Université de Limoges
École doctorale Sciences et Ingénierie pour l'Information
Faculté des Sciences et Techniques
Département de Mathématiques et Informatique
Laboratoire XLIM

Thèse N° - 2011

Thèse

pour obtenir le grade de

Docteur de l'Université de Limoges

Discipline : Mathématiques et Applications

présentée et soutenue par

Aurore BERNARD

Formes quadratiques binaires et applications cryptographiques

Thèse soutenue le 8 décembre 2011

devant le jury composé de :

Rapporteurs :
Daniel AUGOT — Directeur de recherche INRIA Rocquencourt
Denis SIMON — Professeur, Université de Caen

Examinateurs :
Thierry BERGER — Professeur, Université de Limoges
François LAUBIE — Professeur, Université de Limoges
François ARNAULT — Maître de Conférences, Université de Limoges
Fabien LAGUILLAUMIE — Maître de Conférences, Université de Caen

Table des matières

Liste des figures

Liste des Algorithmes

Introduction

Les formes quadratiques binaires apparaissent progressivement au cours du XVII ème siècle, lorsque Descartes et Fermat introduisent la notion de coordonnées comme un outil algébrique pour résoudre des problèmes géométriques.

Ces formes ont de larges applications en mathématiques et en physique, plus particulièrement en géométrie, analyse numérique et topologie algébrique.

Une forme quadratique binaire est un polynôme homogène de degré deux à deux variables, ce qui peut être considéré comme l'équation cartésienne d'une surface $f(x, y) = ax^2 + bxy + cy^2$ sur une base donnée de $\mathbb{R}^2$.

Bien entendu, cette équation varie avec la base de l'expression, et il est naturel de définir une relation d'équivalence : regrouper toutes ces équations possibles dans des classes.

Sur le corps des nombres réels, il y a six classes : chacune correspond à une signature de Sylvester [40]. Elles sont distinguées par le signe de leur *discriminant* $\Delta_f = b^2 - 4ac$, et le signe de $a + c$.

Les formes de discriminant strictement négatif (formes *imaginaires*) ont un zéro unique à l'origine, qui est aussi leur unique extremum local et global.

Les formes de discriminant strictement positif (formes *réelles*) ont pour représentation une surface en forme de selle.

D'un autre coté, les formes quadratiques ont également été utilisées sur l'anneau des nombres entiers par Fermat, Lagrange et Gauss pour résoudre les problèmes de longue date de la théorie des nombres.

Cette fois les formes quadratiques binaires sont des équations à coefficients entiers, elles représentent un nuage de point discret sur une base d'un réseau $\mathbb{Z}^2$ donné.

Dans ce cas on définit une relation d'équivalence par changement de base. Les matrices de transformation sont maintenant unimodulaires (des matrices carrées à coefficients entiers dont le déterminant vaut $+1$ ou -1) et elles préservent la valeur du discriminant.

Les problèmes liés à cette équivalence sont plus compliqués que sur le corps des nombres réels. Par exemple dans les deux cas imaginaire et réel, nous ne connaissons pas d'algorithme de complexité polynomial qui calcule le nombre de classes d'équivalence pour un discriminant donné.

Décider de l'équivalence de deux formes est facile dans le cas imaginaire, où chaque classe contient une unique représentante réduite qui est calculable en temps polynomial.

Toutefois, le problème est difficile dans le cas réel, où il y a, en fonction de la notion de réduction, soit un nombre exponentiel de représentantes réduites qui sont calculables en temps polynomial, soit quelques représentantes calculables en temps exponentiel.

Un algorithme de réduction prend en entrée une forme quadratique et retourne une forme réduite (une représentante) et la *matrice de réduction*, cette dernière est une matrice unimodulaire de changement de base utilisée pour obtenir la forme réduite.

Le plus célèbre des algorithmes polynomiaux de réduction est l'algorithme de Lagrange [26] en 1773, communément appelé "réduction de Gauss" car Gauss reprend cet algorithme dans [20] en 1801.

En 1980 dans [25], Lagarias modifie l'algorithme de réduction de Gauss pour le rendre plus efficace.

C'est cet algorithme qui est utilisé dans la pratique, et nous l'appellerons l'algorithme de réduction de Gauss, ou l'algorithme de réduction *classique*, si nous avons besoin de le différencier de la nouvelle version que nous proposons.

La cryptanalyse de [10] montre que la possible petite taille des matrices de réduction a beaucoup d'importance dans l'application à la factorisation de certains grands nombres qui sont des clés publiques de cryptosystèmes, et tout particulièrement pour les cryptosystèmes *NICE* (voir [22, 24]).

Cependant, la meilleure majoration connue avant ma thèse, de la taille des matrices de réduction, qui est citée dans [25] en 1980, et dans [4] en 1999 (reprise dans [8] en 2007), est trop grande, ce qui entraîne que les résultats obtenus par [10] sur la factorisation sont heuristiques.

Dans cette thèse, nous avons mis en place une variante efficace de l'algorithme de réduction de Gauss des formes réelles. Cette variante (algorithme ***RéductionRéelle*GL2**) minimise la taille des matrices de réduction, et nous prouvons de façon constructive une majoration de la taille de ces matrices (théorème 4.2). Cette majoration est très fine à la fois dans le pire des cas mais aussi dans le cas moyen.

Un prolongement de ce nouvel algorithme est l'algorithme ***Cycle*** qui permet d'énumérer successivement toutes les représentantes réelles réduites équivalentes entre elles sans avoir besoin d'identifier une forme à son opposée (dont tous les coefficients sont de signes inverses) comme c'est le cas dans [8].

La majoration de la taille des matrices de réduction des formes réelles est plus délicate à obtenir que dans le cas imaginaire, car l'étape de réduction contient une étape finale qui peut ajouter la taille du discriminant à la taille de la matrice.

La réduction dans le cas imaginaire ne contient pas cette dernière étape, mais a de fortes similitudes avec la réduction du cas réel : il semble donc probable que la majoration de la taille de la matrice de réduction du cas imaginaire et du cas réel soit du même ordre.

Notre majoration dans le cas réel étant plus petite que la meilleure majoration du cas imaginaire de [4], nous avons amélioré la majoration de la taille des matrices de réduction des formes imaginaires (théorème 4.1), cette majoration est du même ordre que notre majoration dans le cas réel. Notre théorème 4.1 a les mêmes hypothèses que celles de [4].

Pour compléter ce travail, nous avons étudié la taille des matrices de passages entre deux formes réelles : il existe une infinité de matrice de passage entre deux formes réelles données, alors qu'il en existe au plus 6 pour deux formes imaginaires données. Plus particulièrement nous avons caractérisé la taille des plus petites matrices de passage entre deux formes données (théorème 4.4), et introduit une notion de distance (définition 4.5) entre deux formes, qui à l'inverse de la distance de Shanks [38] (communément utilisée [8, 3, 13, 24, 44]) vérifie à la fois l'inégalité triangulaire, mais aussi la condition qui impose à une distance d'être positive.

Nous avons également majoré les puissances entière de la plus petite transformation non triviale qui laisse une forme réelle inchangée (*automorphisme fondamental*). Cette majoration est donnée par la proposition 3.7

Nos majorations, citées ci-dessus, combinées avec notre amélioration de la méthode de [10] (algorithme ***Rationel Boneh-Durfee-HowgraveGraham***), nous permettent de prouver (sans hypothèses heuristiques) que notre variante de l'attaque des cryptosystèmes *NICE* fonctionne efficacement : la factorisation de la clé publique est trouvée par notre algorithme en temps polynomial.

Ce manuscrit suit le plan suivant :

Dans une première partie nous introduisons les notions élémentaires concernant à la fois les formes quadratiques binaires, mais aussi la cryptologie, et la notion de complexité.

Cela nous permet d'avoir les connaissances suffisantes afin d'aborder plus en détails l'étude des classes de formes et des automorphismes de formes, ainsi que l'étude des cryptosystèmes à base de formes qui sont respectivement les parties 2 et 3.

Dans la deuxième partie nous faisons une étude approfondie des classes de formes et nous majorons la taille d'une puissance entière de l'automorphisme fondamental, avant d'introduire nos résultats relatifs aux matrices de transformation : à la fois sur la majoration de la taille des matrices de réduction dans le cas imaginaire et le cas réel, mais aussi sur la taille des plus petites matrices de passage.

La troisième partie est consacrée à l'étude des cryptosystèmes à base de formes. Puisque ces cryptosystèmes utilisent deux particularités des classes de formes, nous commençons par étudier ces deux particularités : la loi $\star$ du groupe de classes, et la fonction ***Lift***. De plus nous introduirons ce que l'on a appelé la q-ceinture (section 5.2.5) qui nous permet de mieux visualiser certaines formes. Dans le dernier chapitre nous mettons en application nos résultats du chapitre 4 pour prouver que notre amélioration de l'attaque

des cryptosystèmes *NICE* fonctionne en complexité polynomiale en la taille de la clé publique.

Première partie

Préliminaires

Chapitre 1

Formes quadratiques binaires

Ce chapitre énonce les définitions, notions et propriétés générales concernant les formes quadratiques binaires. Les références de ce chapitre sont [9, 8, 13, 31, 15].

1.1 Définitions et propriétés élémentaires

1.1.1 Formes quadratiques binaires

Nous commençons par définir les objets mathématiques que l'on étudie dans cette thèse : les formes quadratiques binaires.

Définition 1.1 *Une forme quadratique binaire est un polynôme homogène de degré 2 en deux variables*

$$f(x,y) = ax^2 + bxy + cy^2 \text{ avec } (a,b,c) \in \mathbb{Z}^3,$$

que l'on pourra abréger en $f = (a,b,c)$ *ou juste* (a,b,c).

Les coefficients a, b et c d'une forme $f = (a,b,c)$ pourront aussi être appelés respectivement : le premier, le deuxième et le troisième coefficient de f.

Dans ce manuscrit le mot *forme* sera utilisé dans le sens de forme quadratique binaire.

Nous nous intéresserons plus particulièrement aux formes dont les coefficients vérifient que leur plus grand commun diviseur est 1.

Définition 1.2 *Une forme* $f = (a,b,c)$ *est dite primitive lorsque*

$$\mathrm{pgcd}(a,b,c) = 1,$$

et on abrégera $\mathrm{pgcd}(a,b,c)$ *par* $\mathrm{pgcd}(f)$.

Pour toute forme il existe un entier spécial, qu'on appelle son discriminant, et qui s'écrit avec les coefficients de la forme.

Définition 1.3 *Le discriminant d'une forme f, noté Δ_f, est l'entier*

$$\Delta_f = b^2 - 4ac.$$

Par définition le discriminant Δ_f d'une forme $f = (a, b, c)$ est forcément de même parité que b car il vérifie que $b^2 \equiv \Delta_f \pmod 2$.

Nous montrons la proposition suivante :

Proposition 1.1 *Un entier Δ est un discriminant de forme si et seulement si il est congru à 0 ou 1 modulo 4.*

Démonstration. Montrons le premier sens. Si un entier Δ_1 est le discriminant d'une forme (a, b, c) alors il s'écrit $\Delta_1 = b^2 - 4ac$. Il suit que Δ_1 vérifie que $b^2 \equiv \Delta_1 \pmod 4$. Or tous les entiers carrés modulo 4 sont forcément congru à 0 ou 1 modulo 4. Donc Δ_1 est congru à 0 ou 1 modulo 4.
On montre l'autre sens en observant que tous les entiers Δ_2 qui sont congru à 0 ou 1 modulo 4 sont forcément des discriminants de forme : $(1, 0, -\Delta_2/4)$ ou $(1, 1, (-\Delta_2+1)/4)$ suivant que le discriminant Δ_2 est congru à 0 ou 1 modulo 4. □

Puisque les discriminants de formes, sont exactement les entiers congrus à 0 ou 1 modulo 4, par la suite on pourra appeler un discriminant de forme juste *discriminant*.

Le discriminant suivant joue un rôle important lorsque l'on veut étudier seulement les formes primitives.

Définition 1.4 *Un discriminant Δ est dit fondamental lorsque toutes les formes de discriminant Δ sont nécessairement primitives.*

Nous utiliserons le lemme suivant pour définir algébriquement ce qu'est un discriminant fondamental.

Lemme 1.1 *Un discriminant fondamental est forcément un entier qui ne s'écrit pas sous la forme k^2 pour un entier $k \in \mathbb{Z}\backslash\{\pm 1\}$.*

Démonstration. Supposons qu'un discriminant Δ soit un carré parfait : $\Delta = k^2$, pour un entier $k \in \mathbb{Z}\backslash\{\pm 1\}$. Puisque dans ce cas la forme $(0, k, k)$ est de discriminant Δ mais n'est pas primitive on en déduit qu'un discriminant fondamental ne peut s'écrire que de deux façon : soit c'est un entier qui n'est pas un carré parfait, soit c'est un entier égal à 1. □

Nous montrons la proposition suivante :

Proposition 1.2 *Un discriminant Δ qui n'est pas un carré parfait est fondamental si et seulement si il n'existe pas de discriminant s'écrivant Δ/r^2 pour un entier $r \neq \pm 1$.*

Démonstration. Supposons que Δ_1 et Δ_2 sont des discriminants qui ne sont pas des carrés parfaits : d'après le lemme 1.1 ils peuvent donc potentiellement être des discriminants fondamentaux.
On montre le premier sens de l'équivalence. Toute forme f non primitive de discriminant Δ_1, s'écrit comme le produit d'un entier $t \neq \pm 1$ et d'une forme primitive g, ce qui entraîne que $\Delta_1 = t^2\Delta_g$. S'il n'existe pas de discriminant s'écrivant Δ_1/u^2 pour un entier $u \neq \pm 1$ alors toute forme de discriminant Δ_1 est forcément primitive, donc Δ_1 est un discriminant fondamental.
On montre le deuxième sens de l'équivalence. Si il existe un entier $s \neq \pm 1$ vérifiant que le discriminant Δ_2 s'écrit $\Delta_2 = s^2\Delta_h$ pour un discriminant Δ_h d'une forme $h = (a, b, c)$, alors il existe une forme $(s \cdot a, s \cdot b, s \cdot c)$ non primitive de discriminant Δ_2, donc Δ_2 n'est pas un discriminant fondamental. On en déduit que si le discriminant Δ_2 est fondamental alors il n'existe pas de discriminant s'écrivant Δ_2/v^2 pour un entier $v \neq \pm 1$. □

On remarquera que cette proposition entraîne que tout discriminant qui n'est pas un carré parfait, s'écrit comme le produit d'un entier au carré, et d'un discriminant fondamental.

Exemple 1.1 *Le discriminant $\Delta_1 = 4 \cdot 9 \cdot 11$ s'écrit $\Delta_1 = 9\overline{\Delta_1}$ avec $\overline{\Delta_1} = 44$ un discriminant fondamental. Et le discriminant $\Delta_2 = -9 \cdot 11$ s'écrit $\Delta_2 = 9\overline{\Delta_2}$ avec $\overline{\Delta_2} = -11$ un discriminant fondamental.*

On déduit de la proposition 1.2 le corollaire suivant :

Corollaire 1.1 *Un discriminant Δ qui n'est pas un carré parfait est fondamental si et seulement si :*
lorsque $\Delta \equiv 1 \pmod 4$: Δ n'a pas de facteur carré
lorsque $\Delta \equiv 0 \pmod 4$: $\Delta/4$ est congru à 2 ou 3 modulo 4, et n'a pas de facteur carré.

Démonstration. D'après la proposition 1.2 un discriminant Δ qui n'est pas un carré parfait est fondamental si et seulement il n'existe pas d'entier $t \neq \pm 1$ vérifiant que Δ/t^2 est un discriminant.
Montrons le cas où $\Delta \equiv 1 \pmod 4$. Si Δ n'a pas de facteur carré alors forcément il n'est pas divisible par un carré et est donc fondamental. Dans l'autre sens si Δ s'écrit $\Delta = s^2t$ pour un entier $s \neq \pm 1$ et un entier t (forcément congru à 1 modulo 4), alors Δ/s^2 est un discriminant, donc Δ n'est pas fondamental.
Montrons le cas où $\Delta \equiv 0 \pmod 4$: $\Delta = 4k$ pour un entier k non nul (car Δ n'est pas un carré parfait). Si $\Delta/4$ est congru à 2 ou 3 modulo 4, et n'a pas de facteur carré alors $\Delta/4$ n'est pas divisible par un carré et n'est pas un discriminant, donc Δ est un discriminant fondamental. Dans l'autre sens si $\Delta/4 = u^2v$ pour un entier $u \neq \pm 1$ et un entier v, et que u^2v est congru à 0 ou 1 modulo 4, alors $\Delta/(2u)^2$ est un discriminant, donc Δ n'est pas fondamental. Et enfin si $u = \pm 1$ et que v est congru à 1 modulo 4, le premier

cas de ce corollaire nous dit que $\Delta/4$ est fondamental si et seulement si $\Delta/4$ n'a pas de facteur carré. □

1.1.2 Propriétés des formes selon leur discriminant

Étudions les formes en fonction de la nature de leur discriminant.

Formes minimales

Définition 1.5 *Les formes dont le discriminant est un carré parfait sont dites minimales.*

On remarquera que toutes les formes (a, b, c) de premier ou troisième coefficient nul sont forcément minimales, puisque elles sont de discriminant égal à b^2.

La réciproque n'est en général pas vraie : les formes qui s'écrivent $f(x) = (x - ry)^2$ pour un entier $r \in \mathbb{Z}^*$ sont minimales (de discriminant égal à 0) et leur premier et troisième coefficients sont non nuls.

Formes réelles et imaginaires

Regardons les différentes propriétés d'une forme dans le cas où son discriminant n'est pas un carré parfait.

Puisque le discriminant Δ_f de la forme $f = (a, b, c)$ n'est pas un carré, on a $ac \neq 0$ et donc le polynôme univarié $f(x, 1)$ a deux racines complexes distinctes.

Définition 1.6 *On définit la représentation affine d'une forme dont le discriminant n'est pas un carré parfait comme étant la représentation graphique de la fonction* $\begin{array}{ccc} \mathbb{R} & \to & \mathbb{R} \\ x & \mapsto & f(x,1) \end{array}$ *: une parabole que l'on note $p_f(x)$. Le polynôme univarié $f(x, 1)$ a deux racines complexes que l'on note ζ_f^- et ζ_f^+.*

Il est utile de connaître la forme factorisée d'une forme.

Propriété 1.1 *Une forme $f = (a, b, c)$ dont le discriminant n'est pas un carré parfait se met sous la forme factorisée :*

$$f(x, y) = a(x - y\zeta_f^-)(x - y\zeta_f^+). \tag{1.1}$$

Comme l'entier c se calcule à partir des entiers $a \neq 0$, b et de la formule du discriminant, lorsque le discriminant est connu on pourra aussi abréger la forme par $f = (a, b, *)$, ou seulement $(a, *, *)$ si ses deuxième et troisième coefficients sont inconnus ou non pertinents.

On peut aussi vérifier la propriété suivante :

Propriété 1.2 *Une forme $f = (a, b, c)$ vérifie :*

$$4af(x, y) = (2ax + by)^2 - \Delta_f y^2. \tag{1.2}$$

Nous distinguons deux cas : suivant le signe du discriminant Δ_f.

Dans le cas où $\Delta_f > 0$, les racines de f sont dans $\mathbb{R}\backslash\mathbb{Q}$.

Définition 1.7 *Une forme de discriminant non carré parfait et positif est dite réelle.*

Dans le cas d'une forme réelle on distinguera sa plus petite racine et sa plus grande racine.

Définition 1.8 *Pour une forme réelle f sa racine ζ_f^- est définie comme étant sa plus petite racine, et ζ_f^+ sa plus grande racine.*

Les racines d'une forme réelle vérifient la propriété suivante.

Propriété 1.3 *Les racines ζ_f^- et ζ_f^+ d'une forme réelle $f = (a, b, c)$ s'écrivent :*

$$\zeta_f^- = \frac{-b - \text{signe}(a)\sqrt{\Delta_f}}{2a} \text{ et } \zeta_f^+ = \frac{-b + \text{signe}(a)\sqrt{\Delta_f})}{2a}.$$

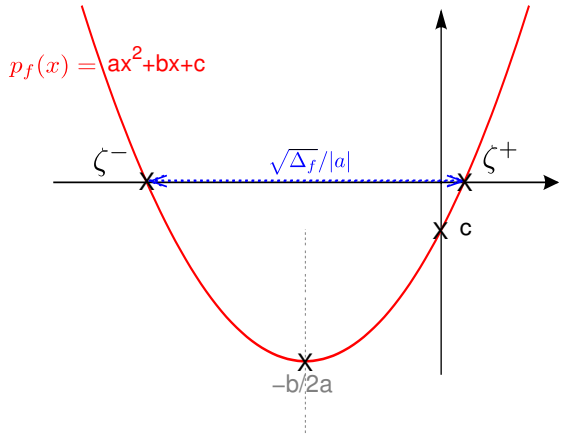


FIG. 1.1 – *Représentation affine $p_f(x)$ d'une forme réelle f*

Cette figure illustre la parabole $p_f(x)$ associée à une forme réelle $f = (a, b, c)$. On distingue les deux racines ζ_f^- et ζ_f^+, la distance entre les racines $\sqrt{\Delta_f}/|a|$ et la moyenne des deux racines $-b/2a$.

Dans le cas où $\Delta_f < 0$, les racines de $f = (a, b, c)$ sont dans $\mathbb{C}\backslash\mathbb{R}$.

Définition 1.9 *Une forme de discriminant non carré parfait et négatif est dite imaginaire.*

De plus puisque dans ce cas $-b^2 + 4ac > 0$, alors les coefficients a et c sont de même signe.

L'écriture 1.2 implique que tous les entiers qui s'écrivent $f(x, y)$ pour $(x, y) \in \mathbb{Z}^2$ sont du signe de a, et donc la forme est dite *positive* si $a > 0$, sinon *négative*.

Par la suite on étudiera l'ensemble de toutes les formes primitives qui ont le même discriminant : un discriminant non carré parfait.

Définition 1.10 $\mathfrak{F}_\Delta$ *est l'ensemble des formes de discriminant* Δ *qui sont primitives, avec* Δ *un discriminant non carré parfait.*

On insiste sur le fait que l'ensemble $\mathfrak{F}_\Delta$ n'est définit que pour des discriminants Δ qui ne sont pas des carrés parfaits.

1.1.3 Minimum et surface d'une forme

Nous introduisons la notion de premier minimum d'une forme.

Définition 1.11 *Le premier minimum d'une forme* f *est :*

$$\lambda(f) = \min\left\{|f(x, y)| : (x, y) \in \mathbb{Z}^2 \backslash (0, 0)\right\}.$$

De premier abord, il n'est pas évident que l'on puisse trouver de façon efficace le premier minimum d'une forme.

La proposition suivante, permet sans calcul de définir le premier minimum d'une forme minimale.

Proposition 1.3 *Le premier minimum d'une forme est nul si et seulement si la forme est minimale.*

Démonstration. Soit une forme $f = (a, b, c)$ de discriminant Δ_f.
Commençons par prouver le premier sens de cette équivalence, on suppose que le discriminant s'écrit $\Delta_f = k^2$ pour un entier k, si $a = 0$ alors $f(x, y) = y(bx + cy)$ et nous avons bien $\lambda(f) = 0$, sinon $4af(x, y) = (2ax - y(b + k))(2ax - y(b - k))$ et $f(b + k, 2a) = 0 = f(b - k, 2a)$ prouve que le premier minimum de la forme f est $\lambda(f) = 0$ puisque a n'est pas nul.
Montrons l'équivalence dans l'autre sens, on suppose que $\lambda(f) = 0 = f(x', y')$ pour $(x', y') \in \mathbb{Z}^2 \backslash (0, 0)$, si $a = 0$ alors $\Delta_f = b^2$, sinon $y' \neq 0$ car $f(x', 0) = ax'^2$ contredit l'hypothèse $\lambda(f) = 0$, et donc le discriminant est bien un carré parfait $\Delta_f = \left({}^{(2ax' - by')}/_{y'}\right)^2$ (propriété 1.2). □

La notion suivante est logiquement reliée au premier minimum d'une forme.

Définition 1.12 *Un entier n est représenté par la forme f s'il s'écrit $n = f(x, y)$ pour un couple d'entier (x, y).*

Pour comprendre les différences entre les formes imaginaires et réelles, il est intéressant d'introduire la notion de surface d'une forme.

Définition 1.13 *La surface d'une forme f est :*

$$s_f(x, y) = \big\{(x, y, z) \in \mathbb{R}^3 : z = f(x, y)\big\}.$$

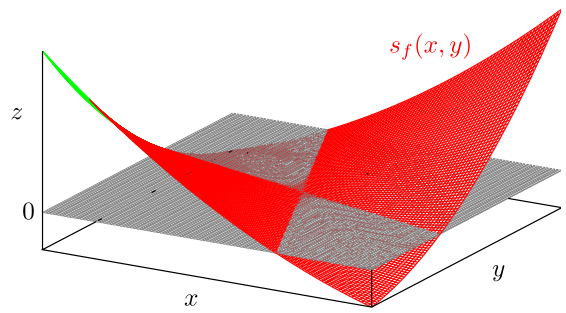


FIG. 1.2 – *Surface $s_f(x, y)$ d'une forme réelle f*

Cette figure illustre la surface associée à une forme réelle f. Les entiers représentés par la forme ne sont pas tous du même signe.

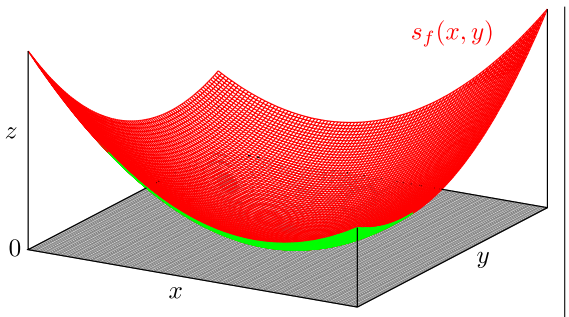


FIG. 1.3 – *Surface $s_f(x, y)$ d'une forme imaginaire f*

Cette figure illustre la surface associée à une forme imaginaire positive f. Les entiers représentés par la forme sont tous positifs.

1.1.4 Représentation polaire et action de composition

Une matrice $\mathcal{M} = \begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix}$ pourra être abrégée par $\mathcal{M} = (\alpha, \beta; \gamma, \delta)$, et sa transposée est notée $\mathcal{M}^t$.

Définition 1.14 *La représentation polaire d'une forme $f = (a, b, c)$ est notée $\mathcal{M}_f$ et c'est la matrice symétrique $\begin{pmatrix} a & b/2 \\ b/2 & c \end{pmatrix}$ de déterminant $-\Delta_f/4$.*

L'ensemble des matrices 2×2 à coefficients entiers est noté $\mathcal{M}_2(\mathbb{Z})$, et la matrice identité de $\mathcal{M}_2(\mathbb{Z})$ est notée $\mathcal{I}_2$.

