypto.stanford.edu/miller/miller.ps), 1986.

[34] D. Mumford. Tata Lectures on Theta I. Volume 28 of Progress in Mathematics. Birkhäuser Boston Inc., Boston, MA, 1983. With the assistance of C. Musili, M. Nori, E. Previato and M. Stillman.

[35] D. Mumford. Tata Lectures on Theta II. Volume 28 of Progress in Mathematics. Birkhäuser Boston Inc., Boston, MA, 1983. With the assistance of C. Musili, M. Nori, E. Previato and M. Stillman.

[36] K. Nagao. Decomposition attack for the jacobian of a hyperelliptic curve over an extension field, Algorithmic number theory, Lecture notes in computer science, vol. 6197, pp. 285-300, 2010.

[37] National Institute of Standards and Technology, "Digital Signature Standard," Federal Information Processing Standards Publication 186-2, 2000.

[38] N. Ogura, N. Kanayama, S. Uchiyama and E. Okamoto. Cryptographic pairings based on elliptic nets, Advances in information and computer security (2011), pp. 65-78.

[39] Y. Onishi. Determinant expressions for hyperelliptic functions (with an appendix by Shigeki Matsutani), Tokyo journal of mathematics, vol. 27, n. 2, pp. 299-312, 2007.

[40] A. Rice et E. Brown, Why Ellipses are not elliptic curves, Mathematics Magazine, Vol. 85, No. 3 (June 2012), pp. 163-176.

[41] T. Saito, S. Yokoyama, T. Kobayashi et G. Yamamoto, Some relations between Semaev's summation polynomials and Stange's elliptic nets, J. Math-for-Ind. 3A (2011) 89-92.

[42] J. Silverman. The arithmetic of elliptic curves, Springer-Verlag, New York, 1985, pp 157-178.

[43] I. Semaev. Summation polynomials and the discrete logarithm problem on elliptic curves, Cryptologie ePrint Archive, Report 2004/031, 2004.

[44] N. Smart. On the performance of hyperelliptic cryptosystems, Advances in cryptology -Eurocrypt 99, Lecture notes in computer science, vol. 1892, 1999, pp. 165–175.

[45] K. Stange. The Tate pairing via elliptic nets, Pairing based cryptography-Pairing 2007, Lecture Notes in Computer Science Volume 4575, 2007, pp 329-348.

[46] P. Stevenhagen et B. de Smit. Kernvak Algebra, notes de cours (se trouve à l'adresse : http://websites.math.leidenuniv.nl/algebra/ellcurves.pdf)

[47] N. Theriault. Index calculus attack for hyperelliptic curves of small genus, Advances in Cryptology- ASIACRYPT 2003, lectures notes in computer science vol. 2894, pp. 75-92.

[48] C. Tran. Formulae for computation of Tate pairing on hyperelliptic curve using hyperelliptic nets, Progress in Cryptology - AFRICACRYPT 2014, Lecture Notes in Computer Science Volume 8469, 2014, pp 199-214

[49] Y. Uchida. Division polynomials and canonical local heights on hyperelliptic Jacobians, Manuscrypta Mathematica, vol. 134, issue 3-4 (2011), pp 273-308.

[50] Y. Uchida, S. Uchiyama. The Tate-Lichtenbaum pairing on a hyperelliptic curve via hyperelliptic nets, Pairing based cryptography-Pairing 2012, pp. 218-233

[51] S. Vanstone, Responses to NIST's Proposal, Communications of the ACM, 35, July 1992, 50-52 (communicated by John Anderson).

[52] V. Vitse, Attaques algébriques du problème du logarithme discret sur courbes elliptiques, dissertation de thèse, soutenue le 20 octobre 2011.

[53] K. Weierstrass Zur Theorie der Abelschen Functionen, Journ.reine angew.Math., 47 :289–306, 1854.

[54] K. Weierstrass Gesammelte Werke, volume 4. Teu ?bner, 1902.

[55] A. Weil. Sur les fonctions algébriques à corps de constantes finis, C.R.Acad.Sci.Paris,210 :592–594, 1940 (= Oeuvres Scientifiques, Volume I, pp. 257–259).

Résumé

Dans cette thèse, j'étudie deux aspects distincts de la cryptographie basée sur les courbes elliptiques et hyperelliptiques.
Dans une première partie, je confronte deux méthodes de calcul de couplages, originales car ne reposant pas sur le traditionnel algorithme de Miller. Ainsi, dans [45], K. Stange calcula le couplage de Tate sur une courbe elliptique à partir d'un nouvel outil, les elliptic nets. Y. Uchida et S. Uchiyama généralisèrent ces objets au cas hyperelliptique ([50]), mais ne donnèrent un algorithme pour le calcul de couplages que dans le cas des courbes de genre 2. Mon premier travail dans cette thèse fut de donner cet algorithme pour le cas général. De leur côté, D. Lubicz et D. Robert donnèrent dans [30] une autre méthode de calcul de couplage, basée sur les fonctions thêta. Le second résultat de ma thèse est de réunifier ces deux méthodes : je montre que la formule de récurrence à la base des nets est une conséquence des formules d'addition des fonctions thêta utilisées dans l'algorithme de Lubicz et Robert.
Dans la seconde partie de ma thèse, je me suis intéressé à l'algorithme de calcul d'index attaquant le problème du logarithme discret sur les courbes elliptiques et hyperelliptiques. Dans le cas elliptique, une des étapes principales de cette attaque repose sur les polynômes de Semaev. Je donne une nouvelle construction ces polynômes en utilisant la fonction sigma de Weierstrass, pour pouvoir ensuite les généraliser pour la première fois au cas hyperelliptique.

Abstract

In this thesis, I study two different aspects of elliptic and hyperelliptic curves based cryptography.
In the first part, I confront two methods of pairings computation, whose original feature is that they are not based the traditional Miller algorithm. Therefore, in [45], K. Stange computed Tate pairings on elliptic curves using a new tool, the elliptic nets. Y. Uchida and S. Uchiyama generalized these objects to hyperelliptic case ([50]), but they gave an algorithm for pairing computation only for the genus 2 case. My first work in this thesis was to give this algorithm for the general case. Meanwhile, D. Lubicz and D. Robert gave in [30] an other pairing computation method, based on theta functions. The second result of my thesis is the reunification of these two methods : I show that the recurrence equation which is the basis of nets theory is a consequence of the addition law of theta functions used in the Lubicz and Robert's algorithm.
In the second part, I study the index calculus algorithm attacking the elliptic and hyperelliptic curve discrete logarithm problem. In the elliptic case, one of the main steps of this attack requires the Semaev polynomials. I reconstruct these polynomials using Weierstrass sigma function, with the purpose of giving their first hyperelliptic generalization.