Nous définissons l'action des matrices de $\mathcal{M}_2(\mathbb{Z})$ sur l'ensemble des formes, et l'on remarquera bien que cette action ne préserve pas le discriminant des formes, et n'est pas une action de groupe.

Définition 1.15 *L'action de composition d'une matrice $\mathcal{M} = (\alpha, \beta; \gamma, \delta) \in \mathcal{M}_2(\mathbb{Z})$ sur une forme f est la forme $g(x, y) = f(\alpha x + \beta y, \gamma x + \delta y)$. Cette action est notée $f.\mathcal{M}$ et la forme g a pour représentation polaire $\mathcal{M}^t \mathcal{M}_f \mathcal{M}$.*

L'action de composition est caractérisée par les quatre propriétés suivantes.

Propriété 1.4 *Si l'action de composition s'écrit $g = f.\mathcal{M}$ pour une matrice $\mathcal{M} = \begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} \in \mathcal{M}_2(\mathbb{Z})$ et deux formes f et g alors :*

1) $g = (f(\alpha, \gamma), b(\alpha\delta + \gamma\beta) + 2(a\alpha\beta + c\gamma\delta), f(\beta, \delta))$

2) $f.\mathcal{M} = f.(-\mathcal{M})$

3) $\Delta_g = \det(\mathcal{M})^2 \Delta_f$

4) si $\det(\mathcal{M}) \neq 0$ *et que f et g sont imaginaires ou réelles alors pour chaque racine ζ_g de g,* $\left(\dfrac{\alpha\zeta_g + \beta}{\gamma\zeta_g + \delta}\right)$ *est une racine de f.*

Démonstration. Soit $g = (a_g, b_g, c_g)$ la forme de la propriété 1.4.
Le premier point s'obtient simplement en développant $f(\alpha x + \beta y, \gamma x + \delta y)$ et entraîne le deuxième.
Le troisième point vient du fait que la représentation polaire de g est $\mathcal{M}^t \mathcal{M}_f \mathcal{M}$. Dans le cas où $\det(\mathcal{M}) \neq 0$ (la matrice $\mathcal{M}$ est inversible) la fonction qui associe à une racine de g une racine de f est une fonction homographique de $\mathbb{C}$ dans $\mathbb{C}$, on obtient le quatrième point en développant la fraction $(\delta\zeta_f - \beta)/(-\gamma\zeta_f + \alpha)$ qui est égale à $(-b_g \pm \det(M)\sqrt{\Delta_f})/(2a_g)$. □

Pour simplifier les notations, on étend la fonction racine carrée $\sqrt{\cdot}$ à $\mathbb{R}$ tout entier : à tout $x \in \mathbb{R}^-$ elle associe le complexe $i\sqrt{|x|}$.

Définition 1.16 *La matrice de l'action de composition sur une forme pourra aussi être appelée matrice de passage.*

1.1.5 Normes de matrices et normes de formes

Soit $\mathcal{M} = (\alpha, \beta; \gamma, \delta)$ une matrice de $\mathcal{M}_2(\mathbb{Z})$.

La *norme Euclidienne* est $\|\mathcal{M}\|_2 = \sqrt{\alpha^2 + \beta^2 + \gamma^2 + \delta^2}$, et la *norme maximale* est $\|\mathcal{M}\| = \max(|\alpha|, |\beta|, |\gamma|, |\delta|)$.

La norme $\|\!|\mathcal{M}|\!\| = \sup_{\|\vec{v}\|_2 = 1}(\|\mathcal{M}.\vec{v}\|_2)$ est la *norme Euclidienne induite* (un résumé sur les propriétés de cette norme est énoncé dans [34] et [21]), qui est aussi la racine carrée de la plus grande des valeurs propres de $\mathcal{M}^t\mathcal{M}$.

Toutes ces normes sont équivalentes :

Propriété 1.5 *Pour toute matrice $\mathcal{M} \in \mathcal{M}_2(\mathbb{Z})$ on a :*

$$\|\mathcal{M}\| \leqslant |||\mathcal{M}||| \leqslant \|\mathcal{M}\|_2 \leqslant 2\|\mathcal{M}\|.$$

Une propriété très importante de la norme Euclidienne induite est qu'elle est sous-multiplicative. De plus en utilisant [21] on peut la minorer par le *rayon spectral* $\rho(\mathcal{M})$, qui est la plus grande des valeurs propres en valeur absolue de $\mathcal{M}$.

Propriété 1.6 *La norme Euclidienne induite d'une matrice $\mathcal{M} \in \mathcal{M}_2(\mathbb{Z})$ vérifie :*
- $\forall \mathcal{N} \in \mathcal{M}_2(\mathbb{Z})$, $|||\mathcal{M}\mathcal{N}||| \leqslant |||\mathcal{M}|||\,|||\mathcal{N}|||$,
- $|||\mathcal{I}_2||| = 1$,
- $|||\mathcal{M}||| \geqslant \rho(\mathcal{M})$.

Par extension, nous définissons les normes $\|f\|$,$\|f\|_2$ et $|||f|||$ d'une forme f comme la norme de sa représentation polaire correspondante.

Définition 1.17 *Pour toute forme f de représentation polaire $\mathcal{M}_f$:*

$$\|f\| = \|\mathcal{M}_f\|,\ \|f\|_2 = \|\mathcal{M}_f\|_2,\ |||f||| = |||\mathcal{M}_f|||.$$

1.2 Action des groupes GL_2 et SL_2

On note GL_2 le *groupe linéaire général* des matrices appartenant à $\mathcal{M}_2(\mathbb{Z})$ qui sont inversibles, et son sous-groupe SL_2 le *groupe linéaire spécial* des matrices de déterminant $+1$.

On se concentre sur l'action de composition restreint à l'action de composition des matrices de GL_2 (ou SL_2) (et non plus $\mathcal{M}_2(\mathbb{Z})$) sur l'ensemble des formes $\mathfrak{F}_\Delta$.

Cette action vérifie la proposition suivante.

Proposition 1.4 *L'action du groupe GL_2 (ou SL_2) sur l'ensemble $\mathfrak{F}_\Delta$ est une action de groupe à droite.*

Démonstration. Soit une matrice $\mathcal{M} = (\alpha, \beta; \gamma, \delta) \in \mathrm{GL}_2$ et une forme $f \in \mathfrak{F}_\Delta$.
En premier on montre que cette action vérifie :
1) $f.\mathcal{I}_2 = f$,
2) $\forall (\mathcal{U}, \mathcal{V}) \in \mathrm{GL}_2^2, (f.\mathcal{U}).\mathcal{V} = f.(\mathcal{U}\mathcal{V})$.
Le seul point à montrer est le deuxième point, et celui-ci est immédiat en utilisant la représentation polaire de l'action de composition et l'associativité du produit matriciel :
$\forall (\mathcal{U}, \mathcal{V}) \in \mathrm{GL}_2^2, \mathcal{V}^t(\mathcal{U}^t\mathcal{M}_f\mathcal{U})\mathcal{V} = (\mathcal{U}\mathcal{V})^t\mathcal{M}_f(\mathcal{U}\mathcal{V})$.
L'autre point à montrer estque l'action du groupe GL_2 sur l'ensemble $\mathfrak{F}_\Delta$ est une application de $\mathfrak{F}_\Delta \times \mathrm{GL}_2$ dans $\mathfrak{F}_\Delta$.

(Préservation du discriminant)
Par définition de l'action de composition 1.15 l'action de $\mathcal{M}$ sur f est la forme $g(x,y) = f(\alpha x + \beta y, \gamma x + \delta y)$ de discriminant Δ, puisque $\Delta_g = \det(\mathcal{M})^2\Delta$ et que $\det(\mathcal{M})^2 = 1$. Donc l'action des matrices du groupe GL_2 sur l'ensemble $\mathfrak{F}_\Delta$ et une forme de discriminant Δ.
(Préservation du caractère primitif)
La description des coefficients de $f.\mathcal{M}$ implique que $\mathrm{pgcd}(f)$ divise $\mathrm{pgcd}(f.\mathcal{M})$, et inversement $\mathrm{pgcd}(f.\mathcal{M})$ divise $\mathrm{pgcd}(f)$, car $\mathcal{M}$ étant dans GL_2 on peut écrire $(f.\mathcal{M}).\mathcal{M}^{-1} = f$.
Donc l'action des matrices du groupe GL_2 sur l'ensemble $\mathfrak{F}_\Delta$ est bien une forme de $\mathfrak{F}_\Delta$.
Pour finir on remarque que toutes ces propriétés restent vraies si l'on remplace GL_2 par son sous-groupe SL_2. □

Par compatibilité avec le produit des matrices $f.(\mathcal{M}\mathcal{N})$ sera simplement écrit $f.\mathcal{M}\mathcal{N}$.

Il est important de mémoriser que cette action de groupe préserve à la fois le discriminant des formes, et la propriété primitive (ou non primitive) des formes sur lesquelles elle agit.

1.2.1 Transformations génératrices de GL_2 et SL_2

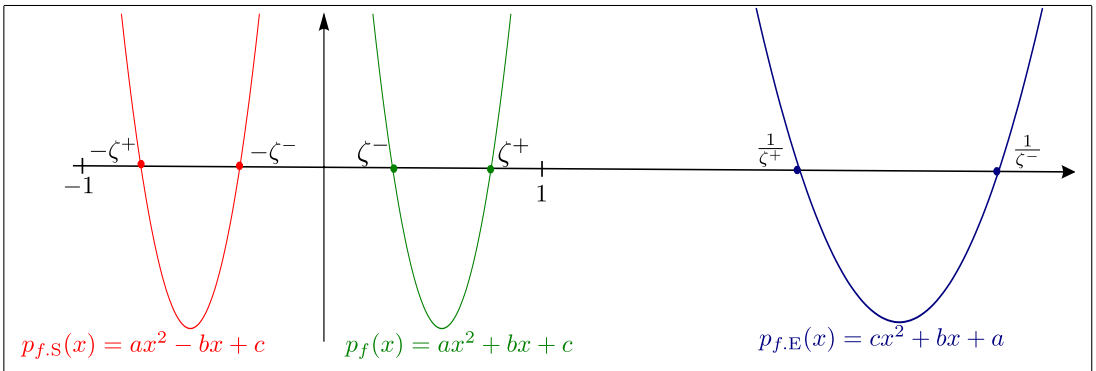


FIG. 1.4 – *Représentation affine des transformations* $\boldsymbol{S}$ *et* $\boldsymbol{E}$

Cette figure illustre les paraboles correspondantes à l'action de $\boldsymbol{S}$ *et* $\boldsymbol{E}$ *sur une forme réelle* f.

Nous fixons trois transformations (linéaires) de GL_2 particulières.

Définition 1.18 *La symétrie est* $\boldsymbol{S} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$, *l'échange est* $\boldsymbol{E} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ *et la translation par un entier* h *est* $\boldsymbol{T}(h) = \begin{pmatrix} 1 & h \\ 0 & 1 \end{pmatrix}$.

Ces trois transformations jouent un rôle important puisque à elles seules elles engendrent le groupe GL_2. Plus précisément nous avons le théorème suivant.

Théorème 1.1 *Toutes les matrices de GL_2 s'écrivent comme un produit de puissance de $\boldsymbol{S}$, $\boldsymbol{E}$ et $\boldsymbol{T}(1)$. Son sous-groupe SL_2 est quant à lui généré par le produit $\boldsymbol{SE}$ et $\boldsymbol{T}(1)$.*

On trouve la démonstration de ce théorème dans [43] et [1].

L'action de ces transformations sur une forme vérifie la propriété suivante.

Propriété 1.7 *Pour toute forme $f = (a, b, c)$ on a :*

$$\begin{aligned} f.\boldsymbol{S} &= (a, -b, c), \\ f.\boldsymbol{E} &= (c, b, a), \\ f.\boldsymbol{T}(h) &= (a, b + 2ah, p_f(h)). \end{aligned}$$

Cette propriété découle juste de la définition de la composition 1.15.

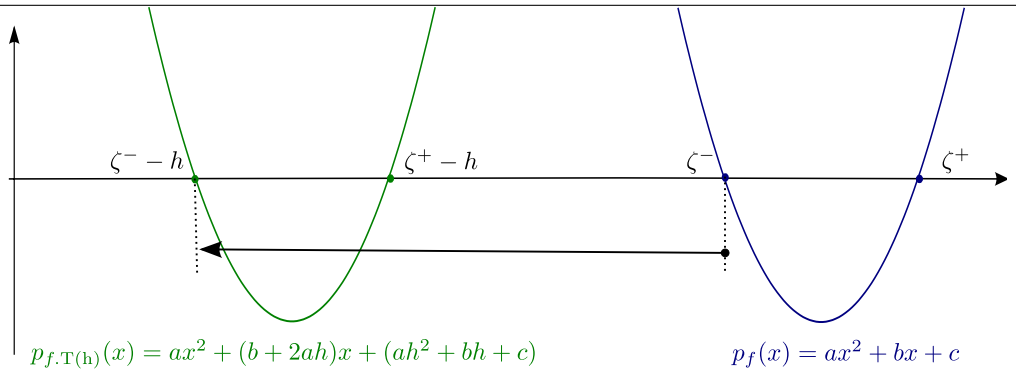


FIG. 1.5 – *Représentation affine de la translation $\boldsymbol{T}(h)$ pour un entier h*

Cette figure illustre la parabole correspondant à l'action de la translation $\boldsymbol{T}(h)$ pour un entier h, sur une forme réelle f.

On déduit de cette propriété, le résultat de l'action des ces matrices en fonction des racines des formes.

Propriété 1.8 *Pour toute forme $f = (a, b, c)$ imaginaire ou réelle :*

– les racines de $f.\boldsymbol{S}$ sont les opposées des racines f,
– les racines de $f.\boldsymbol{E}$ sont les inverses des racines de f,
– et $\boldsymbol{T}(h)$ soustrait h à chaque racine de f.

Il est utile de remarquer que la représentation affine d'une translation $\boldsymbol{T}(h)$ sur une forme f, revient soit à translater p_f par le vecteur de coordonnées $(-h, 0)$ soit à translater l'axe des ordonnées par le vecteur de coordonnées $(h, 0)$.

1.3 SL_2-Orbites

L'*orbite* d'un élément f de $\mathfrak{F}_\Delta$ est l'ensemble $\mathcal{O}_f = \{f.\mathcal{M} : \mathcal{M} \in \mathrm{SL}_2\}$.

On remarquera bien que l'orbite de f est l'orbite de l'action de SL_2, et non pas de GL_2, sur f.

Définition 1.19 *L'ensemble des orbites de l'action de SL_2 sur $\mathfrak{F}_\Delta$ est*

$$\mathcal{O}_{\mathfrak{F}_\Delta} = \{\mathcal{O}_f : f \in \mathfrak{F}_\Delta\}.$$

On remarquera que l'ensemble des orbites $\mathcal{O}_{\mathfrak{F}_\Delta}$ forme une partition de $\mathfrak{F}_\Delta$, car l'action de SL_2 sur $\mathfrak{F}_\Delta$ est une action de groupe.

Définissons l'équivalence entre deux formes de $\mathfrak{F}_\Delta$.

Définition 1.20 *Deux formes f et g sont équivalentes si elles sont dans la même SL_2-orbite, ce que l'on note $f \sim g$.*

Par définition, deux formes f et g sont donc équivalentes lorsque il existe une matrice $\mathcal{M} \in \mathrm{SL}_2$ vérifiant $f.\mathcal{M} = g$.

Dans ce cas on schématise la relation d'équivalence $f \sim g$ par :

$$f \xmapsto{\mathcal{M}} g \text{ ou plus simplement } f \longmapsto g.$$

Cette relation d'équivalence est bien réflexive, symétrique et transitive :

* $\forall f \in \mathfrak{F}_\Delta$, $f.\mathcal{I}_2 = f$:

$$f \xmapsto{\mathcal{I}_2} f$$

* $\forall (f,g) \in \mathfrak{F}_\Delta^2$, si $\exists \mathcal{M} \in \mathrm{SL}_2$ telle que $f.\mathcal{M} = g$ alors $f = g.\mathcal{M}^{-1}$ ($\mathcal{M}^{-1} \in \mathrm{SL}_2$) :

$$f \underset{\mathcal{M}^{-1}}{\overset{\mathcal{M}}{\rightleftarrows}} g$$

* $\forall (f,g,h) \in \mathfrak{F}_\Delta^3$, si $\exists (\mathcal{M},\mathcal{N}) \in \mathrm{SL}_2^2$ telles que $f.\mathcal{M} = g$ et $g.\mathcal{N} = h$ alors $f.\mathcal{M}\mathcal{N} = h$:

$$f \xmapsto{\mathcal{M}} g \xmapsto{\mathcal{N}} h \qquad f \xmapsto{\mathcal{M}\mathcal{N}} h$$

Définition 1.21 *La classe d'équivalence d'une forme f est*

$$\overline{f} = \{g : \exists \mathcal{M} \in SL_2 f.\mathcal{M} = g\}.$$

Notons le point important suivant.

Proposition 1.5 *Si deux formes sont équivalentes alors elles représentent les mêmes entiers.*

Démonstration. Supposons qu'il existe un couple d'entiers $(k,l) \in \mathbb{Z}^2$ vérifiant que $f(k,l) = t$ pour un entier t. Si les deux formes f et g sont équivalentes par la matrice de passage $\mathcal{M} = (\alpha, \beta; \gamma, \delta)$ alors on a $g(x,y) = f(\alpha x + \beta y, \gamma x + \delta y)$. Et comme $\mathcal{M}$ appartient à SL_2 on peut inverser la transformation $k = \alpha x + \beta y$ et $l = \gamma x + \delta y$ pour deux entiers x et y. Il existe donc deux entiers $(m,n) \in \mathbb{Z}^2$ vérifiant $k = \alpha m + \beta n$ et $l = \gamma m + \delta n$. Et puisque $g(m,n) = f(\alpha m + \beta n, \gamma m + \delta n)$ il vient $g(m,n) = f(k,l) = t$. □

Notons que l'inverse n'est en général pas vrai : un entier peut être représenté par deux formes dans deux classes différentes.

Exemple 1.2 *Les formes $f = (3,2,5)$ et $g = (3,-2,5)$ sont de discriminant -56 et ne sont pas équivalentes, pourtant elles représentent toutes les deux l'entier 59 : $f(3,2) = 59$ et $g(-3,2) = 59$.*
Nous verrons dans la section 3.1.1 qu'il existe une unique forme réduite par classe dans $\mathfrak{F}_{\Delta<0}$, et que les formes f et g sont toutes les deux réduites : elles ne sont donc pas équivalentes.

La proposition 1.5 implique en particulier que deux formes équivalentes ont le même premier minimum.

Proposition 1.6 *Pour toute forme $f = (a,b,c) \in \mathfrak{F}_\Delta$ et tout entier n non nul, la forme f est équivalente à une forme dont le premier coefficient est premier à n.*

Démonstration. Tout d'abord, puisque f est une forme primitive on a forcément $\mathrm{pgcd}(a, c, a+b+c) = 1$. Et comme $a = f(1,0)$, $c = f(0,1)$ et $a+b+c = f(1,1)$ alors pour chaque diviseur premier d de n il existe un couple d'entier $(\alpha_d, \gamma_d) \in \{(1,0),(0,1),(1,1)\}$ vérifiant que $\mathrm{pgcd}(f(\alpha_d, \gamma_d), d) = 1$. Pour finir, le théorème chinois assure l'existence d'un couple d'entier (α, γ) vérifiant $\alpha \equiv \alpha_d \pmod d$ et $\gamma \equiv \gamma_d \pmod d$ pour chaque diviseur premier d de n. En conclusion, il existe deux entiers u et v vérifiant $\alpha u - \gamma v = \mathrm{pgcd}(\alpha, \gamma)$, on a donc $f.(\alpha/\mathrm{pgcd}(\alpha,\gamma), v; \gamma/\mathrm{pgcd}(\alpha,\gamma), u)$ de premier coefficient un entier premier à n ($f(\alpha,\gamma) = \mathrm{pgcd}(\alpha,\gamma)^2 f(\alpha/\mathrm{pgcd}(\alpha,\gamma), \gamma/\mathrm{pgcd}(\alpha,\gamma))$). □

Pour illustrer la proposition1.6, nous cherchons une forme équivalente à $f = (2,3,5) \in \mathfrak{F}_{-31}$, dont le premier coefficient est premier à $n = 100 = 2 * 5 * (5+3+2) = 2^2 5^2$.
On a $\mathrm{pgcd}(f(0,1),2) = 1$ et $\mathrm{pgcd}(f(1,0),5) = 1$, et on vérifie que $\alpha = 6$ et $\gamma = 15$ satisfont $\alpha \equiv 0 \pmod 2$, $\alpha \equiv 1 \pmod 5$, $\gamma \equiv 1 \pmod 2$ et $\gamma \equiv 0 \pmod 5$. Le coefficient $f(2,5) = 163$ (on a $\mathrm{pgcd}(6,15) = 3$) est bien premier à n, et une forme recherchée est $f.(2,-1;5,-2)$.

Définition 1.22 *Le nombre de classes est noté B_Δ et c'est le nombre d'orbites de l'action du groupe SL_2 sur l'ensemble $\mathfrak{F}_\Delta$.*

1.4 SL_2-Stabilisateur

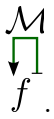
L'ensemble stabilisateur de l'action de SL_2 sur une forme $f \in \mathfrak{F}_\Delta$ est un groupe noté $Aut(f)$, on le définit formellement de la façon suivante.

Nous rappelons que l'ensemble $\mathfrak{F}_\Delta$ n'est définit que pour des discriminants Δ qui ne sont pas des carrés parfaits.

Définition 1.23 *Le groupe des automorphismes $Aut(f)$ d'une forme $f \in \mathfrak{F}_\Delta$ est le sous-groupe de SL_2 définit par l'ensemble*

$$Aut(f) = \{\mathcal{M} \in SL_2 : f.\mathcal{M} = f\}.$$

On schématise un automorphisme $\mathcal{M} \in \mathrm{SL}_2$ d'une forme f par :



1.4.1 Équation de Pell d'un discriminant

Une description plus précise de $Aut(f)$ pour une forme $f = (a, b, c) \in \mathfrak{F}_\Delta$ est obtenue en introduisant l'équation de Pell de Δ.

Définition 1.24 *L'équation de Pell d'un discriminant Δ est l'équation $x^2 - \Delta y^2 = 4$ pour tout couple d'entiers (x, y).*

On s'intéresse aux solutions de cette équation.

Dans la suite de ce manuscrit, l'ensemble des solutions de l'équation de Pell d'un discriminant Δ est noté $\mathcal{P}_\Delta$.

On note $\dagger$ l'application qui à deux solutions (x, y) et (z, t) de $\mathcal{P}_\Delta$ associe $\left(\dfrac{xz + \Delta_f yt}{2}, \dfrac{xt + yz}{2}\right)$.

Propriété 1.9 *L'ensemble $\mathcal{P}_\Delta$ pour un discriminant Δ muni de la loi $\dagger$ forme un groupe.*

Cette propriété se vérifie aisément. L'élément neutre de $(\mathcal{P}_\Delta, \dagger)$ est le couple $(2, 0)$, et l'inverse de toute solution $(x, y) \in \mathcal{P}_\Delta$ est $(x, -y)$.

Proposition 1.7 *Le groupe des automorphismes $Aut(f)$ d'une forme primitive $f \in \mathfrak{F}_\Delta$ est isomorphe au groupe $\mathcal{P}_\Delta$.*

La démonstration de l'isomorphisme $Aut(f) \cong \mathcal{P}_\Delta$ pour une forme primitive $f \in \mathfrak{F}_\Delta$ se trouve dans [8] (théorème 2.5.5 et 2.5.7), et [9] (théorème 3.9).

Proposition 1.8 *Tout automorphisme d'une forme primitive $(a, b, c) \in \mathfrak{F}_\Delta$ s'écrit*

$$\mathcal{A}_{f,(x,y)} = \begin{pmatrix} (x - by)/2 & -cy \\ ay & (x + by)/2 \end{pmatrix}$$

avec $(x, y) \in \mathcal{P}_\Delta$.

La démonstration de cette proposition se trouve également dans [8] (théorème 2.5.5).

Cette proposition nous permet d'écrire un automorphisme d'une forme $f \in \mathfrak{F}_\Delta$ à partir d'une solution de $\mathcal{P}_\Delta$.

Pour éviter les confusions nous faisons remarquer qu'en prenant deux entiers x et y qui vérifient $x^2 - \Delta y^2 = -4$ la matrice $((x - by)/2, -cy; ay, (x + by)/2)$ est dans GL_2 et de déterminant -1, ce n'est donc pas un automorphisme de f.

Les théorèmes fondamentaux sur le groupe des automorphismes d'une forme de $\mathfrak{F}_\Delta$ sont décris dans les parties suivantes.

1.4.2 Groupe des automorphismes d'une forme imaginaire

Théorème 1.2 *Le groupe des automorphismes $Aut(f)$ d'une forme imaginaire $f \in \mathfrak{F}_\Delta$ est un groupe cyclique.*

Démonstration. On montre en premier que l'ensemble des solutions $\mathcal{P}_\Delta$ est une groupe cyclique. L'équation de Pell de discriminant $\Delta < 0$ s'écrit $x^2 + |\Delta|y^2 = 4$ pour tout couple d'entiers (x, y). Toute solution $(x, y) \in \mathcal{P}_\Delta$ vérifie $|x| \leqslant 2$ (car $0 \leqslant |\Delta|y^2 = 4 - x^2$) elle vérifie donc aussi $|\Delta|y^2 \leqslant 4$. On en déduit que :
– $\mathcal{P}_{-3} = \{(\pm 2, 0), (\pm 1, \pm 1)\}$,
– $\mathcal{P}_{-4} = \{(\pm 2, 0), (0, \pm 1)\}$,
– $\mathcal{P}_{\Delta<-4} = \{(\pm 2, 0)\}$.
On vérifie que $\mathcal{P}_{-3}$ est cyclique d'ordre 6 de générateur $(1, 1)$, $\mathcal{P}_{-4}$ est cyclique d'ordre 4 de générateur $(0, 1)$ et enfin $\mathcal{P}_{\Delta<-4}$ est le groupe d'ordre 2 de générateur $(-2, 0)$.
L'isomorphisme $Aut(f) \cong \mathcal{P}_\Delta$ pour toute forme primitive $f \in \mathfrak{F}_\Delta$ (proposition 1.7) permet de conclure. □

1.4.3 Groupe des automorphismes d'une forme réelle

Dans le cas d'une forme réelle $f \in \mathfrak{F}_\Delta$ c'est beaucoup plus compliqué, car la solution dans $\mathcal{P}_\Delta$ de plus petite taille est généralement de taille exponentielle en la taille du discriminant Δ.

Pour étudier les automorphismes de f on distingue les automorphismes de trace positive, et ceux de trace négative. Rappelons que la trace d'une matrice carrée est la somme de ses coefficients diagonaux.

Définition 1.25 *L'ensemble des automorphismes de trace positive d'une forme $f \in \mathfrak{F}_\Delta$ est $Aut^+(f) = \{\mathcal{M} \in Aut(f) : \text{trace}(\mathcal{M}) > 0\}$, et l'ensemble de ses automorphismes de trace négative est $Aut^-(f) = \{\mathcal{M} \in Aut(f) : \text{trace}(\mathcal{M}) < 0\}$.*

Nous obtenons la relation suivante.

Propriété 1.10 *Pour toute forme réelle $f \in \mathfrak{F}_\Delta$ on a : $Aut(f) = Aut^+(f) \bigcup Aut^-(f)$.*

Cette propriété découle directement des propositions 1.7 et 1.8.

De la même manière nous distinguons les solutions (x, y) de $\mathcal{P}_\Delta$ avec $x > 0$.

Définition 1.26 *On définit l'ensemble $\mathcal{P}_\Delta^+ = \{(x, y) \in \mathcal{P}_\Delta : x > 0\}$ pour un discriminant Δ.*

Cet ensemble est bien un groupe.

Propriété 1.11 *L'ensemble $\mathcal{P}_\Delta^+$ est un groupe pour la loi $\dagger$.*

Démonstration. Puisque que $\mathcal{P}_\Delta$ est un groupe pour la loi $\dagger$ (propriété 1.9), pour démontrer la propriété il suffit de prouver que la loi $\dagger$ est une loi interne. Soient deux solutions (x, y) et (z, t) de $\mathcal{P}_\Delta^+$ on a $0 < x^2 = 4 + \Delta y^2$ et $0 < z^2 = 4 + \Delta t^2$, ce qui entraîne que $(xz)^2 = 16 + 4\Delta(t^2 + y^2) + (yt\Delta)^2$ et donc que $xz > \Delta|yt|$. La loi $\dagger$ est donc bien une loi interne. □

D'après la propriété 1.10 pour étudier la structure de $Aut(f)$ il est important de connaître la structure de $Aut^+(f)$. C'est pour cela qu'on introduit le théorème suivant :

Théorème 1.3 *Le groupe des automorphismes $Aut^+(f)$ d'une forme réelle $f \in \mathfrak{F}_\Delta$ est un groupe monogène.*

Dans les références de ce chapitre, les démonstrations de ce théorème se font à base de réseaux ou bien d'idéaux. Aussi il nous a paru intéressant d'alléger cette démonstration en la réécrivant en n'utilisant que les solutions d'une équation de Pell.

Avant de faire la démonstration de ce théorème, nous étudions la fonction

$$\begin{aligned} \psi : \mathcal{P}_\Delta^+ &\longrightarrow \mathbb{R}_+^* \\ (x, y) &\longmapsto \frac{x + y\sqrt{\Delta}}{2} \end{aligned}$$

qui est définie pour tout discriminant Δ positif et qui n'est pas un carré parfait.

On montre que la fonction ψ est bien définie pour tout couple de $\mathcal{P}^+$. Remarquons que toute solution (x, y) de $\mathcal{P}_\Delta^+$ vérifie $x^2 = 4 + \Delta y^2 > \Delta y^2$ et donc $x > \sqrt{\Delta}|y|$. Ainsi si $y \geqslant 0$, $x + y\sqrt{\Delta}$ est bien > 0 car $x > 0$ par définition de $\mathcal{P}_\Delta^+$, et si $y < 0$ on a $x + y\sqrt{\Delta} = x - |y|\sqrt{\Delta} > 0$ d'après la remarque précédente.

On étudie cette fonction pour démontrer que $\mathcal{P}_\Delta^+$ est un groupe monogène.

Lemme 1.2 *L'application ψ est un morphisme injectif, et $\mathrm{Im}(\psi)$ est un sous groupe monogène de $\mathbb{R}_+^*$.*

Démonstration. On vérifie facilement que ψ est un morphisme multiplicatif. Pour montrer que ψ est injectif il suffit de dire que $\sqrt{\Delta}$ est irrationnel. Nous avons donc montré que $\mathrm{Im}(\psi)$ est un sous-groupe multiplicatif de $\mathbb{R}_+^*$.
On considère l'image réciproque de l'intervalle $]1, +\infty[$ par ψ, et on remarque que c'est forcément l'ensemble des solutions de $\mathcal{P}^+$ dont les deux coordonnées sont positives. Car pour toute solution $(x, y) \in \mathcal{P}^+$, x est un entier non nul toujours positif et $1/\psi(x, y) = \psi(x, -y)$: si $\psi(x, -|y|) > 1$ alors $\psi(x, |y|) < 1$ contredit l'hypothèse $\psi(x, |y|) \geqslant \psi(x, -|y|)$.
Comme $(1, *)$ ne peut pas être solution de $\mathcal{P}_\Delta^+$, et que $(2, 0)$ est la solution triviale de $\mathcal{P}_\Delta^+$, on en déduit que pour toute solution non triviale (x, y) de $\mathcal{P}_\Delta^+$ avec $y > 0$, $\psi(x, y) \geqslant (2 + \sqrt{\Delta})/2$.
Donc $\mathrm{Im}(\psi) \cap]1, (2 + \sqrt{\Delta})/2[= \varnothing$, ce qui fait que l'image de ψ n'est pas dense dans $\mathbb{R}$ et donc $\mathrm{Im}(\psi)$ est un sous groupe monogène de $\mathbb{R}^{+*}$. □

On vient de montrer que $\mathrm{Im}(\psi)$ est un sous groupe monogène de $\mathbb{R}_+^*$, et que son générateur vérifie la proposition suivante.

Proposition 1.9 *Le plus petit élément qui engendre $\mathrm{Im}(\psi)$ et qui est plus grand que 1 s'écrit $\epsilon_\Delta = (x_\epsilon + y_\epsilon\sqrt{\Delta})/2$ avec $y_\epsilon > 0$ et $(x_\epsilon, y_\epsilon) \in \mathcal{P}_\Delta^+$, et c'est l'unité fondamentale.*

On a un isomorphisme naturel entre $\mathrm{Im}(\psi)$ et $\mathcal{P}_\Delta^+$, le groupe $\mathcal{P}_\Delta^+$ est donc monogène grâce au lemme 1.2, et engendré par $\psi^{-1}(\epsilon_\Delta) \in \mathcal{P}_\Delta^+$.

Définition 1.27 *Le groupe monogène $\mathcal{P}_\Delta^+$ est engendré par la solution fondamentale $(x_\epsilon, y_\epsilon) = \psi^{-1}(\epsilon_\Delta)$.*

La démonstration du théorème 1.3 se résume donc à :

Démonstration. Tout d'abord puisque $Aut(f) \cong \mathcal{P}_\Delta$ (proposition 1.7), on vérifie facilement que $Aut^+(f) \cong \mathcal{P}_\Delta^+$. Et puisque $\mathcal{P}_\Delta^+$ est monogène et engendré par (x_ϵ, y_ϵ) alors $Aut^+(f)$ est monogène et engendré par l'automorphisme construit à partir de la solution fondamentale (x_ϵ, y_ϵ). □

Définissons le générateur du groupe monogène $Aut^+(f)$.

Définition 1.28 *Le groupe monogène $Aut^+(f)$ d'une forme réelle $f \in \mathfrak{F}_\Delta$ est engendré par l'automorphisme fondamental $\mathcal{U}_f$ qui est l'automorphisme construit à partir de la solution fondamentale.*

Les automorphismes de trace négative étant juste les opposés des automorphismes de $Aut^+(f)$, on peut également conclure que :

Proposition 1.10 *Le groupe des automorphismes d'une forme réelle $f \in \mathfrak{F}_\Delta$ est $Aut(f) = \{\pm\mathcal{U}_f^i : i \in \mathbb{Z}\}$ avec $\mathcal{U}_f$ l'automorphisme fondamental de la forme f.*

En conclusion tout automorphisme d'une forme réelle $f \in \mathfrak{F}_\Delta$ se construit toujours à partir d'une puissance de la solution fondamentale (x_ϵ, y_ϵ).

Nous verrons dans la section 3.5 comment calculer en pratique la solution fondamentale (x_ϵ, y_ϵ), et donc en déduire l'automorphisme fondamental.

Il est important de noter que pour toute forme réelle $f = (a, b, c) \in \mathfrak{F}_\Delta$ l'unité fondamentale ϵ_Δ ne dépend que du discriminant Δ, et que c'est la plus grande valeur propre de l'automorphisme fondamental.

Propriété 1.12 *L'automorphisme fondamental d'une forme réelle $f \in \mathfrak{F}_\Delta$ a deux valeurs propres : ϵ_Δ qui est l'unité fondamentale et son inverse ϵ_Δ^{-1}.*

Démonstration. L'automorphisme fondamental de la forme f s'écrit $\mathcal{U}_f = \begin{pmatrix} (x_\epsilon - by_\epsilon)/2 & -cy_\epsilon \\ ay_\epsilon & (x_\epsilon + by_\epsilon)/2 \end{pmatrix}$. Le polynôme caractéristique de $\mathcal{U}_f$ est $P_{\mathcal{U}_f}(X) = X^2 - \text{trace}(\mathcal{U}_f)X + \det(\mathcal{U}_f) = X^2 - x_\epsilon X + 1$. Il est de discriminant $x_\epsilon^2 - 4 = \Delta y_\epsilon > 0$ car $(x_\epsilon, y_\epsilon) \in \mathcal{P}_\Delta^+$. Les deux valeurs propres de $\mathcal{U}_f$ sont $\epsilon_\Delta = \frac{x_\epsilon + y_\epsilon\sqrt{\Delta}}{2}$ et $\epsilon_\Delta^{-1} = \frac{x_\epsilon - y_\epsilon\sqrt{\Delta}}{2}$. □

Nous verrons dans la section 3.5 comment calculer en pratique l'automorphisme fondamental $\mathcal{U}_f$ d'une forme réelle f.

Dans la section 3.5.2 nous donnerons une majoration d'une puissance entière de $\mathcal{U}_f$.

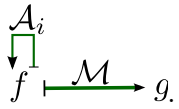
1.5 SL_2-Matrices de passage et automorphisme de forme

Le groupe des automorphismes d'une forme $f \in \mathfrak{F}_\Delta$ détermine l'ensemble de toutes les matrices de passage de f à une forme qui lui est équivalente g.

Proposition 1.11 *Si $\{\mathcal{A}_i : i \in \mathbb{Z}\}$ est l'ensemble de tous les automorphismes d'une forme $f \in \mathfrak{F}_\Delta$ et $\mathcal{M} \in SL_2$ une matrice de passage de f à une forme g, alors l'ensemble de toutes les matrices de passage de f à g est $\{\mathcal{A}_i\mathcal{M} : i \in \mathbb{Z}\}$.*

On rappelle que même pour des discriminants négatifs il existe au moins 2 automorphismes pour une forme de $\mathfrak{F}_\Delta$ donnée.

Les hypothèses de cette proposition sont illustrées dans le schéma suivant :



Démonstration. Si $\mathcal{M} \in \mathrm{SL}_2$ est une matrice de passage de f à g. Alors pour tout autre matrice de passage de f à g, $\mathcal{N} \in \mathrm{SL}_2$, on a $f.\mathcal{N}\mathcal{M}^{-1} = f$. Donc il existe $i \in \mathbb{Z}$ et un automorphisme $\mathcal{A}_i$ de f qui s'écrit $\mathcal{A}_i = \mathcal{N}\mathcal{M}^{-1}$ et donc une autre SL_2-matrice de passage de f à g vérifie $\mathcal{N} = \mathcal{A}_i\mathcal{M}$. □

Il y a donc une bijection entre l'ensemble des matrices de passage $\in \mathrm{SL}_2$ de $f \in \mathfrak{F}_\Delta$ à g et le groupe des d'automorphismes de f. On distingue le cas où $f \in \mathfrak{F}_\Delta$ est une forme imaginaire, et la cas où elle est réelle.

1.5.1 Cas des formes imaginaires

D'après la démonstration du théorème 1.2 pour toute forme imaginaire $f \in \mathfrak{F}_\Delta$ de discriminant :

$\Delta = -3$, $Aut(f)$ est un groupe cyclique d'ordre 6. Donc en utilisant la proposition 1.11 on sait que dans ce cas il existe exactement 6 matrices de passage entre f et chacune des formes de sa classe.

$\Delta = -4$, $Aut(f)$ est un groupe cyclique d'ordre 4. Donc en utilisant la proposition 1.11 on sait que dans ce cas il existe exactement 4 matrices de passage entre f et chacune des formes de sa classe.

$\Delta < -4$, on a $Aut(f) = \{\pm\mathcal{I}_2\}$. Donc en utilisant la proposition 1.11 on sait que dans ce cas, s'il existe une matrice de réduction $\mathcal{M} \in \mathrm{SL}_2$ telle que $f.(\mathcal{M}) = g$, alors il existe exactement deux matrices de passage de f à g qui sont $\mathcal{M}$ et $-\mathcal{M}$. On rappelle que $f.\mathcal{M} = f.(-\mathcal{M})$ (propriété 1.4).

1.5.2 Cas des formes réelles

En utilisant la démonstration du théorème 1.3 pour toute forme réelle $f \in \mathfrak{F}_\Delta$ de discriminant $\Delta > 0$, on a $Aut(f) = \{\pm\mathcal{U}_f^i : i \in \mathbb{Z}\}$.

Donc en utilisant la proposition 1.11 dans ce cas, s'il existe une matrice de réduction $\mathcal{M} \in \mathrm{SL}_2$ telle que $f.\mathcal{M} = g$, alors il y a une infinité de matrices de passage de f à g et ces matrices s'écrivent : $\{\pm\mathcal{U}_f^i\mathcal{M} : i \in \mathbb{Z}\}$.

Chapitre 2

Cryptologie et complexité

Ce chapitre à pour objectif de présenter la cryptologie et d'introduire les notions d'algorithme et de complexité.

La signification du mot cryptologie est donnée par son étymologie : concaténation du préfixe crypto- signifiant en grec "caché" et du suffixe -logie représentant la "science".

La cryptologie est donc la science du caché.

Cette science prend naissance dès l'apparition de l'écriture, c'est-à-dire à l'époque de l'Antiquité. Ce premier moyen de communication entre des entités d'une civilisation engendre les deux problématiques suivantes :

Tout d'abord, "peut-on transmettre une lettre sans que le facteur ne puisse la comprendre ?".

Ensuite si l'on trouve une façon de rendre la lettre à fortiori incompréhensible au facteur (lettre protégée), celui-ci "peut-il l'analyser jusqu'à l'interpréter de façon correcte ?".

Ces problématiques sont les deux branches de la cryptologie : la cryptographie qui est l'ensemble des techniques permettant de protéger une communication, et la cryptanalyse qui permet de retrouver une communication protégée.

Supposer le facteur analphabète ne résout pas les problématiques puisqu'un pirate peut toujours intercepter la lettre. De la même manière un canal de transmission ne peut décemment pas être considéré comme sûr, même s'il ne joue pas le rôle d'attaquant.

2.1 Objectifs de la cryptographie

Les grands objectifs de la cryptographie se divisent en trois parties : confidentialité, intégrité et authenticité.

2.1.1 La confidentialité

On veut que le message transmis ne puisse être lu que par les personnes concernées. Pour cela on utilise deux protocoles de façon coordonnée. Un premier en amont de la transmission qui va brouiller le message en un *chiffré* correspondant, seul ce chiffré sera transmis : c'est le protocole de *chiffrement*. Le second qui ne peut être effectué que par les entités concernées et qui à partir du chiffré reçu retrouve le message, s'appelle le protocole de *déchiffrement*.

2.1.2 L'intégrité

On cherche à assurer que le message ne puisse pas être modifié intentionnellement sans que le destinataire s'en aperçoive. Dans ce cas ou utilise une *fonction de hachage* qui fournit une *empreinte* du message. L'on transmet à la fois le message et son empreinte. Cette fonction doit être conçue de telle manière que la probabilité que deux messages différents aient la même empreinte soit extrêmement faible, d'où le nom d'empreinte par analogie avec l'empreinte digitale des individus. Pour vérifier l'intégrité du message reçu, le destinataire calcule son empreinte puis vérifie qu'elle est identique à l'empreinte qu'il à reçu.

2.1.3 L'authenticité et la non-répudiation

On veut s'assurer que l'entité avec laquelle on va communiquer est exactement celle qu'elle prétend être. On veut donc être convaincu de l'authenticité de l'entité avec laquelle on veut communiquer, celle ci doit alors nécessairement posséder une preuve de son identité et l'on doit pouvoir vérifier cette preuve. On utilise dans ce cas un protocole d'*identification*, qui ne doit bien sûr pas permettre d'usurper l'identité de l'expéditeur. Ensuite on peut vouloir, en plus de l'identification de l'expéditeur, s'assurer que celui-ci ne puisse pas renier le fait qu'il soit l'expéditeur (la non-répudiation). On peut faire l'analogie avec la signature manuscrite en bas d'un document. Dans ce cas on utilise encore deux protocoles de façon coordonnée. L'expéditeur utilise un protocole de *signature* qui va calculer la signature de l'empreinte du message grâce au fait qu'il est bien l'expéditeur (lui seul doit pouvoir calculer la signature). Cette signature doit donc contenir la preuve de l'identité de l'expéditeur en fonction du message. L'expéditeur transmet alors le message et sa signature. Les destinataires n'ont plus qu'à appliquer le protocole de *vérification* correspondant sur l'empreinte du message, qui va attester ou ne pas attester, que la signature du message est bien la preuve attendue par rapport au message reçu.

Il nous faut de manière évidente ne pas faire reposer ces objectifs sur le secret de leurs algorithmes. Cela serait comme supposer que la sécurité des coffres forts repose sur leur fonctionnement et non sur la combinaison (la *clé*). On considère donc les algorithmes utilisés dans les différents protocoles cryptographiques comme connus.

2.2 Cryptographie symétrique

La cryptographie *symétrique* est la plus ancienne forme de chiffrement, elle nécessite que les entités communicant ensemble aient une clé commune.

Il en découle une certaine symétrie entre ces entités puisque qu'elles connaissent le même secret. On en déduit l'inconvénient majeur de ce chiffrement : comment ces entités peuvent elles se transmettre la clé de façon sécurisée et être sûr que cette clé ne sera pas divulguée à d'autre entités ?

Ces protocoles de chiffrement symétrique sont par exemple en 1977 le DES (Data Encryption Standard) avec au plus 2^{56} clés possibles et en 2000 AES (Advanced Encryption Standard) qui peut être utilisé avec au plus 2^{256} clés. Ces chiffrements effectuent une suite d'itérations sur le message découpé en morceaux, et n'utilisent que des fonctions simples sur les blocs du message (décalage, permutation, 'ou' logique) ce qui les rend extrêmement rapides et donc très attractifs en pratique.

La sécurité de ces chiffrements repose essentiellement sur le rapport entre le nombre de clés possibles et la puissance de calcul des ordinateurs. Le nombre de clés possibles doit être assez grand, pour qu'on ne puisse pas déchiffrer un message reçu simplement en testant toutes les clés possibles (*attaque exhaustive*) sur un ordinateur. Actuellement, 2^{128} doit être le minimum de clés possibles pour qu'un chiffrement symétrique soit raisonnablement sûr. Pour donner un ordre de grandeur, 2^{128} est proche de $34 * 10^{37}$, l'âge de l'univers est d'environ 10^{10} années et le nombre d'instructions par seconde des processeurs actuels est inférieur à 10^{10}. Donc si on suppose qu'on peut tester 10^{12} clés par seconde (mille milliards), alors en un an on va tester seulement $31 * 10^{18}$ clés. Il faudrait donc encore au moins deux fois l'âge de l'univers pour tester l'ensemble des clés.

2.3 Cryptographie asymétrique

Pour résoudre le problème de l'échange de clés, la cryptographie *asymétrique* a été mise au point dans les années 1970 par Whitfield Diffie et Martin Hellman dans [16]. Par contre, les fonctions employées en cryptographie asymétrique demandent plus de temps de calcul que les fonctions utilisées en cryptographie symétrique.

Dans le chiffrement asymétrique ce n'est plus l'expéditeur qui veut s'assurer de la confidentialité de son message, mais le destinataire qui fait en sorte que lui seul puisse déchiffrer. Il s'assure donc de la confidentialité des messages qui lui sont transmis. Ce protocole se compose de deux clés.

Une *clé secrète* connue seulement du destinataire : cette clé n'a pas besoin d'être transmise et va lui permettre de déchiffrer.

Une *clé publique* que le destinataire diffuse aux expéditeurs qui veulent communiquer avec lui : les expéditeurs n'ont besoin que de cette clé pour chiffrer les messages.

Ce type de cryptographie repose essentiellement sur deux types de fonctions.

Les *fonctions à sens unique* : facilement calculables en pratique et difficilement inversables en pratique.

Et les *fonctions à trappe* qui sont des fonctions à sens unique avec en plus une *trappe*, seule la connaissance de cette trappe permet d'inverser la fonction de façon efficace.

On voit par exemple qu'un protocole de chiffrement pourrait être mis en place avec en clé publique la fonction à trappe et en clé secrète la trappe. Les expéditeurs chiffrent le message en lui appliquant la fonction, le destinataire est le seul à pouvoir retrouver le message (inverser la fonction) grâce à la trappe.

Beaucoup de ces fonctions proviennent de problèmes mathématiques qu'on ne sait actuellement pas résoudre en pratique.

La recherche en cryptographie asymétrique est donc étroitement liée à l'étude et la compréhension de structures mathématiques (comme par exemple les entiers, les entiers modulaires ou $\mathcal{O}_{\mathfrak{F}_\Delta}$ qui est l'ensemble des orbites de $\mathfrak{F}_\Delta$ par l'action de SL_2) et utilise la théorie de la complexité pour pouvoir quantifier et comparer la difficulté des problèmes entre eux.

2.4 Complexité

La théorie de la complexité a deux objectifs principaux. Tout d'abord quantifier les ressources nécessaires à la résolution d'un problème donné, et ensuite classer ces problèmes par rapport à la difficulté de les résoudre. Par ressources nécessaires on entend le temps de calcul et l'espace mémoire nécessaires à la résolution du problème.

2.4.1 Algorithme et complexité

Pour atteindre le première objectif, on modélise les algorithmes de façon a avoir un contenu précis quelque soit l'algorithme.

Un *algorithme* est défini comme étant une procédure qui pour une entrée donnée du problème se termine et fournit une solution.

La modélisation de l'algorithme se fait par une *machine de Turing* (Alan Turing en 1937). On décompose l'algorithme en une suite d'instructions élémentaires sur des données élémentaires. La machine de Turing exécute cette suite d'instructions avec une logique de

fonctionnement précise et sur un support précis appelé le *ruban*. Elle est caractérisée par le codage de ses entrées et sorties, et la table de transitions qui décrit son fonctionnement.

On a alors une représentation unifiée des algorithmes. Et puisque tout problème calculable peut être représenté par une machine de Turing (thèse de Alonzo Church en 1943), ces machines de Turing permettent de classifier les algorithmes suivant si ils sont réalisables en pratique, presque irréalisables en pratique, ou alors irréalisables en pratique.

Complexité en espace. La complexité en espace d'un algorithme est l'estimation de la taille de ruban utilisé lors de l'exécution de la machine de Turing sur une entrée.

Complexité en temps. La complexité en temps d'un algorithme est l'estimation asymptotique du nombre de transitions qu'effectue la machine de Turing en fonction de la taille de son entrée.

2.4.2 Difficulté d'un problème et complexité

La correspondance entre la difficulté d'un problème et la complexité en espace ou en temps est la suivante :

Remarque 2.1 *Si f et g sont deux fonctions, on dit que g est en $O(f)$ s'il existe des constantes $c > 0$ et n telles que $g(x) < cf(x)$ pour tout $x > n$. Il s'agit d'une borne supérieure à partir d'un certain rang, cela indique que la fonction g ne croit pas plus vite que f à partir de ce rang.*

On définit la *taille* d'un entier z comme étant $\log(z)$.

Partout dans cette thèse la fonction logarithme naturel est notée $\log(\cdot)$, alors que le logarithme en base b sera noté $\log_b(\cdot)$.

Les problèmes considérés comme *réalisables en pratique* sont ceux dont on connait un algorithme les résolvant de complexité inférieure à un polynôme en la taille de l'entrée de degré fixe. Cette *complexité polynomiale* est notée $O(x^y)$ avec x la taille de l'entrée, et y un petit entier positif fixe. Cette catégorie d'algorithme comprend par exemple les algorithmes de complexité constante $O(1)$, logarithmique $O(\log x)$, linéaire $O(x)$, quadratique $O(x^2)$ et cubique $O(x^3)$.

Les problèmes *difficiles en pratique* à résoudre sont ceux dont les seuls algorithmes connus pour les résoudre sont de complexité exponentielle $O(2^x)$.

Et enfin les problèmes intermédiaires qui sont *presque difficiles en pratique* à résoudre, sont ceux dont les algorithmes les plus rapides les résolvants sont de complexité sous-exponentielle $L_x(\alpha, \beta) = \exp(\beta x^\alpha \log x^{1-\alpha})$ avec $0 < \alpha < 1$ et une constante non nulle positive β.

Pour comparer la difficulté de deux problèmes difficiles en pratique on utilise une *réduction polynomiale* entre les problèmes. Soit $P1$ et $P2$ deux problèmes, la réduction polynomiale de $P1$ à $P2$ consiste à construire une machine de Turing qui en temps polynomial résolve $P1$ grâce à la solution de $P2$ sur certaines de ses données. En faisant cette réduction on a montré que si $P2$ se résout de façon efficace alors $P1$ aussi. Donc $P1$ ne peux pas être plus difficile à résoudre que $P2$, et de façon équivalente $P2$ est *au moins aussi difficile* à résoudre que $P1$. Si l'on peut faire une réduction polynomiale de $P1$ à $P2$, et aussi de $P2$ à $P1$ alors les deux problèmes sont considérés comme équivalents.

2.5 Problèmes difficiles de la théorie des nombres

Donnons deux exemples de problèmes issus de la théorie des nombres, et qui sont difficiles à résoudre en pratique.

2.5.1 Factoriser un entier RSA et ses variantes

Le problème calculatoire d'arithmétique le plus connu pour être difficile en pratique est de factoriser un grand entier positif produit de deux nombres premiers inconnus de taille proche. Cet entier sera appelé par la suite *un entier RSA* , en l'honneur de Ron Rivest, Adi Shamir et Len Adleman.

En 1978 dans [36], ils construisent le premier chiffrement asymétrique de clé publique un entier RSA, dont la sécurité est conjecturée comme équivalente au problème de la factorisation de sa clé publique. Le record actuel de factorisation d'entier RSA date du 12 décembre 2009 par Thorsten Kleinjung qui a factorisé un entier RSA de 768 bits (232 décimales). Actuellement on demande à un entier RSA d'être au moins de 2048 bits, pour qu'il soit considéré comme difficile à factoriser en pratique.

C'est en 1979 où pour la première fois la sécurité d'un cryptosystème est prouvée équivalente à la factorisation d'un entier RSA : c'est le cryptosystème de Rabin, en l'honneur de son auteur Michael Rabin. Plus précisément ce cryptosystème repose sur la difficulté d'extraire des racines carrées d'entier modulo un entier RSA, cette difficulté est prouvée équivalente à la factorisation de l'entier RSA utilisé.

2.5.2 Calculer le logarithme discret et ses variantes

Ce problème vit dans les groupes finis cycliques de générateur connu. Étant donnés le générateur et un élément du groupe, le problème du logarithme discret est de trouver le plus petit exposant tel que le générateur élevé à cette puissance est égal à l'élément.

Une variante est le problème Diffie-Hellman qui étant donné deux éléments du groupe cherche à calculer le générateur du groupe à la puissance $e_1 * e_2$ avec e_1 et e_2 les logarithmes discrets des deux éléments.

Remarquons que si l'on sait résoudre le problème du logarithme discret de façon efficace alors on peut résoudre le problème Diffie-Hellman aussi de manière efficace. Le problème Diffie-Hellman ne peut donc pas être plus difficile à résoudre en pratique que le problème du logarithme discret.

On utilise souvent ces problèmes sur les groupes $\mathbb{Z}_p^*$ avec p un grand nombre premier, et $\mathbb{Z}_n^*$ avec l'entier RSA n. L'algorithme le plus rapide qui résout le logarithme discret dans les groupes $\mathbb{Z}_p^*$ est une variante de l'algorithme de calcul d'index nommé NFS (Number Field Sieve). Cet algorithme est de complexité sous-exponentiel, et est aussi l'algorithme le plus rapide pour factoriser des entiers RSA. L'utilisation du groupe $\mathbb{Z}_n^*$ nous donne une relation avec la factorisation de n. En effet résoudre le problème Diffie-Hellman dans $\mathbb{Z}_n^*$ est au moins aussi difficile à résoudre que le problème de factoriser l'entier RSA. Des groupes aussi très attractifs sont les sous groupes de $\mathbb{Z}_p^*$ d'ordre un grand nombre premier. Ils permettent de manipuler des exposants plus petits que lorsque l'on est dans $\mathbb{Z}_p^*$ et d'avoir la même sécurité, puisque la seule manière connue pour résoudre le logarithme discret dans ces sous groupes est de résoudre dans $\mathbb{Z}_p^*$.

2.6 Complexité de problèmes à résoudre dans $\mathcal{O}_{\mathfrak{F}_\Delta}$

Dans ce manuscrit nous nous intéressons à l'ensemble $\mathcal{O}_{\mathfrak{F}_\Delta}$ qui est l'ensemble des orbites de $\mathfrak{F}_\Delta$ par l'action de SL_2, mais aussi à son utilisation en cryptographie.

Pour utiliser l'ensemble $\mathcal{O}_{\mathfrak{F}_\Delta}$ dans des schémas cryptographiques, il nous faut bien connaître le comportement de cet ensemble et surtout avoir conscience de ce que l'on peut résoudre ou pas en pratique dans cet ensemble.

On énumère une liste de problèmes à résoudre dans $\mathcal{O}_{\mathfrak{F}_\Delta}$. Pour chacun d'eux on énonce la complexité de l'algorithme le plus connu qui les résout.

Ces problèmes sont énoncés plus en détail dans la partie 3. L'objectif de cette section et de résumer les principales manipulations dans $\mathcal{O}_{\mathfrak{F}_\Delta}$ qui sont réalisables en pratique ou pas.

2.6.1 Calculer une représentante réduite pour une classe donnée

Pour toute forme on veut trouver de façon efficace une forme dans sa classe d'équivalence et on veut que cette forme soit facile à manipuler : avec des coefficients de taille bornée.

Cette forme que l'on cherche est une forme représentative de sa classe, on l'appelle forme réduite (car ses coefficients sont de taille bornée) et elle n'est pas forcément unique.

Pour résoudre ce problème on utilise un algorithme dit de réduction qui pour une forme $f \in \mathfrak{F}_\Delta$ donnée, retourne en temps quadratique $O(\log(\|f\|)^2)$ (voir [4, 41]) une forme réduite de $\overline{f}$ dont les coefficients sont de taille majorée par $\log(|\Delta|)$.

Proposition 2.1 *L'algorithme classique de réduction est de complexité quadratique en la taille de son entrée.*

La démonstration de cette proposition ce trouve dans [4] et [41].

2.6.2 Calculer l'unité fondamentale pour un discriminant donné

On rappelle que l'unité fondamentale ϵ_Δ (section 1.4.3) n'est définie que pour les discriminants Δ qui ne sont pas des carrés parfaits et qui sont positifs.

L'unité fondamentale ϵ_Δ joue un rôle très important dans l'étude de $\mathcal{O}_{\mathfrak{F}_\Delta}$, puisque tout automorphisme d'une forme de $\mathfrak{F}_{\Delta_f}$ se construit à partir d'une puissance de ϵ_Δ (proposition 1.10).

De la section 8.3.3 page 162 de [8], et de [7] on extrait la proposition importante suivante :

Proposition 2.2 *Soit un discriminant Δ choisi aléatoirement et qui n'est ni un carré parfait ni négatif, alors ϵ_Δ est de taille exponentielle en Δ avec une forte probabilité.*

Cette proposition signifie qu'il y a une forte probabilité que l'on ne puisse pas en pratique calculer et mettre en mémoire ϵ_Δ pour de grands discriminants Δ choisi aléatoirement.

De la même manière cette proposition entraîne qu'il y a une forte probabilité que l'on ne puisse pas calculer l'automorphisme fondamental d'une forme de $\mathfrak{F}_{\Delta_f}$ (puisque par définition la 1.28 cet automorphisme se construit à partir de la solution fondamentale).

2.6.3 Tester l'équivalence de deux formes données

Même s'il est possible en pratique de calculer une représentante réduite d'une classe, cela n'entraîne pas que l'on puisse décider en pratique de l'équivalence entre deux formes de $\mathfrak{F}_\Delta$.

Cela dépend du nombre de représentantes réduites par classe.

Nous verrons que dans le cas des formes imaginaires chaque orbite de $\mathcal{O}_{\mathfrak{F}_\Delta}$ contient une unique réduite, alors que dans le cas des formes réelles chaque orbite de $\mathcal{O}_{\mathfrak{F}_\Delta}$ peut contenir plusieurs réduites.

Proposition 2.3 *Pour un discriminant Δ qui n'est pas un carré parfait et qui est positif, alors chaque orbite de $\mathcal{O}_{\mathfrak{F}_\Delta}$ contient $O(\log(\epsilon_{\Delta_f}))$ réduites.*

Une preuve de ce théorème se trouve dans [8] proposition 6.13.2.

On remarque que, expérimentalement, le nombre de formes par cycle est de l'ordre de $\sqrt{\Delta}$ (voir [8] page 135 et [29] page 185).

Cette proposition signifie qu'il y a une forte probabilité que l'on ne puisse pas en pratique décider de l'équivalence entre deux formes réelles de $\mathfrak{F}_\Delta$ (voir l'algorithme 4) pour un discriminant Δ choisi aléatoirement, puisque la proposition 2.2 nous dit que dans ce cas il y a une forte probabilité pour que ϵ_Δ soit de taille exponentielle en Δ.

Par contre le coût en temps pour décider de l'équivalence de deux formes imaginaires de $\mathfrak{F}_\Delta$ est le même que le coût asymptotique en temps de l'algorithme de réduction, puisque dans ce cas il existe une unique réduite par classe et que celle-ci est obtenue en appliquant l'algorithme polynomial de réduction (l'algorithme 3).

Proposition 2.4 *On peut tester l'équivalence de deux formes imaginaires de $\mathfrak{F}_\Delta$ en temps quadratique en la taille de son entrée.*

La démonstration de cette proposition ce trouve à la section 5.8 de [8].

Deuxième partie

Étude des classes d'équivalences

Chapitre 3

Classes d'équivalence

Ce chapitre est consacré aux orbites de l'ensemble $\mathfrak{F}_\Delta$ par l'action du groupe SL_2. Cet ensemble d'orbites est noté $\mathcal{O}_{\mathfrak{F}_\Delta}$ dans le chapitre 1. Nous rappelons que l'ensemble $\mathfrak{F}_\Delta$ n'est définit que pour des discriminants Δ qui ne sont pas des carrés parfaits.

3.1 Formes représentatives de sa classe

Pour toute forme f on peut trouver de façon efficace une forme g dans la classe d'équivalence $\overline{f}$ qui a des coefficients de taille bornée. La forme g est donc une forme représentative de la classe $\overline{f}$, et il existe une matrice de passage $\mathcal{M} \in \mathrm{SL}_2$ vérifiant que $f.\mathcal{M} = g$.

On utilise pour cela un *algorithme de réduction*, nous verrons qu'il y a un algorithme de réduction pour chaque type de forme : les formes imaginaires, et les formes réelles.

Définition 3.1 *Pour une forme f donnée, une matrice $\mathcal{M} \in SL_2$ vérifiant que $f.\mathcal{M}$ est une forme réduite, est appelée une matrice de réduction de f.*

La réduction d'une forme f par la matrice de réduction $\mathcal{M}$ est schématisée par :

$$f \overset{\mathcal{M}}{\longmapsto} g \text{ ou plus simplement } f \longmapsto g.$$

Une forme h représentative de sa classe $\overline{h}$ est obtenu en appliquant l'algorithme de réduction à une forme de $\overline{h}$, elle sera donc appelée *forme réduite*.

Une forme imaginaire ou réelle (a, b, c) est *réduite* si elle satisfait deux conditions simultanément, une condition de *normalisation* qui définit le choix d'un représentant de $b \mod 2a$, et une condition de *réduction*, qui sert à majorer $|a|$ (ou $|c|$).

Pour normaliser une forme, une seule étape de translation est nécessaire, et pour obtenir la condition de réduction on en effectue un certain nombre.

3.1.1 Formes imaginaires réduites

Dans le cas imaginaire, les conditions pour qu'une forme soit réduite sont naturelles : la forme $f = (a, b, c)$ est *normalisée* si et seulement si $b \in]-|a|, |a|]$, et elle est *réduite* si en plus $|a|$ est le premier minimum $\lambda(f)$.

Dans [8], la section 5.10 et le théorème 5.7.6 montre que cette définition d'une forme imaginaire réduite est équivalente à la définition ci-dessous.

Définition 3.2 *Une forme imaginaire (a, b, c) est réduite si $b \in]-|a|, |a|]$ et $|a| \leqslant |c|$, dans le cas où $a = c$ on impose $b \geqslant 0$.*

La réduction de Gauss réduit une forme en lui appliquant une suite de $\boldsymbol{SE}$ et normalisation $\boldsymbol{T}(\lfloor -b/2a \rceil)$.

La fonction $\lfloor \cdot \rceil$ retourne l'entier le plus proche, et afin de satisfaire la première condition d'une forme réduite on prend la convention : $\lfloor -1/2 \rceil = 0$ et $\lfloor 1/2 \rceil = 1$ (on vérifie que $\boldsymbol{T}(\lfloor -b/2a \rceil) = (a, B, C)$ vérifie $-|a| < B (= b + 2a \lfloor -b/2a \rceil) \leqslant |a|$).

Définition 3.3 *Pour une forme imaginaire (a, b, c) l'entier $\lfloor -b/2a \rceil$ est noté μ_f.*

En utilisant la représentation affine des formes, on comprend que faire la translation par μ_f, qui est l'entier le plus proche de la moyenne des racines complexes de f, minimise en valeur absolue le deuxième et troisième coefficient de $f.\boldsymbol{T}(\mu_f)$ car sa moyenne des racines est alors dans $]-1/2, 1/2]$ et son troisième coefficient est égal à $p_f(\mu_f)$.

Remarque 3.1 *La normalisation d'une forme f peut toujours être obtenue par l'action de la translation $\boldsymbol{T}(\mu_f)$. Pour le voir on prend en compte la définition de μ_f et on utilise la représentation affine (définition 1.6) des formes imaginaires. Cette remarque est illustrée dans la figure 3.1. On remarquera aussi que si f est une forme imaginaire normalisée alors $f.\boldsymbol{T}(\mu_f) = f$ car dans ce cas $\mu_f = 0$ et donc $\boldsymbol{T}(\mu_f) = \mathcal{I}_2$.*

On donne l'algorithme 1 de réduction d'une forme imaginaire normalisée, et on notera que toutes les matrices de passage utilisées par l'algorithme sont bien dans SL_2.

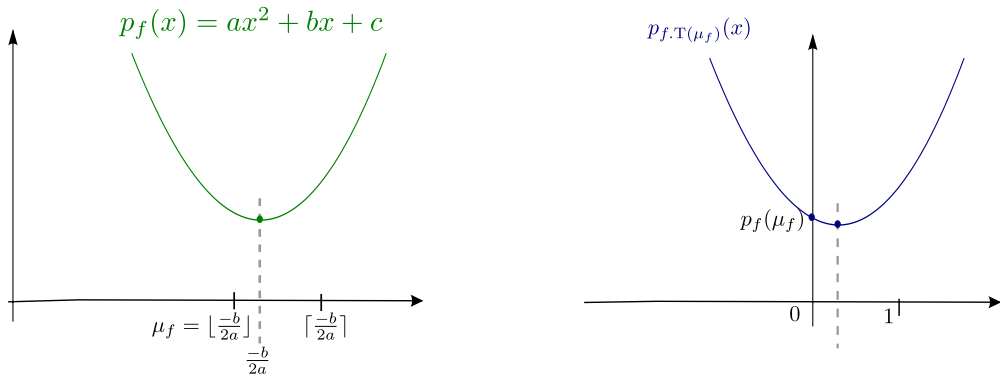


FIG. 3.1 – *Représentation affine d'une forme imaginaire f et de $f.\boldsymbol{T}\mu_f$*

Cette figure illustre la parabole $p_f(x)$ associée à une forme imaginaire $f = (a, b, c)$, ainsi que la parabole $p_{f.\boldsymbol{T}\mu_f}(x)$ associée à la forme $f.\boldsymbol{T}\mu_f$.
La forme $f.\boldsymbol{T}\mu_f$ est une forme normalisée.

Algorithme 1 : ***RéductionGaussImaginaire***

Entrée : $f = (a, b, c)$ une forme imaginaire normalisée
Sorties : $f.\mathcal{M}$ la forme réduite et $\mathcal{M} \in \mathrm{SL}_2$

1: $\mathcal{M} \leftarrow \mathcal{I}_2$
2: **tantque** $b \notin]-|a|, |a|]$ ou $|a| > |c|$ **faire**
3: $\quad f \leftarrow f.\boldsymbol{SE}$ et $\mathcal{M} \leftarrow \mathcal{M}\boldsymbol{SE}$
4: $\quad \mathcal{M} \leftarrow \mathcal{M}\boldsymbol{T}(\mu_f)$
5: $\quad f \leftarrow f.\boldsymbol{T}(\mu_f)$ ▷ Étape de normalisation
6: **fin tantque**
7: **si** $|a| = |c|$ **alors**
8: $\quad \mathcal{M} \leftarrow \boldsymbol{SE}$
9: $\quad f \leftarrow f.\boldsymbol{SE}$
10: **finsi**
11: **retourner** f et $\mathcal{M}$

Puisque à chaque boucle "**tantque**" de l'algorithme 1, la forme est normalisée et que la valeur absolue du premier coefficient décroît strictement, l'algorithme 1 finit bien sur une forme imaginaire réduite.

Comme l'algorithme s'exécute pour toute forme imaginaire normalisée, et que toute forme imaginaire est équivalente à une forme imaginaire normalisée (remarque 3.1), alors pour toute forme imaginaire il existe au moins une forme réduite dans sa classe d'équivalence.

Proposition 3.1 *Toute forme imaginaire réduite $f = (a, b, c)$, vérifie que $|a|$ et $|b|$ sont majorés par $\sqrt{|\Delta_f|/3}$, et $|c| \leqslant |\Delta_f|/(3|a|)$.*

Cette proposition montre que chaque classe d'équivalence contient un nombre fini de représentantes réduites.

Montrons la proposition 3.1.

Démonstration. En premier si la forme $f = (a, b, c)$ est réduite, $|\Delta_f| = 4ac - b^2$ est minoré par $4a^2 - a^2 = 3a^2$ et donc on en déduit la majoration de la valeur absolue de ses coefficients a (et donc aussi b puisque la forme est réduite) par $\sqrt{|\Delta_f|/3}$. On en déduit aussi une majoration sur $|c|$ puisque en écrivant $4|ac| = |\Delta_f| + b^2 \leqslant |\Delta_f| + a^2 \leqslant |\Delta_f| + |\Delta_f|/3$ il vient $|c| \leqslant |\Delta_f|/(3|a|)$. □

Proposition 3.2 *Si une forme imaginaire $f = (a, b, c)$ vérifie $|a| < \sqrt{\Delta_f}/2$ et qu'elle est normalisée alors f est une forme imaginaire réduite.*

Démonstration. Comme f est imaginaire $|\Delta_f| = -b^2 + 4ac$, on a $|c| = (|\Delta_f| + b^2)/4|a| \geqslant |\Delta_f|/4|a| > a^2/|a| = |a|$ car $|a| < \sqrt{\Delta_f}/2$. Et comme en plus f est normalisée on en déduit que f est bien une forme réduite. □

Théorème 3.1 *Dans chaque classe d'équivalence de $\mathfrak{F}_{\Delta<0}$ il existe une unique forme réduite.*

Démonstration. Pour démontrer l'unicité on fait un raisonnement par l'absurde. Supposons qu'en plus de la forme réduite $f = (a, b, c)$ il existe une deuxième forme réduite distincte $g = (a_g, b_g, c_g)$, et une matrice $\mathcal{M} = (\alpha, \beta; \gamma, \delta) \in \mathrm{SL}_2$ et vérifiant $\mathcal{M} \neq \pm\mathcal{I}_2$, $a_g = f(\alpha, \gamma)$, et $|a_g| \leqslant |a|$ (sinon il suffit d'échanger a et a_g).

Comme le discriminant est négatif on peut minorer $4af(\alpha, \gamma) = (2a\alpha - b\gamma)^2 + |\Delta_f|\gamma^2$ par $\gamma^2|\Delta_f|$, de plus les coefficients a et a_g sont du même signe.

- Si $|\gamma| \geqslant 2$, nous obtenons la contradiction $|a_g| \geqslant 3|a|$ car la forme étant réduite on a $|\Delta_f| \geqslant 3a^2$.
- Ensuite pour $|\gamma| = 1$ il vient $a_g = a\alpha^2 \pm b\alpha + c$.

  Si $|\alpha| \geqslant 2$ alors $|a_g| = |a\alpha^2 \pm b\alpha + c| \geqslant ||a\alpha^2 + c| - |b\alpha|| = ||a|\alpha^2 + |c| - |b\alpha||$ (car a et c sont de même signe) on a donc $|a_g| \geqslant ||\alpha|(|a||\alpha| - |b|) + |c|| = |\alpha|(|a||\alpha| - |b|) + |c|$ (car la forme est réduite $0 \leqslant |b| < |a\alpha|$) et il vient $|a_g| \geqslant |\alpha|(|a||\alpha| - |b|) + |c| > |c|$ ce qui contredit l'hypothèse $|a_g| = \lambda(f)$.

  Si $|\alpha| = 1$ alors $a_g = a \pm b + c$. On a $|a_g| \geqslant ||a + c| - |b|| = ||a| + |c| - |b|| = |a| + |c| - |b| \geqslant 2|a| - |a|$ car f est une forme réduite. Il suit que $a_g = a$, et comme $b \in]-|a|, |a|]$ et que b_g doit l'être aussi on a $g = (a, \pm b, c)$. Mais comme $a = a \pm b + c$, alors $\pm b = c$, donc la forme f est $(a, |b|, \pm b)$ car elle est réduite. Il suit que $g = (a, |b|, \pm b) = f$ ce qui contredit les hypothèses.

Sinon $\alpha = 0$ donc $a_g = c$, et comme on a $|a_g| \leqslant |a| \leqslant |c|$ il vient que $a_g = c = a$, puisque a et c ont le même signe ($|\Delta_f| = -b^2 + 4ac \Rightarrow 4ac \geqslant b^2$). La forme g est dans ce cas $g = (a, b, a) = f$, puisque $b \in]-|a|, |a|]$ est positif, ce qui contredit les hypothèses.

- Et enfin $\gamma = 0$ entraîne que $a = a_g$, en effet comme $\mathcal{M} \in \mathrm{SL}_2$ on a que $\alpha = \delta = \pm 1$. Dans ce cas $b_g = \pm b$ car $b \in]-|a|, |a|]$ et b_g doit l'être aussi. La forme g est donc de type $(a, \pm b, c)$, mais comme $\alpha = \delta$ alors la forme g est finalement $(a, b, c) = f$ ce qui contredit les hypothèses.

□

3.1.2 Formes réelles réduites

Dans le cas réel ($\Delta_f > 0$), imposer à une forme $f = (a, b, c)$ de vérifier $|a| = \lambda(f)$ pour qu'elle soit réduite (comme dans le cas imaginaire) est trop restrictif car les plus petits entiers $(\alpha, \beta) \neq (0, 0)$ tels que $|f(\alpha, \beta)| = \lambda(f)$ sont généralement exponentiels en la taille de f.

Définition 3.4 *Une forme réelle $f = (a, b, c)$ est normalisée si et seulement si*
- $b \in]-|a|, |a|]$ *lorsque* $|a| \geqslant \sqrt{\Delta_f}$,
- $b \in \left]\sqrt{\Delta_f} - 2|a|, \sqrt{\Delta_f}\right[$ *quand* $|a| < \sqrt{\Delta_f}$,

et f est réduite si

$$\left|\sqrt{\Delta_f} - 2|a|\right| < b < \sqrt{\Delta_f}.$$

Définition 3.5 *Pour une forme réelle $f = (a, b, c)$ on définit l'entier ν_f comme étant*
- $\lfloor -b/2a \rceil$ *lorsque* $|a| \geqslant \sqrt{\Delta_f}$
- $\mathrm{signe}(a) \left\lfloor (-b + \sqrt{\Delta_f})/2|a| \right\rfloor$ *lorsque* $|a| < \sqrt{\Delta_f}$.

La réduction de Gauss réduit une forme $f = (a, b, c)$ en lui appliquant une suite de $\boldsymbol{SE}$ et normalisation $\boldsymbol{T}(\nu_f)$.

Remarque 3.2 *Comme pour les formes imaginaires, la normalisation d'une forme réelle $f = (a, b, c)$ peut toujours être obtenue par l'action de la translation $\boldsymbol{T}(\nu_f)$.*

On obtient cette remarque en utilisant la définition de l'entier ν_f dans les deux cas suivants :
- *Si la distance entre les racines de f est $\leqslant 1$ ($\sqrt{\Delta_f} \leqslant |a|$) alors ν_f est l'entier le plus proche de la moyenne des racines de f, et $f.\boldsymbol{T}(\nu_f)$ est bien une forme normalisée puisque la moyenne de ses racines est dans l'intervalle $]-1/2, 1/2]$.*
- *Sinon la distance entre les racines de f est strictement plus grande que 1.*
  Dans le cas où a est positif l'entier ν_f est $\lfloor \zeta_f^+ \rceil$, sinon c'est l'entier $\lceil \zeta_f^- \rceil$ (voir la propriété 4.3). Dans le premier cas la forme $f.\boldsymbol{T}(\nu_f)$ a sa plus grande racine dans $]0; 1[$, et dans le deuxième cas c'est sa plus petite racine qui est dans $]-1; 0[$. Dans les

deux cas cela entraîne que $0 < -b + \sqrt{\Delta_f} < 2|a|$*, en d'autre termes la forme* $f.\boldsymbol{T}(\nu_f)$ *est normalisée.*

Comme dans le cas imaginaire on remarquera que si f *est une forme réelle normalisée alors* $f.\boldsymbol{T}(\nu_f) = f$ *car dans ce cas* $\nu_f = 0$ *et donc* $\boldsymbol{T}(\nu_f) = \mathcal{I}_2$*.*

On donne l'algorithme 2 de réduction d'une forme réelle normalisée, et on notera que toutes les matrices de passage utilisées par l'algorithme sont bien dans SL_2.

Algorithme 2 : ***RéductionGaussRéelle***

Entrée : $f = (a, b, c)$ une forme réelle normalisée
Sorties : $f.\mathcal{M}$ la forme réduite et $\mathcal{M} \in \mathrm{SL}_2$

1: $\mathcal{M} \leftarrow \mathcal{I}_2$
2: **tantque** f n'est pas réduite **faire**
3: $\quad f \leftarrow f.\boldsymbol{SE}$ et $\mathcal{M} \leftarrow \mathcal{M}\boldsymbol{SE}$
4: $\quad \mathcal{M} \leftarrow \mathcal{M}\boldsymbol{T}(\nu_f)$
5: $\quad f \leftarrow f.\boldsymbol{T}(\nu_f)$ ▷ Étape de normalisation
6: **fin tantque**
7: **retourner** f et $\mathcal{M}$

Une preuve de la terminaison de l'algorithme ***RéductionGaussRéelle*** se trouve dans [8], elle est donnée par le théorème 6.5.3.

Puisque l'algorithme termine on déduit que chaque classe d'équivalence de $\mathfrak{F}_{\Delta>0}$ contient au moins une forme réduite.

Proposition 3.3 *Toute forme réelle réduite* $f = (a, b, c)$ *vérifie les deux points suivants :*

$$0 < b < \sqrt{\Delta_f} \text{ et } |a| + |c| < \sqrt{\Delta_f}.$$

Elle vérifie donc en particulier : soit $|a| < \sqrt{\Delta_f}/2$ *et* $|c| < \sqrt{\Delta_f}$*, soit l'inverse* $|c| < \sqrt{\Delta_f}/2$ *et* $|a| < \sqrt{\Delta_f}$*.*

Comme dans le cas des formes imaginaires cette proposition, montre que chaque classe d'équivalence contient un nombre fini de représentantes réduites. Mais ici, dans le cas des formes réelles, cette représentante n'est pas unique.

Démonstration. Démontrons la proposition 3.3. Si la forme $f = (a, b, c)$ est réduite, on a $0 < b < \sqrt{\Delta_f}$ ce qui entraîne que a et c sont de signe inverse ($0 > b^2 - \Delta_f = 4ac$). De plus on a $0 > (2|a| - \sqrt{\Delta_f})^2 - b^2 = 4a^2 + \Delta_f - 4|a|\sqrt{\Delta_f} - b^2 = 4|a|(|a| - \sqrt{\Delta_f} + |c|)$ car $|\sqrt{\Delta_f} - 2|a|| < b$, ce qui entraîne que $|a| - \sqrt{\Delta_f} + |c| < 0$. Puisque la somme de deux entiers positifs est plus grande que l'un des deux entiers cette dernière inégalité entraîne que $|a|$ et $|c|$ sont majorés par $\sqrt{\Delta_f}$. Et dans le cas où $|a| < |c|$ (sinon il suffit d'inverser a et c), elle implique $|a| < \sqrt{\Delta_f}/2$. □

Proposition 3.4 *Si une forme réelle normalisée $f = (a, b, c)$ vérifie $|a| < \sqrt{\Delta_f}/2$ alors c'est une forme réelle réduite.*

Démonstration. On a $2|a| < \sqrt{\Delta_f}$ donc $\sqrt{\Delta_f} - 2|a| > 0$ et comme f est normalisée il vient $|\sqrt{\Delta_f} - 2|a|| = \sqrt{\Delta_f} - 2|a| < b < \sqrt{\Delta_f}$ donc f est une forme réelle réduite. □

Théorème 3.2 *Un nombre fini de formes réelles de discriminant Δ_f sont réduites, et elles forment un unique "cycle de réduites" dans chaque classe.*

En d'autre mot chaque classe contient un unique "cycle de réduites", qui lui même contient un nombre fini de formes réduites.

Une preuve de ce théorème se trouve dans [9] proposition 3.4 et dans [8] proposition 6.13.2.

En particulier, comme le montre [18] deux formes réelles réduites sont équivalentes si et seulement si elles se trouvent sur le même cycle de réduites.

On rappelle que ce cycle contient généralement un nombre exponentiel de formes (proposition 2.3), généralement on ne peut donc pas le parcourir en pratique.

Nous donnerons dans le chapitre 4 notre interprétation de cette réduction, celle-ci transcrit les conditions sur les coefficients à des conditions sur les racines des formes et leur représentation affine. De cette manière l'étude de l'algorithme ***RéductionGaussRéelle*** nous semble plus naturelle.

Nous verrons que cet algorithme ne minimise pas la norme de la matrice de réduction des formes réelles, et nous produirons un algorithme qui lui retourne la plus petite matrice de réduction d'une forme réelle.

3.1.3 Cycle de formes réelles réduites

Explicitons ce qu'est le *cycle de réduites* (cycle de formes réelles réduites) d'une forme réelle réduite $f \in \mathfrak{F}_\Delta$.

Définition 3.6 *Si $f = (a, b, c)$ est une forme réelle réduite, alors*
sa voisine de droite est la forme $f.\boldsymbol{SET}(\nu_{f.\boldsymbol{SE}}) = (c, -b+2c\nu_{f.\boldsymbol{SE}}, c\nu^2_{f.\boldsymbol{SE}} - b\nu_{f.\boldsymbol{SE}} + a)$,
sa voisine de gauche est la forme $f.\boldsymbol{T}(\nu_f)\boldsymbol{SE} = (a\nu_f^2 + b\nu_f + c, -(b + 2a\nu_f), a)$.

Remarquons que puisque f est réduite alors ses deux voisines (de droite et de gauche) sont elles aussi réduites : on pourra facilement s'en convaincre en utilisant notre proposition 4.2 qui relie la notion de réduction à la position des racines de la forme, et en n'oubliant pas que le premier et le troisième coefficients d'une forme réelle réduite sont de signe inverse. On remarque également que ces deux voisines sont uniques.

Définition 3.7 *On définit la fonction* ***VoisineDroite****f qui pour une forme réelle réduite f renvoie la voisine de droite de f.*

L'action de $\boldsymbol{SET}(\nu_{f.\boldsymbol{SE}})$ sur la forme réelle réduite f est la voisine de droite g de f. Ce que l'on schématise par :

$$f \xmapsto{\mathbf{SET}(\nu_{f.\mathbf{SE}})} g \text{ ou simplement } f \longmapsto g.$$

Puisque nous avons définit formellement ce qu'est un cycle de réduite, on peut compléter le Théorème 3.2 par le théorème suivant :

Théorème 3.3 *Une itération successive de voisines de droite permet d'énumérer toutes les formes réduites équivalentes à la forme réelle réduite $f \in \mathfrak{F}_\Delta$ et forme le cycle de réduites de f.*

Exemple 3.1 *Le cycle de réduites de la forme réduite $(1, 9, -6)$ de discriminant $\Delta = 105$ est l'énumération successive de voisine de droite. C'est donc le cycle :* $((1, 9, -6)(-6, 3, 4)(4, 5, -5)(-5, 5, 4)(4, 3, -6)(-6, 9, 1))$.

La voisine de droite de $(1, 9, -6)$ est $(-6, 3, 4)$; la voisine de droite de $(-6, 3, 4)$ est $(4, 5, -5)$; et ainsi de suite jusqu'à ce qu'à $(-6, 9, 1)$ qui a pour voisine de droite la forme $(1, 9, -6)$.

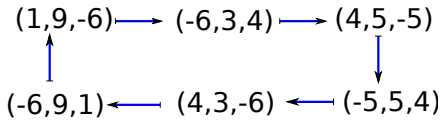


FIG. 3.2 – *Cycle de réduites d'une forme réduite*

Cette figure illustre le cycle de réduites de $(1, 9, -6)$.
Nous avons :

$$(1, 9, -6).\begin{pmatrix} 0 & 1 \\ -1 & 1 \end{pmatrix} = (-6, 3, 4)$$
$$(-6, 3, 4).\begin{pmatrix} 0 & 1 \\ -1 & -1 \end{pmatrix} = (4, 5, -5)$$
$$(4, 5, -5).\begin{pmatrix} 0 & 1 \\ -1 & 1 \end{pmatrix} = (-5, 5, 4)$$
$$(-5, 5, 4).\begin{pmatrix} 0 & 1 \\ -1 & -1 \end{pmatrix} = (4, 3, -6)$$
$$(4, 3, -6).\begin{pmatrix} 0 & 1 \\ -1 & 1 \end{pmatrix} = (-6, 9, 1)$$
$$(-6, 9, 1)\begin{pmatrix} 0 & 1 \\ -1 & -9 \end{pmatrix} = (1, 9, -6).$$

Finalement puisque toute forme $g \in \overline{f}$ se réduit par l'algorithme de réduction à une forme réduite h qui appartient au cycle de réduites de f (théorème 3.2), on définit le cycle de réduites de toute forme réelle $g \in \overline{f}$ comme étant simplement le cycle de réduites de f.

Définition 3.8 *Le cycle de réduites d'une classe $\overline{f}$, où $f \in \mathfrak{F}_{\Delta_f}$ est une forme réduite, est le cycle de réduites de f.*

Exemple 3.2 *Le cycle de réduites de la forme $(-6, -33, -41)$ de discriminant $\Delta = 105$ est le cycle de réduites de **RéductionGaussRéelle**$((-6, -33, -41)) = (-6, 3, 4)$.*

C'est donc le même cycle que dans l'exemple précédent.

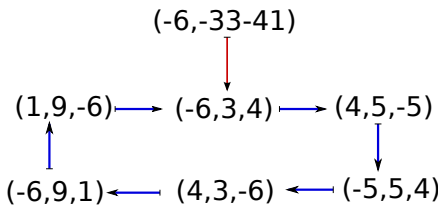


FIG. 3.3 – *Cycle de réduites d'une classe d'équivalence*

Cette figure illustre la réduction de la forme $(-6, -33, -41)$ de discriminant $\Delta = 105$, et son cycle de réduites.

Dans la suite on pourra abréger cycle de réduites pour une forme réelle g, simplement par *cycle* de g.

3.1.4 Matrice de réduction

Nous donnons dans cette partie la matrice de réduction retournée par les algorihtmes 1 et 2 ayant en entrée une forme $f \in \mathfrak{F}_\Delta$ normalisée.

Soit $\mathcal{M} \in \mathrm{SL}_2$ cette matrice, on sait donc que $f.\mathcal{M}$ est une forme réduite.

Appelons $f_i = (a_i, b_i, c_i)$ les valeurs successives de f au début de la boucle "tantque" de ces algorithmes.

Si la forme f est imaginaire on pose $h_i = \mu_i = \mu_{(f_i.\boldsymbol{ES})}$ car dans ce cas la réduction de la forme f se fait par application de l'algorithme 1.

Si la forme f est réelle on pose $h_i = \nu_i = \nu_{(f_i.\boldsymbol{ES})}$ puisque dans ce cas on exécute l'algorithme 2.

Supposons que la forme f_0 ne soit pas réduite et soit m le plus petit indice tel que f_m est réduite. Pour chaque $i \in [1; m]$, la matrice de réduction de f_0 à f_i est :

$$\mathcal{M}_i = \begin{pmatrix} 0 & -1 \\ 1 & h_0 \end{pmatrix} \begin{pmatrix} 0 & -1 \\ 1 & h_1 \end{pmatrix} \dots \begin{pmatrix} 0 & -1 \\ 1 & h_{i-1} \end{pmatrix} = \begin{pmatrix} \alpha_i & \beta_i \\ \gamma_i & \delta_i \end{pmatrix}. \tag{3.1}$$

Les coefficients vérifient la récurrence pour $i \geqslant 2$:

$$\begin{aligned} \alpha_{i+1} = \beta_i = h_{i-1}\beta_{i-1} - \beta_{i-2} \quad &\text{et} \quad (\beta_0, \beta_1) = (0, -1) \\ \gamma_{i+1} = \delta_i = h_{i-1}\delta_{i-1} - \delta_{i-2} \quad &\text{et} \quad (\delta_0, \delta_1) = (1, h_1). \end{aligned}$$

3.2 Nombre de classes

Le nombre de classes B_Δ est le nombre d'orbites de l'action du groupe SL_2 sur l'ensemble des formes $\mathfrak{F}_\Delta$.

On distingue le cas ou f est une forme imaginaire, et le cas ou f est une forme réelle.

3.2.1 Cas des formes imaginaires

Du théorème 3.1 on déduit que le nombre de classes B_Δ est exactement le nombre de formes imaginaires et réduites de $\mathfrak{F}_\Delta$.

Corollaire 3.1 *Si le discriminant Δ est négatif alors B_Δ est exactement le nombre de formes réduites de $\mathfrak{F}_\Delta$.*

Exemple 3.3 *Il y a exactement 5 formes réduites de discriminant $\Delta = -103$ et chacune est l'unique représentante réduite de sa classe. On a $\mathrm{B}_{-103} = 5$.*

(1,1,26) (2,1,13) (2,-1,13) (4,3,7) (4,-3,7)

FIG. 3.4 – *Les 5 uniques formes réduites de discriminant $\Delta = -103$*

3.2.2 Cas des formes réelles

Du théorème 3.2 on déduit que le nombre de classes B_Δ est exactement le nombre de cycles de $\mathfrak{F}_\Delta$.

Corollaire 3.2 *Si le discriminant Δ est positif alors B_Δ est exactement le nombre de cycles de $\mathfrak{F}_\Delta$.*

Exemple 3.4 *Il y exactement 4 cycles de discriminant $\Delta = 105$. Chacun de ces cycles contient un nombre fini de formes réduites. On a $\mathrm{B}_{105} = 4$.*

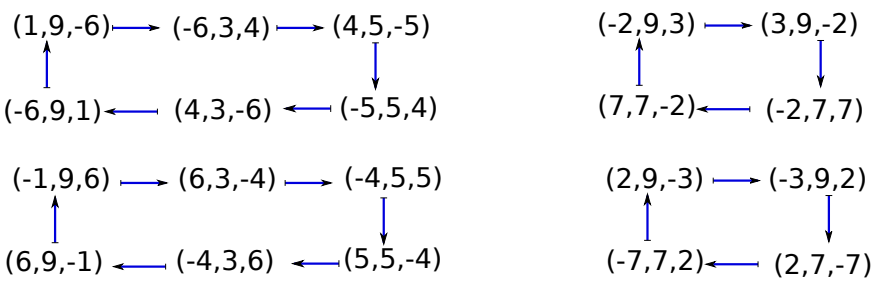


FIG. 3.5 – *Les 4 uniques cycles de discriminant* $\Delta = 105$

3.3 Tester l'équivalence entre deux formes

Nous introduisons le problème décisionnel $\mathcal{P}_\sim$ qui étant données deux formes f et g de $\mathfrak{F}_\Delta$ retourne "Vrai" si $f \sim g$, et "Faux" si f et g ne sont pas équivalentes.

On a donc $\mathcal{P}_\sim(f,g)$ ="Vrai" si et seulement si $\overline{f} = \overline{g}$.

Et puisque chaque classe contient un nombre fini de représentantes réduites, une méthode pour tester l'équivalence entre deux formes f et g de $\mathfrak{F}_\Delta$ données, est de tester si l'ensemble fini des représentantes réduites de $\overline{f}$ est égal à l'ensemble fini des représentantes réduites de $\overline{g}$.

Nous séparons le cas où les formes f et g sont imaginaires, et le cas où elles sont réelles.

3.3.1 Équivalence entre deux formes imaginaires

On veut tester l'équivalence entre deux formes f et g de $\mathfrak{F}_\Delta$ qui sont imaginaires.

Puisque dans ce cas il existe une unique forme représentative réduite par classe, le problème décisionnel $\mathcal{P}_\sim$ avec f et g en entrées se résout par l'algorithme 3.

Algorithme 3 : ***TestEquivalenceImaginaire***

Entrée : f et g de $\mathfrak{F}_{\Delta<0}$
Sorties : $\mathcal{P}_\sim(f,g)$
1: $\tilde{f} \leftarrow$ ***RéductionGaussImaginaire***$(f.Tr(\mu_f))$
2: $\tilde{g} \leftarrow$ ***RéductionGaussImaginaire***$(g.Tr(\mu_g))$
3: **retourner** $\tilde{f} = \tilde{g}$

L'algorithme de réduction ***RéductionGaussImaginaire*** étant de complexité quadratique en la norme de son entrée, on peut facilement tester l'équivalence de deux formes imaginaires puisque la complexité de l'algorithme 3 est aussi quadratique : $O(\log(\max(\|f\|, \|g\|))^2)$.

3.3.2 Équivalence entre deux formes réelles

On veut tester l'équivalence entre deux formes f et g de $\mathfrak{F}_\Delta$ qui sont réelles.

Dans ce cas il existe un unique cycle de formes représentatives réduites par classe, le problème décisionnel $\mathcal{P}_\sim$ avec f et g en entrées se résout par l'algorithme 4.

Algorithme 4 : ***TestEquivalenceRéelle***

Entrée : f et g de $\mathfrak{F}_{\Delta>0}$
Sorties : $\mathcal{P}_\sim(f,g)$

1: $\tilde{f} \leftarrow$ ***RéductionGaussRéelle***$(f.Tr(\nu_f))$
2: $\tilde{g} \leftarrow$ ***RéductionGaussRéelle***$(g.Tr(\nu_g))$
3: $\tilde{f}_0 \leftarrow \tilde{f}$
4: $\tilde{f} \leftarrow$ ***VoisineDroite***$(\tilde{f})$
5: **tantque** $\tilde{f} \neq \tilde{g}$ et $\tilde{f} \neq \tilde{f}_0$ **faire**
6: $\quad \tilde{f} \leftarrow$ ***VoisineDroite***$(\tilde{f})$ Parcourt le cycle de $\tilde{f}$
7: **fin tantque**
8: **retourner** $\tilde{f} = \tilde{g}$

Rappelons que d'après la proposition 2.3 le cardinal du cercle de réduites d'une forme réelle primitive $f \in \mathfrak{F}_\Delta$ est en $O(\log(\epsilon_{\Delta_f}))$ où ϵ_{Δ_f} est l'unité fondamentale. Et puisque généralement ϵ_{Δ_f} est de taille exponentielle en Δ_f (proposition 2.2), pour de grands discriminants Δ_f on est généralement incapable en pratique de parcourir l'ensemble d'un cycle de réduite d'une forme réelle $f \in \mathfrak{F}_\Delta$.

3.4 Forme, classe et cycle principal

Nous verrons que la forme principale joue un rôle particulier.

3.4.1 Forme principale

La *forme principale* de discriminant Δ_f est la réduite de la forme $(1, 1, (1-\Delta_f)/4)$ ou la forme $(1, 0, -\Delta_f/4)$ suivant la parité du discriminant. Dès que l'on connaît la valeur d'un discriminant Δ_f on peut donc construire cette forme.

Lorsque la forme principale est imaginaire, on remarque que $(1, 1, (1-\Delta_f)/4)$ ou $(1, 0, -\Delta_f/4)$ est déjà réduite.

Sinon la forme est réelle, et la forme principale est $(1, b, *)$ avec $b \in \left\{\lfloor\sqrt{\Delta_f}\rfloor, \lfloor\sqrt{\Delta_f}\rfloor - 1\right\}$ qui vérifie que $b^2 - \Delta_f$ soit divisible par 4. On remarquera que cette forme est la seule forme réelle réduite de premier coefficient égal à 1 (il suffit d'appliquer la définition de la réduction).

3.4.2 Classe principale

La *classe principale* de discriminant Δ_f est la classe contenant la forme principale.

Proposition 3.5 *La classe principale est la seule classe contenant la forme de premier ou dernier coefficient égale à* 1.

En effet l'action des translations $\boldsymbol{T}(h)$ avec $h \in \mathbb{Z}$ sur la forme principale donne l'ensemble de toutes les formes s'écrivant $(1, b, {}^{(b^2-\Delta_f)}/_4)$ pour tout entier b de même parité que le discriminant, puis l'action de $\boldsymbol{SE}$ sur ces formes donne l'ensemble de toutes les formes s'écrivant $({}^{(b^2-\Delta_f)}/_4, -b, 1)$. La classe de la forme principale contient donc toutes les formes de premier ou dernier coefficient égal à 1. On peut conclure en rappelant que l'ensemble des SL_2-orbites forment une partition de $\mathfrak{F}_{\Delta_f}$.

3.4.3 Cycle principal

On définit le *cycle principal* $\mathbb{1}_{\Delta_f}$ comme étant le cycle de réduites de la classe principale.

On remarquera que le cycle principal contient la forme principale.

3.5 Automorphisme fondamental de forme réelle

L'automorphisme fondamental d'une forme réelle f de $\mathfrak{F}_\Delta$ est aussi en relation avec le cycle principal de $\mathfrak{F}_\Delta$ car en parcourant le cycle principal on peut extraire la solution fondamentale (x_ϵ, y_ϵ) de l'équation de Pell de Δ (voir proposition1.7).

3.5.1 Cycle et automorphisme fondamental

On donne la relation entre l'automorphisme fondamental d'une forme réelle réduite, et son cycle de réduites (cette relation est donnée en paragraphe 6.14 de [3]).

Théorème 3.4 *L'automorphisme fondamental, noté U_f, d'une forme réelle réduite $f \in \mathfrak{F}_{\Delta_f}$ est la matrice obtenue en multipliant successivement les matrices de passage générées en parcourant le cycle de f en commençant par f est en finissant lorsque l'on revient sur la forme de départ.*

La démonstration de ce théorème se trouve dans la démonstration de la proposition 6.13.2 de [8] et proposition 6.10.3.

La construction d'un automorphisme fondamental $\mathcal{U}_f$ d'une forme réelle $f \in \mathfrak{F}_\Delta$ se fait en deux étapes.

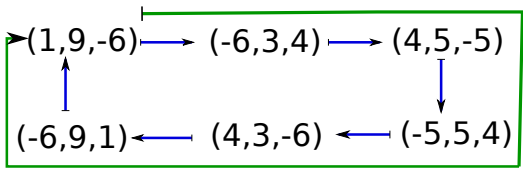


FIG. 3.6 – *Automorphisme fondamental et cycle de réduites d'une forme réduite*

Cette figure illustre l'automorphisme fondamental U_f *(schématisé par la flèche verte) obtenu en parcourant le cycle de réduites de* $(1, 9, -6)$ *(voir la figure 3.2).*
Cet automorphisme est
$U_f = \begin{pmatrix} 0 & 1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ -1 & -1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ -1 & -1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ -1 & -9 \end{pmatrix}$
on trouve $U_f = \begin{pmatrix} -5 & -48 \\ -8 & -77 \end{pmatrix}$.

1. Extraire (x_ϵ, y_ϵ) de $\mathbb{1}_\Delta$
   D'après le théorème 3.4, l'automorphisme fondamental $\mathcal{U}_g$ de la forme principale réduite $(1, b', c')$ de discriminant Δ est la matrice obtenue après avoir parcouru le cycle principal en commençant par $(1, b', c')$.
   Et puisque cette matrice s'écrit $\mathcal{U}_g = \begin{pmatrix} (x_\epsilon - b'y_\epsilon)/2 & -c'y_\epsilon \\ y_\epsilon & (x_\epsilon + b'y_\epsilon)/2 \end{pmatrix}$, on peut en extraire la solution fondamentale (x_ϵ, y_ϵ).
2. Construire $\mathcal{U}_f$
   L'automorphisme fondamental de $f = (a, b, c)$ s'écrit $\mathcal{U}_f = \begin{pmatrix} (x_\epsilon - by_\epsilon)/2 & -cy_\epsilon \\ ay_\epsilon & (x_\epsilon + by_\epsilon)/2 \end{pmatrix}$.

Cette méthode reste malgré tout irréalisable en pratique car la solution fondamentale (x_ϵ, y_ϵ) est de taille exponentielle en Δ (proposition 2.2).

3.5.2 Norme de l'automorphisme fondamental

Nous allons maintenant étudier la norme de l'automorphisme fondamental d'une forme réelle $f = (a, b, c) \in \mathfrak{F}_\Delta$. On note $\mathcal{U}_f$ l'automorphisme fondamental de f.

Cette étude ne se trouve pas dans la littérature classique sur les formes, mais pour étudier de façon optimale la répartition des formes réelles (dans la section 6.2) nous avons besoin de majorer au mieux la norme de $\mathcal{U}_f$.

Pour obtenir cette majoration on utilise la décomposition spectrale (voir [30, 19]) de l'automorphisme $\mathcal{U}_f$, cet automorphisme s'écrit $\mathcal{U}_f = ((x_\epsilon - by_\epsilon)/2, -cy_\epsilon; ay_\epsilon, (x_\epsilon + by_\epsilon)/2)$.

Proposition 3.6 *L'automorphisme fondamental de toute forme réelle* $f \in \mathfrak{F}_\Delta$ *s'écrit*

$$\mathcal{U}_f = \epsilon_\Delta \Pi_1 + \epsilon_\Delta^{-1} \Pi_2,$$

où Π_1 *et* Π_2 *sont les deux projecteurs spectraux*

$$\Pi_1 = \frac{\mathcal{U}_f - \epsilon_\Delta^{-1}\mathcal{I}_2}{\epsilon_\Delta - \epsilon_\Delta^{-1}} \text{ et } \Pi_2 = \frac{\mathcal{U}_f - \epsilon_\Delta \mathcal{I}_2}{\epsilon_\Delta^{-1} - \epsilon_\Delta}.$$

Démonstration. Comme le polynôme caractéristique de $\mathcal{U}_f$ est scindé (il s'écrit $P_{\mathcal{U}_f}(X) = (X - \epsilon_\Delta)(X - \epsilon_\Delta^{-1})$), alors l'automorphisme $\mathcal{U}_f$ est diagonalisable et décomposable en les deux projecteurs spectraux Π_1 et Π_2. □

On déduit de cette proposition l'écriture d'une puissance entière de $\mathcal{U}_f$ en fonction de ces deux projecteurs spectraux.

Corollaire 3.3 *Pour toute puissance entière k, l'automorphisme fondamental d'une forme réelle $f \in \mathfrak{F}_\Delta$ s'écrit*

$$\mathcal{U}_f^k = \epsilon_\Delta^k \Pi_1 + \epsilon_\Delta^{-k} \Pi_2,$$

où Π_1 *et* Π_2 *sont les deux projecteurs spectraux de* $\mathcal{U}_f$.

Démonstration. La proposition 3.6 nous permet d'écrire $\mathcal{U}_f = \epsilon\Pi_1 + \epsilon^{-1}\Pi_2$. Par définition des projecteurs spectraux, les projecteurs Π_1 et Π_2 vérifient :
– $\Pi_1 + \Pi_2 = \mathcal{I}_2$
– $\Pi_i^2 = \Pi_i$
– $\Pi_1\Pi_2 = 0 = \Pi_2\Pi_1$.
Ce qui prouve le corollaire. □

Nous obtenons la majoration de la norme d'une puissance entière de $\mathcal{U}_f$ suivante :

Proposition 3.7 *L'automorphisme fondamental $\mathcal{U}_f$ d'une forme réelle $f = (a, b, c) \in \mathfrak{F}_\Delta$ vérifie pour toute puissance entière k*

$$\|\mathcal{U}_f^k\| \leqslant 2 \cdot \frac{\max(|a|, (|b| + \sqrt{\Delta})/2), |c|)}{\sqrt{\Delta}} (\epsilon_\Delta^k + \epsilon_\Delta^{-k}) \ .$$

Démonstration. On calcule l'expression formelle des deux projecteurs de $\mathcal{U}_f$ en utilisant la proposition 3.6 :

$$\Pi_1 = \frac{\mathcal{U}_f - \epsilon_\Delta^{-1}\mathcal{I}_2}{\epsilon_\Delta - \epsilon_\Delta^{-1}} \qquad \Pi_2 = \frac{\mathcal{U}_f - \epsilon_\Delta \mathcal{I}_2}{\epsilon_\Delta^{-1} - \epsilon_\Delta}$$

$$\Pi_1 = \frac{1}{y_\epsilon\sqrt{\Delta}} \cdot \begin{pmatrix} y_\epsilon(\sqrt{\Delta} - b)/2 & -cy_\epsilon \\ ay_\epsilon & y_\epsilon(\sqrt{\Delta} + b)/2 \end{pmatrix} \quad \Pi_2 = \frac{-1}{y_\epsilon\sqrt{\Delta}} \cdot \begin{pmatrix} -y_\epsilon(\sqrt{\Delta} + b)/2 & -cy_\epsilon \\ ay_\epsilon & y_\epsilon(b - \sqrt{\Delta})/2 \end{pmatrix}$$

$$\Pi_1 = \frac{1}{\sqrt{\Delta}} \cdot \begin{pmatrix} (\sqrt{\Delta} - b)/2 & -c \\ a & (\sqrt{\Delta} + b)/2 \end{pmatrix} \qquad \Pi_2 = \frac{-1}{\sqrt{\Delta}} \cdot \begin{pmatrix} -(\sqrt{\Delta} + b)/2 & -c \\ a & (b - \sqrt{\Delta})/2 \end{pmatrix}$$

Il vient $\|\Pi_1\| = \|\Pi_2\|$ et

$$\begin{aligned}\|\Pi_1\| &= \frac{\max(|\sqrt{\Delta} - b|/2, |c|, |a|, |\sqrt{\Delta} + b|/2)}{\sqrt{\Delta}} \\ &\leqslant \frac{\max(|c|, |a|, (|b| + \sqrt{\Delta})/2)}{\sqrt{\Delta}}.\end{aligned}$$

On conclut en appliquant le corollaire 3.3 et la propriété sur les normes 1.5 :

$$\|\!|\mathcal{U}_f^k|\!\| \leqslant \epsilon_\Delta^k \|\!|\Pi_1|\!\| + \epsilon_\Delta^{-k} \|\!|\Pi_2|\!\| \leqslant \epsilon_\Delta^k \cdot 2\|\Pi_1\| + \epsilon_\Delta^{-k} \cdot 2\|\Pi_2\|.$$

□

On déduit de la proposition 3.7 le corollaire 3.4 qui nous fournit une majoration de la norme d'une puissance entière $\mathcal{U}_f$ dans le cas où la forme f est réduite.

Corollaire 3.4 *L'automorphisme fondamental $\mathcal{U}_f$ d'une forme réelle réduite $f \in \mathfrak{F}_\Delta$ vérifie pour toute puissance entière k*

$$\|\!|\mathcal{U}_f^k|\!\| \leqslant 2 \cdot (\epsilon_\Delta^k + \epsilon_\Delta^{-k}) \ .$$

Démonstration. Si la forme réelle $f = (a, b, c)$ est réduite alors d'après la proposition 3.3 $|b|$ et $|a| + |c|$ sont majorés par $\sqrt{\Delta}$, ce qui entraîne que les normes de ces projecteurs spectraux $\|\Pi_1\|_\infty$ et $\|\Pi_2\|_\infty$ sont toutes les deux plus petites que 1. Dans ces conditions l'application de la proposition 3.7 prouve le corollaire. □

Chapitre 4

Plus petite SL_2-matrice de passage ou de réduction

Dans ce chapitre on s'intéresse à la norme des matrices de passage de SL_2 entre deux formes équivalentes.

En premier nous avons cherché à obtenir une matrice de réduction de norme minimale.

Dans le cas des formes imaginaires et $\Delta < -4$, l'algorithme (classique) de Gauss (algorithme 1) retourne l'unique matrice de réduction, et nous donnons une majoration de cette norme beaucoup plus fine que les seules bornes précédemment obtenues (voir [4]).

Dans le cas des formes réelles, l'algorithme de Gauss (algorithme 2) ne minimise pas la norme de la matrice de réduction. Nous avons construit un algorithme de réduction (algorithme 5) qui minimise la norme de la matrice de réduction qu'il retourne. Notre algorithme manipule une forme sous l'action de GL_2 puis se ramène par l'action de transformations élémentaires au cas classique (: une forme équivalente réduite).

Nous fournissons aussi un nouvel algorithme (algorithme 6) qui parcourt par la droite le cycle d'une forme réelle réduite, et qui a l'avantage de fournir une matrice de passage qui est le produit de $\boldsymbol{S}$ ou $\mathcal{I}_2$, suivi d'une matrice de coefficients entiers et positifs, suivi d'une symétrie $\boldsymbol{S}$ ou de $\mathcal{I}_2$.

Toujours dans le cas réel, puisqu'il existe une infinité de matrices de passage entre deux formes, nous savons qu'il en existe au moins une de norme minimale, et nous avons réussi à caractériser cette plus petite matrice.

Nous donnons donc la propriété que doit vérifier une SL_2-matrice de passage entre deux formes réelles pour que ce soit la plus petite matrice entre ces deux formes.

Ces travaux ont été effectués en collaboration avec Nicolas Gama, Maître de conférences à l'Université de Versailles Saint-Quentin-en-Yvelines. Il a donné lieu à une publication

dans la conférence internationale avec comité de relecture, Algorithmic Number Theory Symposium, en juillet 2010 [2].

4.1 Norme des matrices de réduction des formes imaginaires

Pour une forme $f = (a, b, c)$ normalisée donnée, on veut majorer les coefficients de la plus petite matrice $\mathcal{M} \in \mathrm{SL}_2$ vérifiant que $g = f.\mathcal{M}$ est réduite.

Dans le cas imaginaire et $\Delta_f < -4$, puisque le seul automorphisme d'une forme est $\pm\mathcal{I}_2$, la matrice de réduction est unique au signe près (proposition 1.11). On veut donc majorer les coefficients de cette unique matrice $\mathcal{M}$.

Le lemme 5.6.1 de [4] donne comme résultat $\|\mathcal{M}\| \leqslant \frac{2\max\{|a|,|c|\}}{\sqrt{|\Delta_f|}}$.

Nous avons amélioré ce résultat par le théorème suivant :

4.1.1 Théorème *Borne imaginaire*

Théorème 4.1 (Borne imaginaire) *Si $f = (a, b, c)$ est une forme imaginaire normalisée de discriminant $\Delta_f < -4$, et de matrice de réduction $\mathcal{M} = (\alpha, \beta; \gamma, \delta)$ qui vérifie $g = f.\mathcal{M} = (a_g, b_g, c_g)$, alors soit $\mathcal{M}$ est triviale ($\pm\mathcal{I}_2$ ou $\pm\boldsymbol{SE}$) soit elle satisfait les deux majorations :*

1) $\|\mathcal{M}\| \leqslant \dfrac{1}{\sqrt{3}}\sqrt{\dfrac{\max\{9|a|, 25|c|\}}{|a_g|}}$

2) $|\alpha\beta\gamma\delta|^{1/4} \leqslant 2|\gamma\delta|^{1/2} \leqslant 2 \cdot \dfrac{2}{3^{1/4}}\left(\dfrac{|ac|}{|\Delta_f|}\right)^{1/4}$.

Démonstration. En premier on remarque que le produit $\gamma\delta$ des coefficients de la dernière ligne de $\mathcal{M}$ est nul si et seulement si la matrice de réduction est triviale.

- Le premier sens est évident si la matrice de réduction est $\pm\mathcal{I}_2 = \pm(1, 0; 0, 1)$ ou $\pm\boldsymbol{SE} = \pm(0, 1; -1, 0)$ alors $\gamma\delta = 0$.
- Dans le deuxième sens si $\gamma = 0$ alors la matrice de réduction ne peut être que $\pm\mathcal{I}_2$ car sinon c'est une translation $\pm(1, h; 0, 1)$ pour un entier h non nul et comme f est déjà normalisée effectuer cette translation sur f aurait pour résultat une forme non normalisée et donc non réduite.
- De la même façon si $\delta = 0$ alors la matrice de réduction ne peut être que $\pm\boldsymbol{SE}$ car sinon c'est la transformation $(k, 1; -1, 0) = \boldsymbol{T}(-k)\boldsymbol{SE}$ pour un entier k non nul et comme f est déjà normalisée effectuer cette transformation sur f aurait pour résultat une forme non normalisée (si $f.\boldsymbol{T}(-k)$ n'est pas normalisée alors $f.\boldsymbol{T}(-k)\boldsymbol{SE}$ ne l'est pas non plus) et donc non réduite.

A partir d'ici on se place dans la cas où $\mathcal{M}$ n'est pas triviale : on a donc $\gamma\delta \neq 0$.
On a

$$|a_g| = |f(\alpha,\gamma)| = |a|\gamma^2 \left(\left({}^{\alpha}\!/_{\gamma} + \frac{b}{2a} \right)^2 + \frac{|\Delta_f|}{4a^2} \right) \tag{4.1}$$

$$|c_g| = |f(\beta,\delta)| = |c|\delta^2 \left(\left({}^{\beta}\!/_{\delta} + \frac{b}{2c} \right)^2 + \frac{|\Delta_f|}{4c^2} \right). \tag{4.2}$$

On voit facilement que 4.1 est minoré par $\frac{|\Delta_f|}{4|a|}\gamma^2$, alors que 4.2 est minoré par $\frac{|\Delta_f|}{4|c|}\delta^2$.

Ces deux minorations entrainent $\gamma^2 \leqslant \frac{4|aa_g|}{|\Delta_f|}$ et $\delta^2 \leqslant \frac{4|cc_g|}{|\Delta_f|}$.
On en déduit

$$|\gamma\delta| \leqslant 4\frac{\sqrt{|ac||a_g c_g|}}{|\Delta_f|} \leqslant \frac{4\sqrt{|ac|}}{\sqrt{3|\Delta_f|}},$$

car comme g est réduite on $3|a_g c_g| \leqslant |\Delta_f|$ (proposition 3.1), et on en déduit aussi

$$|\delta| \leqslant \frac{2}{\sqrt{3}}\sqrt{\frac{|c|}{|a_g|}} \text{ et } |\gamma| \leqslant \frac{2}{\sqrt{3}}\sqrt{\frac{|a|}{|c_g|}} \leqslant \frac{2}{\sqrt{3}}\sqrt{\frac{|a|}{|a_g|}},$$

car la proposition 3.1 donne $3|a_g c_g| \leqslant |\Delta_f|$.

Pour conclure on montre que si $\mathcal{M}$ n'est pas triviale alors la condition de normalisation sur f ($b \in]-|a|, |a|]$) entraîne que les coefficients de sa matrice de réduction vérifient $|\alpha| \leqslant {}^{3}\!/_{2}|\gamma|$ et $|\beta| \leqslant {}^{5}\!/_{2}|\delta|$:

- Puisque $|a_g| = \lambda(f)$ (le premier minimum de f vu dans la définition 1.11) on a $|a_g| \leqslant |a|$, puis en utilisant l'écriture 4.1 il vient $1 \geqslant ||{}^{\alpha}\!/_{\gamma}| - |\frac{b}{2a}||$. La forme f étant normalisée on a $|\frac{b}{2a}| \leqslant 1/2$ ce qui entraîne $|{}^{\alpha}\!/_{\gamma}| \leqslant {}^{3}\!/_{2}$.
- Ensuite en utilisant la formule du déterminant il vient ${}^{3}\!/_{2} \geqslant |{}^{\alpha}\!/_{\gamma}| = |{}^{1}\!/_{\gamma\delta} + {}^{\beta}\!/_{\delta}| \geqslant ||{}^{1}\!/_{\gamma\delta}| - |{}^{\beta}\!/_{\delta}||$ et comme γ et δ sont des entiers non nuls $|{}^{1}\!/_{\gamma\delta}| \leqslant 1$ ce qui entraîne $|{}^{\beta}\!/_{\delta}| \leqslant {}^{5}\!/_{2}$.
- On vérifie bien $(\alpha\beta\gamma\delta)^{1/4} \leqslant 2(\gamma\delta)^{1/2}$ et aussi le premier point du théorème.

□

Ainsi, pour un discriminant Δ_f fixé, la norme de la matrice de réduction est de façon asymptotique en $O\left(\sqrt{\|f\|/\sqrt{|\Delta_f|}}\right)$ et non en $O\left(\|f\|/\sqrt{|\Delta_f|}\right)$ comme le laissait penser l'analyse de l'algorithme de réduction de [4].

4.2 Plus petite matrice de réduction d'une forme réelle

Pour une forme $f = (a, b, c)$ normalisée donnée, on veut calculer la plus petite matrice de réduction $\mathcal{M} = (\alpha, \beta; \gamma, \delta)$ telle que $g = f.\mathcal{M}$ est réduite.

Dans le cas réel, on ne peut pas utiliser la preuve précédente (du cas imaginaire) car le terme $\left((\alpha/\gamma + \frac{b}{2a})^2 - \frac{|\Delta_f|}{4a^2}\right)$ peut être exponentiellement proche de 0 (figure 1.2). Le problème dans ce cas est que chaque cycle de réduites contient un grand (généralement exponentiel) nombre de formes réduites équivalentes, et certaines de ces formes sont exponentiellement loin de f (de coefficients exponentiels en $\|f\|$).

Pour construire une matrice de réduction de norme polynomiale il nous faut donc une approche constructive. L'analyse de l'algorithme (classique) de réduction de Gauss dans [4, 25, 33] prouve que la norme de la matrice de réduction calculée est asymptotiquement majorée par $O(\|f\|)$.

Notre objectif est de modifier cet algorithme afin que la norme de matrice de réduction soit asymptotiquement en $O\left(\sqrt{\|f\|/\sqrt{|\Delta_f|}}\right)$ (comme dans le cas imaginaire).

Pour ce faire nous avons introduit les notions de normalisation et de réduction primaire et secondaire, qui sont moins contraignantes que les définitions classiques (définition 3.4).

Ces conditions sont exprimées par rapport aux racines des formes, ce qui nous semble beaucoup plus intuitif (en utilisant la représentation affine des formes) que des conditions exprimées sur les coefficients des formes.

4.2.1 Normalisation primaire et secondaire

Nous introduisons les notions de normalisation primaire et secondaire.

Définition 4.1 *Une forme réelle f est :*
- *normalisée primaire lorsque* $0 < \zeta_f^+ < 1$
- *normalisée secondaire lorsque* $-1 < \zeta_f^- < 0$.

Ces notions peuvent être définies de manière équivalente avec les coefficients d'une forme.

Propriété 4.1 *Une forme $f = (a, b, c)$ est :*

$$\begin{array}{lcccccc} \textit{normalisée primaire} & \Leftrightarrow & \sqrt{\Delta_f} - 2|a| & < & \text{signe}(a)b & < & \sqrt{\Delta_f} \\ \textit{normalisée secondaire} & \Leftrightarrow & -\sqrt{\Delta_f} & < & \text{signe}(a)b & < & 2|a| - \sqrt{\Delta_f}. \end{array}$$

Pour le montrer on utilise simplement l'écriture des racines ζ_f^+ et ζ_f^- (définition 1.3) en fonction des coefficients de la forme f.

Remarque 4.1 *La normalisation primaire et la secondaire d'une forme f peuvent toujours être obtenues par l'action d'une translation $\boldsymbol{T}(h)$ avec h l'entier égal à $\lfloor \zeta_f^+ \rfloor$ ou $\lceil \zeta_f^- \rceil$. Pour le voir on prend en compte la définition de h et on utilise la représentation affine (définition 1.6) des formes réelles. Cette remarque est illustrée dans la figure 4.1.*

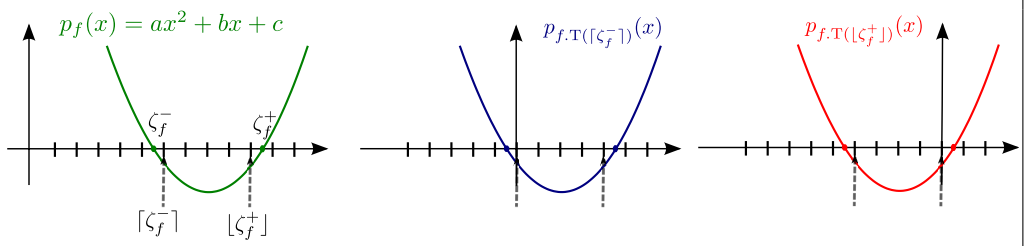


FIG. 4.1 – *Entiers normalisant de façon primaire ou secondaire une forme réelle f*

Cette figure illustre les représentations affines des formes réelles f, $f.\boldsymbol{T}\lceil \zeta_f^- \rceil$ et $f.\boldsymbol{T}\lfloor \zeta_f^+ \rfloor$. Par définition des entiers $\lceil \zeta_f^- \rceil$ et $\lfloor \zeta_f^+ \rfloor$ (qui existent pour toutes formes réelles), la forme $f.\boldsymbol{T}\lceil \zeta_f^- \rceil$ est normalisée secondaire et la forme $f.\boldsymbol{T}\lfloor \zeta_f^+ \rfloor$ normalisée primaire.

Passons aux relations entre la normalisation classique et la normalisation primaire ou secondaire.

Proposition 4.1 *Une forme normalisée classique est normalisée primaire ou secondaire.*

Cette proposition montre bien que les conditions de normalisation classique sont plus restrictives que les conditions de normalisation primaire ou secondaire.

Avant de faire la démonstration de ce théorème, nous prouvons le lemme ci-dessous.

Lemme 4.1 *Une forme réelle normalisée $f = (a, b, c)$ avec $|a| < \sqrt{\Delta_f}$ vérifie*

$$0 < \frac{-b + \sqrt{\Delta_f}}{2|a|} < 1.$$

Démonstration. Puisque $|a| < \sqrt{\Delta_f}$ et que f est normalisée on a :

$$\sqrt{\Delta_f} - 2|a| < b < \sqrt{\Delta_f}\ .$$

Si $b > 0$ il vient $\sqrt{\Delta_f} - 2|a| < |b| < \sqrt{\Delta_f}$ et donc $0 < \dfrac{-|b| + \sqrt{\Delta_f}}{2|a|} < 1$. Dans l'autre cas $b < 0$ et il vient $0 < \dfrac{|b| + \sqrt{\Delta_f}}{2|a|} < 1$. Dans les deux cas la forme f vérifie bien les inégalités de l'énoncé. □

La démonstration de la proposition 4.1 ci-dessous est illustrée par la figure 4.2.

Démonstration. Soit $f = (a, b, c)$ une forme normalisée classique.
Dans le cas où $|a| \geqslant \sqrt{\Delta_f}$, les valeurs absolues des deux racines réelles de f sont majorées par $\frac{|b|+\sqrt{\Delta_f}}{2|a|}$, et puisque la forme est normalisée $\frac{|b|+\sqrt{\Delta_f}}{2|a|} \leqslant \frac{2|a|}{2|a|}$. Les deux racines de la forme sont donc strictement comprises entre -1 et 1 (sinon le discriminant est un carré parfait), ce qui prouve que la forme est normalisée primaire ou secondaire.
Étudions le cas où $|a| < \sqrt{\Delta_f}$. Les deux racines de f s'écrivent

$$\zeta^- = \frac{-\text{signe}(a)b - \sqrt{\Delta_f}}{2|a|} \text{ et } \zeta^+ = \frac{-\text{signe}(a)b + \sqrt{\Delta_f}}{2|a|}.$$

En utilisant le lemme 4.1 on en déduit que dans ce cas, si $a > 0$ la forme est normalisée primaire : $0 < \zeta^+ < 1$. Sinon la forme est normalisée secondaire : $-1 < \zeta^- < 0$.

□

On remarquera bien que la réciproque n'est pas vraie en général : une forme normalisée secondaire ou primaire n'est pas forcément une forme normalisée classique.
Par exemple dans le cas où la distance entre les racines d'une forme est $\leqslant 1$ la normalisation classique entraîne que la moyenne des racines de la forme est dans $]-1/2, 1/2]$ alors que notre condition de normalisation primaire ou secondaire implique seulement que la moyenne des racines est dans $]-1, 1[$.

4.2.2 Réduction primaire et secondaire

Nous introduisons les notions de réduction primaire et secondaire, ainsi que la notion de réduction large.

Définition 4.2 *Une forme réelle f est :*
– *réduite primaire si elle est normalisée primaire et si en plus on a $\zeta_f^- < -1$:*

$$\zeta_f^- < -1 < 0 < \zeta_f^+ < 1$$

– *réduite secondaire si elle est normalisée secondaire et si en plus on a $1 < \zeta_f^+$:*

$$-1 < \zeta_f^- < 0 < 1 < \zeta_f^+.$$

Finalement f est réduite large si elle est réduite primaire ou secondaire.

Remarque 4.2 *Les deux propriétés de réduite primaire et de réduite secondaire sont échangées par l'action de $\boldsymbol{E}$, car $\boldsymbol{E}$ inverse les racines comme l'indique la propriété 1.8.*

Ces notions peuvent être définies de manière équivalente avec les coefficients d'une forme.

Propriété 4.2 *Une forme $f = (a, b, c)$ est :*

$$\begin{array}{lcccccc} \textit{réduite primaire} & \Leftrightarrow & |\sqrt{\Delta_f} - 2|a|| & < & \text{signe}(a)b & < & \sqrt{\Delta_f} \\ \textit{réduite secondaire} & \Leftrightarrow & -\sqrt{\Delta_f} & < & \text{signe}(a)b & < & -|\sqrt{\Delta_f} - 2|a||. \end{array}$$

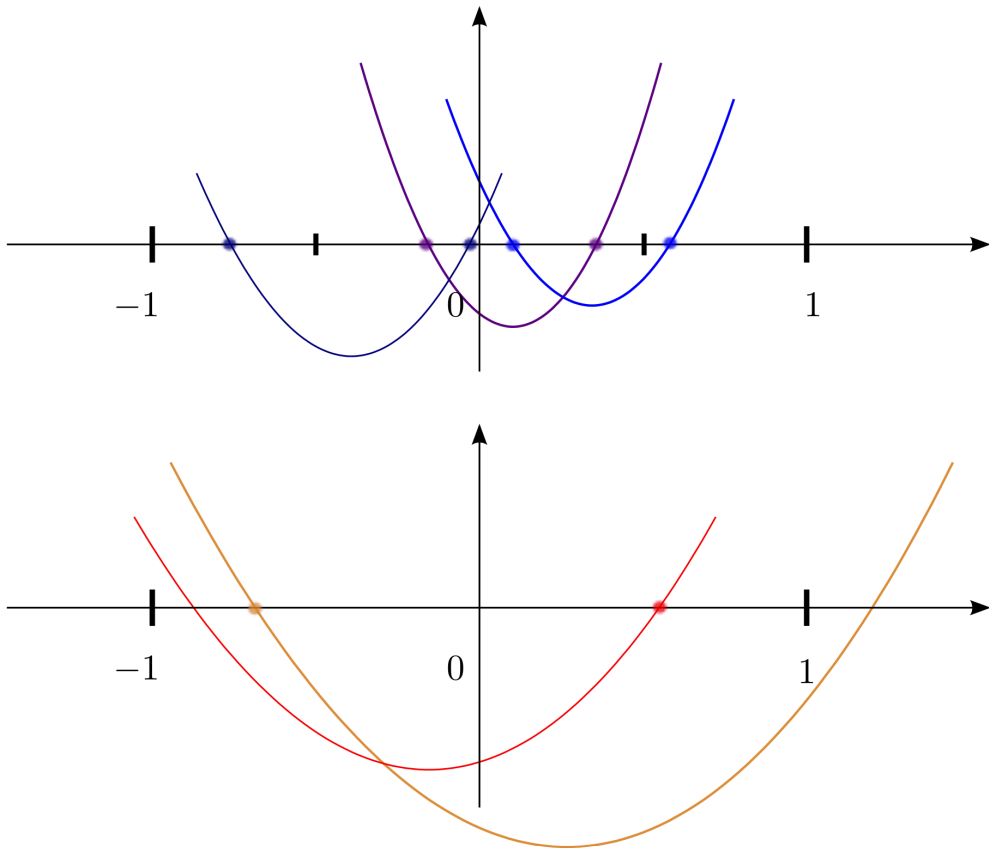


FIG. 4.2 – *Représentation affine de formes normalisées classiques*

Cette figure illustre la démonstration de la proposition 4.1. Elle représente la représentation affine des cinq types de formes normalisées classiques possibles en fonction de la position de leurs racines. Le repère du haut est le cas où la distance entre les racines des formes normalisées est $\leqslant 1$*, le repère du bas le cas où la distance entre les racines des formes normalisées est* > 1*.*

Démonstration. Pour le montrer on utilise la propriété 4.1 et on remarque que :

$$\begin{array}{rcccl} \zeta_f^- < -1 & \Leftrightarrow & -(\sqrt{\Delta_f} - 2|a|) & < & \text{signe}(a)b \\ 1 < \zeta_f^+ & \Leftrightarrow & \text{signe}(a)b & < & -(2|a| - \sqrt{\Delta_f}). \end{array}$$

□

Comme précédemment nous pouvons mettre en relation, la réduction classique et la réduction large.

Proposition 4.2 *Si* (a, b, c) *est une forme réduite alors elle est réduite large et on a* $b > 0$ *:*
– *si* $a > 0$ *la forme est réduite primaire,*

– *sinon la forme est réduite secondaire.*

Démonstration. Si $f = (a, b, c)$ est une forme réduite classique nous avons :

$$|\sqrt{\Delta_f} - 2|a|| < b < \sqrt{\Delta_f}.$$

Ainsi le coefficient b est positif.
Donc si $a > 0$ la forme réduite est réduite primaire, et si $a < 0$ la forme est réduite secondaire (on utilise la propriété 4.2). □

Cette proposition montre bien que les conditions de réduction classique sont plus contraignantes que les conditions de réduction primaire ou secondaire.

Cette fois, nous pouvons mettre en relation la réduction large avec la réduction classique.

Corollaire 4.1 *Une forme $f = (a, b, c)$ est réduite large si et seulement si :*

$$|\sqrt{\Delta_f} - 2|a|| < |b| < \sqrt{\Delta_f}\ .$$

Ce corollaire est une simple réécriture de la propriété 4.2 dans laquelle on a intégré le signe de a.

Ce corollaire montre que toute forme réduite large peut être transformée en une forme réduite (classique) par l'action de $\boldsymbol{S}$ (qui change le signe du deuxième coefficient de la forme) ou de $\mathcal{I}_2$ (la condition de réduction d'une forme réelle est $|\sqrt{\Delta_f} - 2|a|| < b < \sqrt{\Delta_f}$).

Nous faisons la remarque importante suivante :

Remarque 4.3 *Les deux notions de primaire et secondaire (normalisation et réduction) sont échangées par l'action de $\boldsymbol{S}$ puisque la symétrie change le signe des racines comme le montre la propriété 1.8.*

4.2.3 Problème de la plus petite (GL_2 et SL_2)-matrice de réduction

La grande partie de notre contribution est d'avoir résolu les problèmes suivants :

Lemme 4.2 *Le problème 2 est au moins aussi difficile que le problème 1.*

1. **Plus petite SL_2-matrice** *Étant donnée une forme réelle normalisée classique f, trouver $\mathcal{M} \in SL_2$ telle que $f.\mathcal{M}$ est réduite classique et $\|\mathcal{M}\|$ est minimale.*
2. **Plus petite GL_2-matrice** *Étant donnée une forme réelle normalisée primaire f, trouver $\mathcal{M} \in GL_2$ telle que $f.\mathcal{M}$ est réduite large et $\|\!|\mathcal{M}|\!\| \leqslant \sqrt{\epsilon_\Delta}$.*

Démonstration. Nous faisons la réduction polynomiale (section 2.4.2) du *Problème 1* au problème *Problème 2*. Soit $g = (a, b, c)$ une entrée du problème *Problème 1*.
On va montrer qu'en faisant l'hypothèse que le problème *Problème 2* se résolve en temps polynomial, alors on peut résoudre le *Problème 1*.

La forme g est normalisée primaire ou secondaire (car elle est normalisée classique), donc parmi les formes g est $g.\boldsymbol{S}$ exactement une forme est normalisée primaire. Si $\mathcal{A} \in \mathrm{GL}_2$ est la solution du *Problème 2* ayant en entrée la forme normalisée primaire équivalente à g, alors l'action de $\mathcal{I}_2\mathcal{A}$ ou $\boldsymbol{S}\mathcal{A}$ (suivant que g ou $g.\boldsymbol{S}$ est normalisée primaire) sur la forme g est une forme réduite large. On en déduit une solution du *Problème 1* ayant pour entrée g car on multiplie la matrice précédente ($\mathcal{I}_2\mathcal{A}$ ou $\boldsymbol{S}\mathcal{A}$) par $\mathcal{I}_2$ ou $\boldsymbol{S}$ pour s'assurer que le deuxième coefficient de la forme réduite large est positif, et enfin on multiplie par $\mathcal{I}_2$ ou $\boldsymbol{E}$ de façon à obtenir une matrice de déterminant égal à $+1$: on note $\mathcal{B}$ la matrice ainsi obtenue.
Puisque $\mathcal{I}_2$, $\boldsymbol{S}$ et $\boldsymbol{E}$ sont des matrices de permutation, elles ne changent pas les normes $|||\cdot|||$ ou $\|\cdot\|$ on a donc :

$$|||\mathcal{A}||| \leqslant \sqrt{\epsilon_\Delta}, \text{ et } |||\mathcal{A}||| = |||\mathcal{B}|||,$$

ce qui implique qu'on a aussi $|||\mathcal{B}||| \leqslant \sqrt{\epsilon_\Delta}$ donc le théorème 4.4 (section 4.6) entraîne que $\|\mathcal{B}\|(\leqslant |||\mathcal{B}|||)$ est minimale pour le problème *Problème 1.*

On remarquera que la réduction du *Problème 1* au problème *Problème 2* préserve aussi la valeur absolue du produit des coefficients de chaque ligne de la matrice de réduction.

□

Ce lemme 4.2 nous a motivé pour construire un algorithme de réduction résolvant le *Problème 2* moins restrictif, puiqu'en utilisant les matrices $\boldsymbol{S}$ et $\boldsymbol{E}$ on peut obtenir les notions classiques de réduction et une matrice de SL_2.

4.3 Nouvel algorithme de réduction de formes réelles

Nous avons modifié l'algorithme classique de réduction d'une forme réelle f afin que la norme de la matrice de réduction devienne asymptotiquement en $O\left(\sqrt{\|f\|/\sqrt{|\Delta_f|}}\right)$ (comme dans le cas imaginaire), au lieu de $O(\|f\|)$.

De plus nous avons vérifié que cette borne était très fine même dans le cas moyen.

Nous définissons les deux entiers particuliers suivants :

Définition 4.3 *Pour toute forme réelle f, nous définissons les deux entiers h_f^+ et h_f^- comme étant*

$$h_f^+ = \left\lfloor \zeta_f^+ \right\rfloor \text{ et } h_f^- = \left\lceil \zeta_f^- \right\rceil.$$

D'après la remarque 4.1 et la figure 4.1 les entiers h_f^+ et h_f^- sont respectivement les uniques entiers tels que $f.\boldsymbol{T}(h_f^+)$ est normalisée primaire, et $f.\boldsymbol{T}(h_f^-)$ est normalisée secondaire.

Propriété 4.3 *Pour toute forme réelle $f = (a, b, c)$, les deux entiers h_f^+ et h_f^- s'écrivent*

$$h_f^+ = \left\lfloor \frac{-\text{signe}(a)b + \sqrt{\Delta_f}}{2|a|} \right\rfloor \text{ et } h_f^- = -\left\lfloor \frac{\text{signe}(a)b + \sqrt{\Delta_f}}{2|a|} \right\rfloor.$$

Démonstration. Les deux racines ζ_f^+ et ζ_f^- de la forme f s'écrivent :

$$\zeta_f^+ = \frac{-\text{signe}(a)b + \sqrt{\Delta_f}}{2|a|} \text{ et } \zeta_f^- = -\frac{\text{signe}(a)b + \sqrt{\Delta_f}}{2|a|}.$$

Et comme pour tout réel x et y : $\lceil x/|y| \rceil = -\lfloor -x/|y| \rfloor$ il vient bien la propriété 4.3. □

Parmi les deux entiers h_f^-, h_f^+ celui de plus petite valeur absolue est noté $\boldsymbol{h}(f)$.

Définition 4.4 *Pour une forme réelle f, nous définissons l'entier $\boldsymbol{h}(f)$ comme étant*

$$\boldsymbol{h}(f) = h_f^+ \text{ si } |h_f^+| < |h_f^-|, \text{ et } \boldsymbol{h}(f) = h_f^- \text{ sinon.}$$

En d'autre mots, $\boldsymbol{h}(f)$ est la *plus courte normalisation* de f.

En comparaison, il y a un seul entier ν_f (définition 3.5) dans le cas classique tel que $f.\boldsymbol{T}(\nu_f)$ est normalisée, ν_f est l'un des entiers h_f^-, h_f^+ mais pas forcément celui de plus petite valeur absolue : dans la cas où $|a| \geqslant \sqrt{\Delta_f}$ on a $\nu_f \in \{h_f^-, h_f^+\}$ et c'est le plus proche de la moyenne des racines de f, dans le cas où $|a| < \sqrt{\Delta_f}$ on utilise la propriété 4.3 et il vient que, si $a > 0$ alors $\nu_f = h_f^+$ sinon $\nu_f = h_f^-$.

Notre algorithme de réduction, est une variante de l'algorithme de réduction de Gauss qui opère dans GL_2. L'algorithme alterne l'échange $\boldsymbol{E}$ et la plus courte normalisation $\boldsymbol{T}(\boldsymbol{h}(f))$ à chaque boucle, et termine sur une forme réduite large.

Comme nous le verrons dans les sous-sections suivantes, effectuer les normalisations par h_f^- ou h_f^+ nous amène dans tous les cas à une forme réduite large. Mais le choix de la plus courte normalisation $\boldsymbol{h}(f)$ au lieu de la classique ν_f (surtout à la dernière étape) est d'une très grande importance pour minimiser la matrice de réduction.

Comme on le précise dans [28] l'algorithme original de Gauss de 1801 utilisait la normalisation la plus grande à chaque itération.Dans ce cas le nombre d'itérations pouvait être exponentiel sur certaines entrées. C'est Lagarias qui introduisit la normalisation classique pour obtenir une complexité quadratique.

Algorithme 5 : ***RéductionRéelle*GL2**

Entrée : $f = (a, b, c)$ une forme normalisée primaire
Sorties : $f.\mathcal{M}$ une forme réduite large et $\mathcal{M} \in \mathrm{GL}_2$

1: $\mathcal{M} \leftarrow \mathcal{I}_2$
2: **tantque** f n'est pas réduite large **faire**
3: $\quad f \leftarrow f.\boldsymbol{E}$ et $\mathcal{M} \leftarrow \mathcal{M}\boldsymbol{E}$ — Étape d'échange
4: $\quad f \leftarrow f.\boldsymbol{T}(\boldsymbol{h}(f))$ et $\mathcal{M} \leftarrow \mathcal{M}\boldsymbol{T}(\boldsymbol{h}(f))$ — Étape de normalisation
5: **fin tantque**
6: **retourner** f et $\mathcal{M}$

On remarquera que pendant l'exécution de l'algorithme 5, effectuer la plus courte normalisation d'une forme $f = (a, b, c)$ ne dépend pas forcément du signe de a.
Par exemple dans le cas où f vérifie $1 \leqslant h_f^- < h_f^+$ la plus courte normalisation est h_f^- quelque soit le signe de a alors que si f vérifie $h_f^+ < h_f^-$ la plus courte normalisation de f est dans ce cas h_f^+.

4.3.1 Théorème *Borne réelle*

Le résultat principal de ma thèse est le théorème suivant, sur la qualité de la sortie de notre algorithme, qui est l'analogue dans le cas réel du théorème 4.1.

Théorème 4.2 (Borne réelle) *Soit $f = (a, b, c)$ une forme réelle normalisée primaire. Avec f en entrée, l'algorithme **RéductionRéelleGL2** termine après au plus* $\left(\dfrac{\log\left(|a|/\sqrt{\Delta}\right)}{2 \cdot \log(\omega)} + 4\right)$ *itérations où* $\omega = \dfrac{1+\sqrt{5}}{2}$ *est le nombre d'or. Et sa sortie* $\mathcal{M} = (\alpha, \beta; \gamma, \delta)$ *et* $f_r = f.\mathcal{M} = (a_r, b_r, c_r)$ *vérifient :*

1) $\| \mathcal{M} \| \leqslant 4\sqrt{|a|/|a_r|}$

2) $(|\alpha\beta\gamma\delta|)^{1/4} \leqslant |\gamma\delta|^{1/2} \leqslant \sqrt{21}\sqrt{|a|/\sqrt{\Delta}}$.

Avant de prouver ce théorème, nous faisons remarquer que les meilleures bornes connues sur l'algorithme classique de Gauss dans les même conditions (voir le théorème 4.4 de [4]) sont $\|\mathcal{M}\| \leqslant |a|(1+1/\Delta)$ et $|\gamma\delta|^{1/2} \leqslant \left(|a|/\sqrt{\Delta}\right)(1+1/\sqrt{\Delta})$, ce qui est asymptotiquement le carré de notre borne avec ***RéductionRéelle*GL2**

L'inégalité triangulaire appliquée sur la norme de la représentation polaire de $f = f_r.\mathcal{M}^{-1}$ donne $\|\!|\mathcal{M}_f|\!\| \leqslant \|\!|(\mathcal{M}^{-1})^t|\!\|\,\|\!|\mathcal{M}_{f_r}|\!\|\,\|\!|\mathcal{M}^{-1}|\!\|$ d'où $\|\!|\mathcal{M}_f|\!\|/\|\!|\mathcal{M}_{f_r}|\!\| \leqslant 4\|(\mathcal{M}^{-1})^t\|\|\mathcal{M}^{-1}\|$ et comme $\|(\mathcal{M}^{-1})^t\| = \|\mathcal{M}^{-1}\| = \|\mathcal{M}\|$ ($\mathcal{M} \in \mathrm{SL}_2$) il vient

$$\sqrt{\|\!|f|\!\|/\|\!|f_r|\!\|} \leqslant 2\|\mathcal{M}\|,$$

ce qui confirme que notre théorème 4.2 est optimal dans le cas moyen.

L'analyse de l'algorithme de réduction de Gauss dans [4] majore son nombre d'itérations par $\left(\frac{\log(|a|/\sqrt{\Delta})}{2\cdot\log(2)}+2\right)$ étapes. Notre majoration du nombre d'itérations de ***RéductionRéelle*GL2** est fine dans le pire cas, et elle est seulement plus grande d'un facteur multiplicatif d'environ 1.4 par rapport au nombre maximal d'itérations de l'algorithme de Gauss. Cependant notre objectif en construisant ***RéductionRéelle*GL2** est de minimiser la taille de la matrice de réduction.

La figure 4.4 illustre un exemple de formes où l'algorithme de réduction de Gauss donne une matrice de réduction plus grande que la matrice fournie par notre algorithme ***RéductionRéelle*GL2**.

Proposition 4.3 *La famille de formes réelles s'écrivant*

$$F_n = (-n, b, 1)$$

avec un entier $n \geqslant 3$ et $b = \lfloor 2n/3 - 3/2 \rfloor$, est une famille de formes où l'algorithme de réduction ***RéductionGaussRéelle*** *donne une matrice de réduction $\sqrt{\Delta} - 3$ fois plus grande que la matrice fournie par notre algorithme* ***RéductionRéelleGL2*** *(avec Δ le discriminant de F_n).*

Démonstration. Montrons en premier que cette famille de forme vérifie :

$$-1 < \zeta^-_{F_n} < 0 < \zeta^+_{F_n} < 1 \text{ et } \zeta^-_{F_n.\boldsymbol{E}} < -1 < 1 < \zeta^+_{F_n.\boldsymbol{E}} < 2.$$

- Puisque $n \geqslant 3$ l'entier b est positif et le discriminant vérifie $\Delta_{F_n} = b^2 + 4n \geqslant 4n \geqslant 12$, ce qui entraîne $\zeta^-_{F_n.\boldsymbol{E}} = \dfrac{-b - \sqrt{\Delta_{F_n}}}{2} < -1$ (car $b + \sqrt{\Delta_{F_n}} \geqslant 3$).
- De plus comme n est positif on a $\Delta_{F_n} = b^2 + 4n > b^2$, ce qui entraîne $-b + \sqrt{\Delta_{F_n}} > 0$ et donc $\zeta^+_{F_n.\boldsymbol{E}} > 0$.
- On en déduit que $-1 < \zeta^-_{F_n} < 0 < \zeta^+_{F_n}$ car $\boldsymbol{E}$ inverse les racines.
- Comme $n \geqslant 2$ on a :

$$\begin{aligned} -b + \sqrt{b^2+4n} &< -b + \sqrt{(2n/3 - 3/2)^2 + 4n} = -b + \sqrt{(2n/3 + 3/2)^2} \\ &< -(2n/3 - 3/2) + 1 + \sqrt{(2n/3 + 3/2)^2} = 4 \end{aligned}$$

  et

$$\begin{aligned} -b + \sqrt{b^2+4n} &> -b + \sqrt{(2n/3 - 5/2)^2 + 4n} = -b + \sqrt{4n^2/9 + 2n/3 + 25/4} \\ &> -b + \sqrt{4n^2/9 + 2n/3 + 1/4} = -b + \sqrt{(2n/3 + 1/2)^2} \\ &> -2n/3 + 3/2 + 2n/3 + 1/2 = 2. \end{aligned}$$

- Il vient $1 < \zeta^+_{F_n.\boldsymbol{E}} = \dfrac{-b + \sqrt{\Delta_{F_n}}}{2} < 2$, et donc $0 < 1/2 < \zeta^+_{F_n} < 1$.

Nous avons bien montré :

$$-1 < \zeta^-_{F_n} < 0 < \zeta^+_{F_n} < 1 \text{ et } \zeta^-_{F_n.\boldsymbol{E}} < -1 < 1 < \zeta^+_{F_n.\boldsymbol{E}} < 2.$$

En s'aidant de la représentation affine des formes, ces deux inégalités impliquent :

- $F_n = (-n, b, 1)$ est normalisée primaire et secondaire
- F_n n'est pas réduite (ni classique ni large)
- $1 = h^+_{F_n.\boldsymbol{E}}$ est la plus courte normalisation de $F_n.\boldsymbol{E}$
- $F_n.\boldsymbol{ET}(1)$ est réduite primaire
- $-2 < \zeta^-_{F_n.\boldsymbol{SE}} < -1 < 1 < \zeta^+_{F_n.\boldsymbol{SE}}$
- $h^+_{F_n.\boldsymbol{SE}} = -h^-_{F_n\boldsymbol{E}}$
- $-1 = h^-_{F_n.\boldsymbol{SE}}$ est la plus courte normalisation de $F_n.\boldsymbol{SE}$
- $F_n.\boldsymbol{SE} = (1, -b, -n)$ n'est pas réduite classique puisque $b > 0$,
- $F_n.\boldsymbol{SET}(-1) = (1, -b-2, *)$ n'est pas réduite classique car $-b-2 < 0$
- $\mu_{F_n.\boldsymbol{SE}} = h^+_{F_n.\boldsymbol{SE}}$.

Notre algorithme ***RéductionRéelle*GL2** ayant en entrée F_n retourne la forme réduite primaire $F_n.\boldsymbol{ET}(1)$ et la matrice $(0, 1; 1, 1)$.
L'algorithme classique ***RéductionGaussRéelle*** ayant en entrée la forme F_n retourne la forme réduite classique $F_n.\boldsymbol{SET}(-h^-_{F_n.\boldsymbol{E}})$ et la matrice $(0, 1; -1, h^-_{F_n.\boldsymbol{E}})$.
On finit de prouver la proposition pour la famille de forme F_n en remarquant que la distance entre les racines de $F_n.\boldsymbol{E} = (1, b, -n)$ est $\sqrt{\Delta_{F_n}}$, et comme $|h^+_{F_n.\boldsymbol{E}} - h^-_{F_n.\boldsymbol{E}}| > |\zeta^+_{F_n.\boldsymbol{E}} - \zeta^-_{F_n.\boldsymbol{E}}| - 2$ il vient $\|(0, 1; -1, h^-_{F_n.\boldsymbol{E}})\| = |h^-_{F_n.\boldsymbol{E}}| > \sqrt{\Delta_{F_n}} - 3$.

□

4.4 Preuve du théorème *Borne réelle*

Pour prouver le théorème 4.2 nous commençons par étudier les cas de terminaisons de l'algorithme ***RéductionRéelle*GL2**. Ces cas de terminaisons sont caractérisés par la présence d'entiers entre les racines de $f.\boldsymbol{E}$. Dans l'étude de ces cas le choix de la plus courte normalisation est d'un très grande importance.

Dans un second temps nous étudierons le cas général et enfin la complexité de ***RéductionRéelle*GL2**.

4.4.1 Les deux cas de terminaison

Nous commençons par les deux cas de terminaisons de l'algorithme en un seul pas de réduction.

Nous utilisons les propriétés des représentations affines des formes réelles. Les représentations affines des formes réelles sont des fonctions convexes ou concaves, suivant le signe du premier coefficient de la forme. Si le premier coefficient de la forme est positif, la représentation affine de la forme est une fonction convexe : tout arc de la représentation affine de la forme réelle est situé au-dessous de sa corde. Sinon la représentation affine de la forme réelle est une fonction concave (en d'autres mots son opposée est convexe) : tout arc est au-dessus de sa corde.

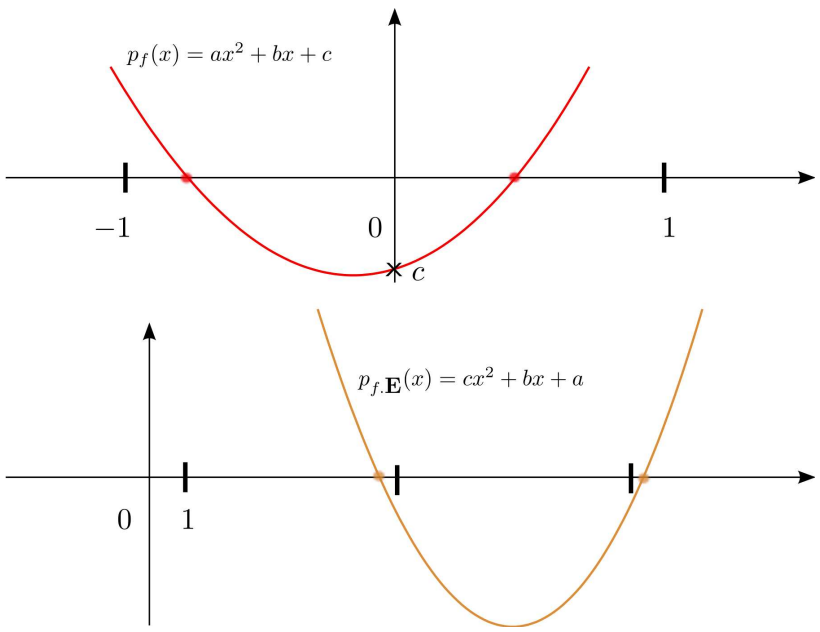


FIG. 4.3 – *Représentation affine des cas de terminaison de l'algorithme 5*

Cette figure illustre les deux cas de terminaison de l'algorithme ***RéductionRéelleGL2****. Pour une forme f normalisée primaire donnée en entrée de l'algorithme, l'algorithme finit en un pas lorsque : soit f contient exactement un entier entre ces racines, soit les racines de f sont > 0 et $f.E$ contient au moins deux entiers entre ces racines.*

Le premier cas arrive lorsque la forme f normalisée contient exactement un entier entre ses racines. On fait remarquer que c'est le seul cas où $h_f^- = h_f^+$, c'est-à-dire que toutes les notions de normalisation (classique, primaire, secondaire, la plus courte) coïncident.

Lemme 4.3 *Soit $f = (a, b, c)$ une forme réelle telle que $-1 < \zeta_f^- < 0 < \zeta_f^+ < 1$, et $h = \boldsymbol{h}(f.\boldsymbol{E})$. La forme $f_r = f.\boldsymbol{ET}(h) = (a_r, b_r, c_r)$ est réduite large, et ses coefficients satisfont $a_r = c$, $|c_r| \leqslant |a|$, et $h^2|a_r| \leqslant |a|$.*

Démonstration. La matrice de réduction de la forme f à la forme f_r est $\boldsymbol{ET}(h) = (0, 1; 1, h)$. On considère la parabole $p(x) = cx^2 + bx + a$ qui est la représentation affine de $g = f.\boldsymbol{E}$. Alors nous avons

$$h = \boldsymbol{h}(g), \text{ et } \zeta_g^- < h_g^- \leqslant -1 < 1 \leqslant h_g^+ < \zeta_g^+,\ c_r = p(\boldsymbol{h}(g)) \text{ et } p(0) = a.$$

Par définition de h nous avons deux cas :
– si $-b/2c > 0$ alors nous avons $h = h_g^- < 0 < -b/2c$,
– sinon nous avons $-b/2c < 0 < h = h_g^+$.

Dans les deux cas nous vérifions graphiquement que $|c_r| = |p(h)| < |p(0)| = |a|$ (voir Figure 4.4). Une inégalité de convexité sur p entre $[0, h]$ et $[-b/2c, -b/2c+h]$ ($|p(0)-p(h)| \geqslant |p(-b/2c) - p(-b/2c+h)|$) montre que $|a - c_r| \geqslant |c|h^2$. Puisque a et c_r sont de même signe et que $|a|$ est le plus grand, alors $|a| \geqslant |a_r|h^2$. □

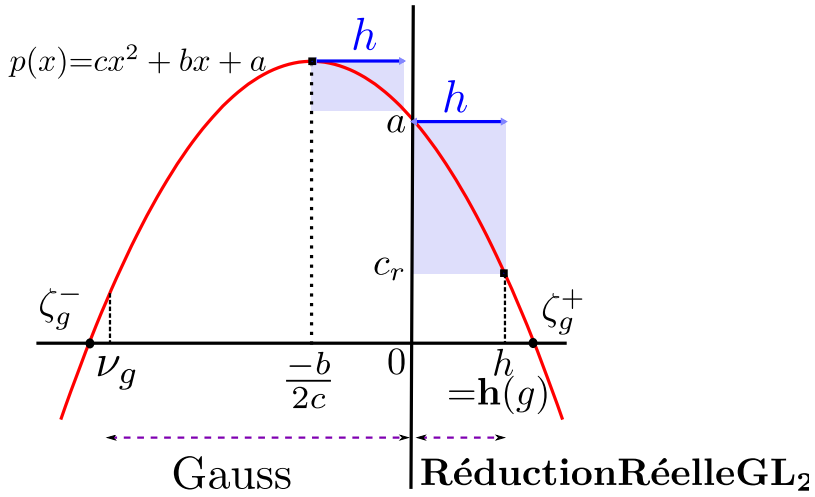


FIG. 4.4 – *Illustration du lemme 4.3*

Cette figure illustre l'inégalité de convexité du lemme 4.3. La plus courte normalisation choisie par ***RéductionRéelleGL2*** *est* $\mathbf{h}(g) = \lfloor \zeta_g^+ \rfloor$*, ce qui est* $O(\sqrt{\Delta})$ *plus petit que la normalisation classique* $\nu(g) = \lceil \zeta_g^- \rceil$ *de l'algorithme de Gauss. Il est clair que* $|c_r|$ *est dans l'intervalle* $[0, |a|]$*. En comparant la hauteur des deux rectangles sur le même convexe et la même branche décroissante de la parabole, nous avons* $|ch^2| \leqslant |a|$*.*

Le théorème 4.2 dans ce cas de terminaison s'applique sur la matrice de réduction $\mathcal{M} = (0, 1; 1, h)$. Par le lemme 4.3, sa norme satisfait $\|\mathcal{M}\| = |h| \leqslant \sqrt{|a|/|a_r|}$. Puisque $f = f_r.\mathcal{M}^{-1}$, son premier coefficient est $a = a_r h^2 - b_r h + c_r$, donc $b_r h = -a + c_r + a_r h^2$ et

$$\begin{aligned}\Delta \cdot h^2 &= (b_r h)^2 - 4a_r c_r h^2 \\ &= a^2 + c_r^2 + a_r^2 h^4 - 2ac_r - 2aa_r h^2 - 2a_r c_r h^2 \\ &\leqslant (|a| + |c_r| + |a_r|h^2)^2 \leqslant 9|a|^2,\end{aligned}$$

ce qui prouve le second point du théorème 4.2.

Le second cas où l'algorithme termine avec une seule itération, se produit lorsque la forme normalisée f a au moins deux entiers entres les racines de $f.\boldsymbol{E}$ (nous avons donc $h^-_{f.\boldsymbol{E}} < h^+_{f.\boldsymbol{E}}$).

Lemme 4.4 *Soit $f = (a, b, c)$ une forme réelle qui satisfait $0 < \zeta_f^- < \zeta_f^+ < 1$, et telle que $h^-_{f.\boldsymbol{E}} < h^+_{f.\boldsymbol{E}}$. Si $h = \boldsymbol{h}(f.\boldsymbol{E})$, alors $f_r = f.\boldsymbol{E}\boldsymbol{T}(h) = (a_r, b_r, c_r)$ est réduite secondaire, et ses coefficients satisfont $a_r = c$, $|c_r| \leqslant |a|$, et $h^2|a_r| \leqslant 4|a|$.*

Démonstration. La preuve de ce lemme est aussi basée sur des inégalités de convexités. Soit $g = f.\boldsymbol{E}$, de représentation affine $p(x) = cx^2 + bx + a$. On peut noter que $h = \lceil \zeta_g^- \rceil \geqslant 2$. Encore, nous avons

$$p(0) = a,\ p(h) = c_r.$$

Il suit de la définition que f_r est réduite secondaire.
La matrice de réduction est $\mathcal{M} = (0, 1; 1, h)$, ce qui prouve que $a_r = c$. Appliquer une inégalité de convexité (voir Figure 4.5) sur p dans les deux intervalles $[0; h-1]$ et $\left[-\frac{b}{2c} - (h-1); -\frac{b}{2c}\right]$ de même longueur donne $|a_r|(h-1)^2 \leqslant |a - p(h-1)| \leqslant |a|$ (car $p(-\frac{b}{2c} - (h-1)) - p(-\frac{b}{2c}) = a_r(h-1)^2$), et donc $|a_r|h^2 \leqslant 4|a_r|(h-1)^2 \leqslant 4|a|$.
Finalement, une autre inégalité de convexité centrée sur $\zeta_g^- \in [0; h]$ donne

$$\frac{\text{signe}(a) \cdot (p(0) - p(\zeta_g^-))}{0 - \zeta_g^-} \leqslant \frac{\text{signe}(a) \cdot (p(h) - p(\zeta_g^-))}{h - \zeta_g^-},$$

$$\text{donc } |a| = \text{signe}(a) \cdot p(0) \geqslant \frac{\zeta_g^-}{h - \zeta_g^-}(-\text{signe}(a) \cdot p(h)) = \frac{\zeta_g^-}{h - \zeta_g^-}|p(h)| \geqslant |c_r|. \qquad \square$$

Une fois encore, théorème 4.2 est vérifié dans ce cas de terminaison, mais cette fois, $\|\mathcal{M}\| = h \leqslant 2\sqrt{|a|/|a_r|}$ et $\Delta h^2 \leqslant (|a| + |c_r| + |a_r|h^2)^2 \leqslant (6|a|)^2$.

Ces deux cas de terminaison sont les seuls possibles puisque dans la section suivante nous montrons que l'Algorithme 5 obtient forcément, après un nombre fini d'itérations, une forme normalisée secondaire ayant au moins un entier entre ses racines, et de plus grande racine positive.

4.4.2 Le cas général

Nous allons maintenant prouver le cas général du théorème 4.2.

Nous appelons $f_i = (a_i, b_i, c_i)$ les valeurs successives de f au début de la boucle "tantque" de l'Algorithme 5, et $h_i = \boldsymbol{h}(f_i.\boldsymbol{E})$.

Nous supposons que la forme normalisée primaire f_0 n'a pas d'entier entre ses racines (sinon elle serait soit déjà réduite soit nous serions comme dans le cas du lemme 4.3). Donc $0 < \zeta_{f_0}^- < \zeta_{f_0}^+ < 1$.

Dès qu'il y a au moins un entier entre les racines de $f_i.\boldsymbol{E}$, alors nous posons $m = i + 1$. L'algorithme est alors dans un des deux cas de terminaisons.

Pour $i \in [0, ..., m-1]$ la plus courte normalisation h_i et aussi la normalisation primaire car il n'y a pas d'entier entre les racines de $f_i.\boldsymbol{E}$, on a $h_i = h^+_{f_i.\boldsymbol{E}} < h^-_{f_i.\boldsymbol{E}}$. Donc f_i est

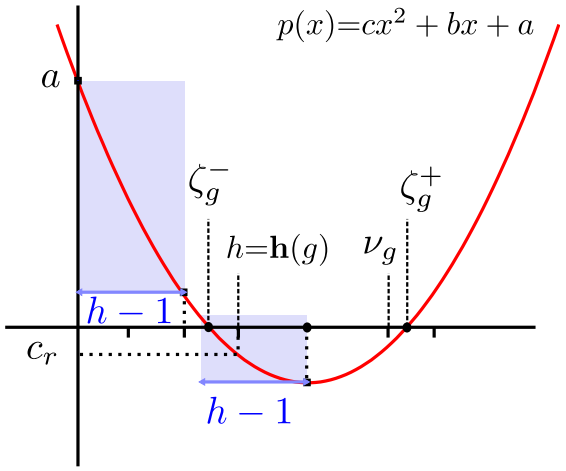


FIG. 4.5 – *Illustration du lemme 4.4*

Cette figure est l'analogue pour le lemme 4.4. Dans ce cas, la plus courte normalisation choisie par ***RéductionRéelleGL2*** *est* $\boldsymbol{h}(g) = \lceil \zeta_g^- \rceil$*, qui peut être* $O(\sqrt{\Delta})$ *fois plus petite que la normalisation classique dans l'algorithme de Gauss qui est* $\nu(g) = \lfloor \zeta_g^+ \rfloor$*. L'inégalité sur les pentes de* p *avant et après* ζ_g^- *nous donne* $|c_r| \leqslant |a|$*. En comparant la hauteur des deux rectangles sur le même convexe et la même branche décroissante de la parabole, nous avons* $|c(h-1)^2| \leqslant |a|$*.*

normalisée primaire, ses racines vérifient $0 < \zeta_{f_i}^- < \zeta_{f_i}^+ < 1$, et $h_i \geqslant 1$ (car $\boldsymbol{E}$ inverse les racines).

Puisque l'entier m est le plus petit indice tel que $f_{m-1}.\boldsymbol{E}$ contient au moins un entier entre ses racines, et que f_{m-1} vérifie $0 < \zeta_{f_{m-1}}^- < \zeta_{f_{m-1}}^+ < 1$, alors la plus courte normalisation $h_{m-1} = h^-_{f_{m-1}.\boldsymbol{E}} \leqslant h^+_{f_{m-1}\boldsymbol{E}}$ vérifie $h_{m-1} \geqslant 2$ (car h_{m-1} est le plus petit entier entre les racines de $f_{m-1}.\boldsymbol{E}$). La définition de l'entier m implique également que la forme $f_m = f_{m-1}.\boldsymbol{ET}(h_{m-1})$ est normalisée secondaire.

Montrons la terminaison de l'algorithme.

La distance entre les racines augmente strictement :

$$\left|\zeta_{f_i}^+ - \zeta_{f_i}^-\right| = \left|\zeta^+_{f_{i-1}.\boldsymbol{E}} - \zeta^-_{f_{i-1}.\boldsymbol{E}}\right| = \left|\zeta^+_{f_{i-1}} \times \zeta^-_{f_{i-1}}\right|^{-1} \cdot \left|\zeta^+_{f_{i-1}} - \zeta^-_{f_{i-1}}\right| > \left|\zeta^+_{f_{i-1}} - \zeta^-_{f_{i-1}}\right|.$$

Un tel processus ne peut être infini, sinon la suite des entiers qui sont les premiers coefficients $|a_i| = \sqrt{\Delta}/\left|\zeta_{f_i}^+ - \zeta_{f_i}^-\right|$ serait strictement décroissante.

Cela prouve la terminaison de l'algorithme.

Pour pouvoir conclure la preuve du Théorème 4.2, nous allons démontrer le lemme suivant :

Lemme 4.5 *Soient $f = (a, b, c)$ et $g = (a_g, b_g, c_g)$ deux formes réelles et $\mathcal{M} = (\alpha, \beta; \gamma, \delta) \in GL_2$ telle que $f.\mathcal{M} = g$. Si toutes les racines de g sont positives et $\gamma > 0$ et $\delta \geqslant 1$ alors $|a_g|\delta^2 \leqslant |a|$.*

Démonstration. Si $\gamma = 0$, alors $\mathcal{M}$ est triangulaire, donc $|\alpha| = |\delta| = 1$ et $|a_g| = |a|$.
Maintenant si $\gamma > 0$. Soit ζ_g une racine de g, alors $\zeta_f = \dfrac{\alpha\zeta_g + \beta}{\gamma\zeta_g + \delta}$ est une racine de f.
Nous avons

$$|\alpha/\gamma - \zeta_f| = 1/|\gamma^2\zeta_g + \gamma\delta| < 1/\gamma\delta$$

à l'aide des conditions de positivités.
Puisque cela arrive pour les deux racines de f, en utilisant la forme factorisée de f (propriété 1.1) il vient :

$$|a_g| = |f(\alpha, \gamma)| < \gamma^2|a| \left|\alpha/\gamma - \zeta_f^-\right| \left|\alpha/\gamma - \zeta_f^+\right| < |a|/\delta^2.$$

□

Continuons la preuve du Théorème 4.2 en appliquant ce lemme à la boucle principale de l'algorithme ***RéductionRéelle*GL2**.

Pour chaque $i \in [1; m]$, la matrice de réduction de f_0 à f_i est

$$\mathcal{M}_i = \begin{pmatrix} 0 & 1 \\ 1 & h_0 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & h_1 \end{pmatrix} \cdots \begin{pmatrix} 0 & 1 \\ 1 & h_{i-1} \end{pmatrix} = \begin{pmatrix} \alpha_i & \beta_i \\ \gamma_i & \delta_i \end{pmatrix}. \tag{4.3}$$

Les coefficients sont tous positifs, et ils satisfont la récurrence pour $i \geqslant 2$:

$$\begin{array}{lll} \gamma_{i+1} = \delta_i = h_{i-1}\delta_{i-1} + \delta_{i-2} & \text{et} & (\delta_0, \delta_1) = (1, h_0) \\ \alpha_{i+1} = \beta_i = h_{i-1}\beta_{i-1} + \beta_{i-2} & \text{et} & (\beta_0, \beta_1) = (0, 1). \end{array}$$

Puisque tous les $(h_j)_{j=0..i}$ sont plus grands que 1 (voir plus haut), il suit que

$$\alpha_i \leqslant \min(\beta_i, \gamma_i) \leqslant \max(\beta_i, \gamma_i) \leqslant \delta_i \text{ et } \|\mathcal{M}_i\| = \delta_i.$$

En appliquant le lemme 4.5 sur f_0 et f_{m-1} on a

$$\|\mathcal{M}_{m-1}\|^2 \leqslant |a_0|/|a_{m-1}|.$$

Le lemme 4.5 peut être appliqué à $f_m.T(-1)$, qui a ses racines positives (puisque f_m est réduite secondaire) et le même premier coefficient a_m que f_m. La matrice de transformation $\mathcal{M}_m T(-1) = \mathcal{M}_{m-1}(0, 1; 1, h_{m-1} - 1)$ satisfait encore les conditions du lemme 4.5 car $h_{m-1} \geqslant 2$. Nous obtenons $\|\mathcal{M}_m T(-1)\|^2 \leqslant |a_0|/|a_m|$, et au final

$$\|\mathcal{M}_m\|^2 \leqslant 4|a_0|/|a_m|$$

après avoir retiré la translation $T(-1)$ (on applique $T(1)$ il vient $\|\mathcal{M}_m\|^2 = \|\mathcal{M}_m T(-1)T(1)\|^2 \leqslant \|\mathcal{M}_m T(-1)\|^2 \cdot (\max(|1|+|1|, |0|+|1|)^2)$.

Pour finir de prouver les parties 1) et 2) du Théorème 4.2 il faut que l'on développe plus en détails les caractéristiques de la forme f_m, car la majoration actuelle de $\|\mathcal{M}_m\|$ n'est pas satisfaisante.

A ce stade nous avons :
– $\forall i \in [0, ..., m-1]$: f_i est normalisée primaire, $0 < \zeta^-_{f_i} < \zeta^+_{f_i} < 1$, $h_i \geqslant 1$,
– $f_m = f_{m-1}.\boldsymbol{ET}(h_{m-1})$ est normalisée secondaire, $-1 < \zeta^-_{f_{m-1}} < 0 < \zeta^+_{f_{m-1}}$, $h_{m-1} \geqslant 2$.

Premier cas

Si la plus grande racine de $f_m = f_{m-1}.\boldsymbol{ET}(h_{m-1})$ est strictement plus grande que 1 (il y avait donc au moins deux entiers entre les racines de $f_{m-1}.\boldsymbol{E}$), alors $r = m$, f_r est réduite secondaire, et la matrice de réduction est $\mathcal{M}_m = (\alpha_m, \beta_m; \gamma_m, \delta_m)$.

On a déjà $\|\mathcal{M}_m\|^2 \leqslant 4|a_0|/|a_r|$ ce qui s'écrit aussi $|a_r\delta_m|^2 \leqslant 4|a_0|$, et on a aussi $|a_{m-1}\delta_{m-1}|^2 \leqslant |a_0|$.

De $f_0 = f_r.\mathcal{M}^{-1}$, on voit que $a_0 = a_r\delta_m^2 - b_r\delta_m\gamma_m + c_r\gamma_m^2$, il vient $(-b_r\delta_m\gamma_m)^2 = (a_0 - a_r\delta_m^2 - c_r\gamma_m^2)^2$ et donc

$$\begin{aligned}\Delta\delta_m^2\gamma_m^2 &= (b_r\delta_m\gamma_m)^2 - 4a_rc_r\delta_m^2\gamma_m^2\\ &= (a_0^2 - 2a_0a_r\delta_m^2 - 2a_0c_r\gamma_m^2 + 2a_rc_r\delta_m^2\gamma_m^2 + a_r^2\delta_m^4 + c_r^2\gamma_m^4) - 4a_rc_r\delta_m^2\gamma_m^2\\ &= a_0^2 + a_r^2\delta_m^4 + c_r^2\gamma_m^4 - 2a_0a_r\delta_m^2 - 2c_ra_0\gamma_m^2 - 2a_rc_r\delta_m^2\gamma_m^2\\ &\leqslant a_0^2 + a_r^2\delta_m^4 + c_r^2\gamma_m^4 + 2|a_0a_r|\delta_m^2 + 2|a_0c_r|\gamma_m^2 + 2|a_rc_r|\delta_m^2\gamma_m^2 = (|a_0| + |a_r\delta_m^2| + |c_r\gamma_m^2|)^2.\end{aligned}$$

Puisque par construction $\gamma_m^2 = \delta_{m-1}^2 = \|\mathcal{M}_{m-1}\|^2$ et que le lemme 4.4 appliqué à f_r et f_{m-1} donne $|c_r| \leqslant |a_{m-1}|$ (par définition de l'entier m on a $0 < \zeta^-_{f_{m-1}} < \zeta^+_{f_{m-1}} < 1$ et $1 < \zeta^-_{f_{m-1}.\boldsymbol{E}} < h^-_{f_{m-1}.\boldsymbol{E}} < h^+_{f_{m-1}.\boldsymbol{E}} < \zeta^+_{f_{m-1}.\boldsymbol{E}}$ puisque $f_{m-1}.\boldsymbol{E}$ a au moins deux entiers entres ses racines), on trouve :

$$\begin{aligned}\Delta\delta_m^2\gamma_m^2 &\leqslant (|a_0| + |a_r\delta_m^2| + |c_r\gamma_m^2|)^2\\ &\leqslant (|a_0| + |a_r\delta_m^2| + |a_{m-1}\delta_{m-1}^2|)^2\\ &\leqslant (|a_0| + 4\cdot|a_0| + |a_0|)^2 = (6\cdot|a_0|)^2.\end{aligned}$$

Deuxième cas

Si la plus grande racine de f_m est strictement plus petite que 1, alors par le lemme 4.3, f_{m+1} est réduite.

La matrice de réduction est $\mathcal{M} = \begin{pmatrix} \alpha_r & \beta_r \\ \gamma_r & \delta_r \end{pmatrix} = \mathcal{M}_m \cdot \begin{pmatrix} 0 & 1 \\ 1 & h_m \end{pmatrix}$, et $r = m + 1$.

Comme précédemment on a déjà $\|\mathcal{M}_m\|^2 \leqslant 4|a_0|/|a_m|$ ce qui s'écrit aussi $|a_m\delta_m|^2 \leqslant 4|a_0|$.

Le lemme 4.3 appliqué à f_r et f_m donne $|a_m| \geqslant h_m^2|a_r|$.

Donc

$$\begin{aligned} \|\mathcal{M}\|^2 &\leqslant \|\mathcal{M}_m\|^2(1+|h_m|)^2 \\ &\leqslant 4|a_0|/|a_m| \cdot 4h_m^2 \qquad (\forall x \in \mathbb{R} \text{ tel que } |x| \geqslant 1 \text{ on a } (1+|x|)^2 \leqslant 4x^2) \\ &\leqslant 4|a_0|/|a_m| \cdot 4|a_m|/|a_r| \\ &\leqslant 16|a_0|/|a_r|. \end{aligned}$$

De $f_0 = f_r.\mathcal{M}^{-1}$, on voit que $a_0 = a_r\delta_r^2 - b_r\delta_r\gamma_r + c_r\gamma_r^2$, donc

$$\begin{aligned} \Delta\delta_r^2\gamma_r^2 &= (b_r\delta_r\gamma_r)^2 - 4a_rc_r\delta_r^2\gamma_r^2 \\ &= a_0^2 + a_r^2\delta_r^4 + c_r^2\gamma_r^4 - 2a_0a_r\delta_r^2 - 2c_ra_0\gamma^2 - 2a_rc_r\delta_r^2\gamma_r^2 \\ &\leqslant (|a_0| + |a_r\delta_r^2| + |c_r\gamma_r^2|)^2. \end{aligned}$$

Nous avons

$$\begin{aligned} \Delta\delta_r^2\gamma_r^2 &\leqslant (|a_0| + |a_r\delta_r^2| + |c_r\gamma_r^2|)^2 \\ &\leqslant (|a_0| + |a_r\delta_r^2| + |a_m\delta_m^2|)^2 \\ &\leqslant (|a_0| + 16|a_0| + 4|a_0|)^2 \\ &\leqslant (21|a_0|)^2, \end{aligned}$$

car $|c_r| \leqslant |a_m|$ par le lemme 4.3 appliqué à f_r et f_m, et par construction on a $\gamma_r = \delta_m$.

Cela conclut la preuve des parties 1) et 2) du Théorème 4.2. La partie sur la complexité du Théorème 4.2 est prouvée dans la sous-section qui suit.

4.4.3 Complexité

Nous prouvons le nombre d'itérations effectué par ***RéductionRéelle*GL2**.

Deux étapes avant la fin, à l'itération $r-2$ de ***RéductionRéelle*GL2**, nous savons que la forme $f_{r-2} = (a_{r-2}, b_{r-2}, c_{r-2})$ vérifie $\sqrt{\Delta} < |a_{r-2}|$, car la distance entre les racines de f_{m-1} et f_{m-2} est plus petite que 1 (r peut prendre deux valeurs : m ou $m+1$).

On rappelle que l'entier m est le plus petit indice, tel que $f_{m-1}.\boldsymbol{E}$ contient au moins un entier entre ses racines.

Pour chaque $i \in [1;m]$ on montre par récurrence et comparaison avec la suite de Fibonacci F_i que la norme de la matrice (4.3) (matrice $\mathcal{M}_{\ i}$ de réduction de f_0 à f_i) vérifie

$$\|\mathcal{M}_{\ i}\| = \delta_i \geqslant \mathrm{F}_i \geqslant \omega^{i-2} \ .$$

– La suite de nombres de Fibonacci F_i est définie par la relation de récurrence $\mathrm{F}_i = \mathrm{F}_{i-1} + \mathrm{F}_{i-2}$, avec des valeurs initiales $\mathrm{F}_0 = 0$ et $\mathrm{F}_1 = 1$.
Nous avons $\delta_2 = h_1\delta_1 + \delta_0 \geqslant h_0 + 1 \geqslant 2 \geqslant \mathrm{F}_2 = 1$. Et par l'hypothèse de récurrence $\delta_i \geqslant \mathrm{F}_i$ il vient $\delta_{i+1} = h_i\delta_i + \delta_{i-1} \geqslant \mathrm{F}_i + \mathrm{F}_{i-1} = \mathrm{F}_{i+1}$.
– Le i$^{\mathrm{e}}$ nombre de la suite des nombres de Fibonacci F_i est plus grand que ω^{i-2} : $\mathrm{F}_2 = 1 \geqslant 1$, et par l'hypothèse de récurrence $\mathrm{F}_i \geqslant \omega^{i-2}$ il vient $\mathrm{F}_{i+1} = \mathrm{F}_i + \mathrm{F}_{i-1} \geqslant \omega^{i-2} + \omega^{i-3} = \omega^{i-3}(1+\omega) = \omega^{i-3}\omega^2 = \omega^{i-1}$ puisque ω est solution de l'équation $x^2 - x - 1$.

Ce résultat combiné au lemme 4.5 nous donne $\omega^{r-4} \leqslant \|\mathcal{M}_{\ r-2}\| \leqslant \sqrt{|a_0/a_r - 2|} \leqslant \sqrt{|a_0/\sqrt{\Delta}|}$. Il suit que $r-4$ est majoré par $\left(\dfrac{\log\left(|a|/\sqrt{\Delta}\right)}{2\log(\omega)}\right)$ étapes, avec $\omega = \frac{1+\sqrt{5}}{2}$.

Le pire cas de complexité de l'algorithme ***RéductionRéelle*GL2** est atteint quand toutes les normalisations jusqu'à l'indice $r-2$ sont par $h = 1$. Expérimentalement nous avons vérifié que c'était le cas sur la famille $g.(\boldsymbol{T}(-1)\boldsymbol{E})^n$ avec g réduite et un naturel n qui augmente.

4.5 Nouvel algorithme de parcours d'un cycle

En utilisant les notions de primaire et secondaire, nous obtenons l'algorithme ***NièmeVoisineDroite*** qui pour une forme réelle réduite f et un indice $n \in \mathbb{N}$ donnés, retourne sa n-ème voisine de droite g et la matrice $\mathcal{M} \in \mathrm{SL}_2$ vérifiant $f.\mathcal{M} = g$.

Une itération successive de ***VoisineDroite***() permet de parcourir par la droite le cycle de réduites de f, la première voisine de droite de f est la forme réduite qui s'écrit ***VoisineDroite***$(f) = f.\boldsymbol{SET}(\nu_{f.\boldsymbol{SE}})$.

Un algorithme classique de parcours à droite du cycle de réduites fournit donc une matrice de type $(\boldsymbol{SET}(k_{(1\leqslant x\leqslant n)}))^n$ avec des entiers $k_{(1\leqslant x\leqslant n)}$ qui sont de signes alternés ce qui complique l'analyse de la norme de cette matrice.

L'avantage de notre algorithme est qu'il fournit une matrice de passage de f à g de type $\boldsymbol{S}^i(\boldsymbol{ET}(h_{(1\leqslant x\leqslant n)}))^n\boldsymbol{S}^j$ avec $\{i,j\} \in \{0,1\}$ et des entiers $h_{(1\leqslant x\leqslant n)} \geqslant 1$. La norme de cette matrice est la norme de $(\boldsymbol{ET}(h_{(1\leqslant x\leqslant n)}))^n = \begin{pmatrix}0 & 1\\ 1 & h_1\end{pmatrix}\begin{pmatrix}0 & 1\\ 1 & h_2\end{pmatrix}\ldots\begin{pmatrix}0 & 1\\ 1 & h_n\end{pmatrix}$, et puisque

$(\boldsymbol{ET}(h_{(1\leqslant x\leqslant n)}))^n$ est exactement du même type que la matrice 3.1 et que tous les entiers $h_{(1\leqslant x\leqslant n)}$ sont positifs on a la propriété suivante :

Propriété 4.4 *La fonction qui pour chaque i-ème voisine de la forme réelle réduite f donne la norme de la matrice retournée par* ***NièmeVoisineDroite****(f, i) est une fonction croissante.*

Algorithme 6 : ***NièmeVoisineDroite***

Entrée : $f = (a, b, c)$ une forme réelle réduite, $n \in \mathbb{N}$
Sorties : $f.\mathcal{M}$ la n-ème voisine de droite de f et $\mathcal{M} \in \mathrm{SL}_2$

1: $\mathcal{M} \leftarrow \mathcal{I}_2$
2: **pour** $i = 1$ à n **faire**
3: **si** f est réduite primaire **alors**
4: $f \leftarrow f.\boldsymbol{ET}(h^+_{f.\boldsymbol{E}})\boldsymbol{S}$ et $\mathcal{M} \leftarrow \mathcal{M}\boldsymbol{ET}(h^+_{f.\boldsymbol{E}})\boldsymbol{S}$
5: **sinon**
6: $f \leftarrow f.\boldsymbol{SET}(h^+_{f.\boldsymbol{SE}})$ et $\mathcal{M} \leftarrow \mathcal{M}\boldsymbol{SET}(h^+_{f.\boldsymbol{SE}})$
7: **finsi**
8: $i \leftarrow i + 1$
9: **fin pour**
10: **retourner** f et $\mathcal{M}$

Théorème 4.3 *L'algorithme* ***NièmeVoisineDroite*** *avec en entrée une forme réelle réduite est correct et il retourne une matrice de la forme $\boldsymbol{S}^i(\boldsymbol{ET}(h_{(1\leqslant x\leqslant n)}))^n\boldsymbol{S}^j$ avec $\{i, j\} \in \{0, 1\}$ et des entiers naturel $1 \leqslant h_{(1\leqslant x\leqslant n)} \leqslant \sqrt{\Delta_f}$.*

Démonstration. Soit $f = (a, b, c)$ une forme réelle réduite en entrée de la boucle "pour" de l'algorithme. On montre que l'algorithme est correct : la prochaine forme à l'entrée de la boucle "pour" est la voisine de droite de f, et on montre aussi que l'entier h de la translation vérifie $1 \leqslant h \leqslant \sqrt{\Delta_f}$.
Puisque f est réduite, de la proposition 4.2 on déduit que soit a et b sont positifs et f est réduite primaire, soit $a < 0$ et b est positif et f est réduite secondaire.

Dans le premier cas, $h^+_{f.\boldsymbol{E}} \geqslant 1$ (car $f.\boldsymbol{E}$ est réduite secondaire) et la forme $f.\boldsymbol{ET}(h^+_{f.\boldsymbol{E}})$ est réduite primaire mais pas réduite classique (la moyenne de ses racines est négative, et puisque signe$(ac) < 0$ (car f est réduite) cela entraîne que son deuxième coefficient est négatif), par contre la symétrique de cette forme est réduite secondaire et réduite classique. La voisine de droite (définition 3.6) g de f est $f.\boldsymbol{SE}(h^-_{f.\boldsymbol{SE}})$ (puisque $c < 0$ on a $\nu_{f.\boldsymbol{SE}} = h^-_{f.\boldsymbol{SE}}$), et comme $h^-_{f.\boldsymbol{SE}} = -h^+_{f.\boldsymbol{E}}$ (la symétrie change le signe des racines) on a bien $g = f.\boldsymbol{ET}(h^+_{f.\boldsymbol{E}})\boldsymbol{S}$.

Dans le deuxième cas, la forme $f.\boldsymbol{SE} = (c, -b, a)$ est réduite secondaire donc $h^+_{f.\boldsymbol{SE}} \geqslant 1$ et $f.\boldsymbol{SET}(h^+_{f.\boldsymbol{SE}})$ est bien la définition de la voisine de droite de f (puisque $c > 0$ on a $\nu_{f.\boldsymbol{SE}} = h^+_{f.\boldsymbol{SE}}$). La voisine de droite de f est dans ce cas réduite primaire.

La distance entre les racines de f est majorée par la racine carrée de Δ_f, et puisque f est réduite : si $a > 0$ la plus petite des racines de $f.\boldsymbol{E}$ est dans $]-1, 0[$, sinon c'est la plus petite des racines de $f.\boldsymbol{ES}$ qui est dans l'intervalle $]-1, 0[$. Dans les deux cas cela entraîne que la distance de 0 à $|\nu_{f.\boldsymbol{SE}}|$ est majorée par $\sqrt{\Delta_f}$.

Donc l'algorithme ***NièmeVoisineDroite*** est correct et à chaque boucle "pour" on translate par un entier $1 \leqslant h \leqslant \sqrt{\Delta_f}$.

Puisque la voisine de droite d'une forme réduite primaire est réduite secondaire, et que la voisine de droite d'une forme réduite secondaire est réduite primaire, on en déduit que la matrice retournée par l'algorithme ***NièmeVoisineDroite*** est de la forme $\boldsymbol{S}^i(\boldsymbol{ET}(h_{(1\leqslant x\leqslant n)}))^n\boldsymbol{S}^j$ avec $\{i, j\} \in \{0, 1\}$ et des entiers naturels $1 \leqslant h_{(1\leqslant x\leqslant n)} \leqslant \sqrt{\Delta_f}$. □

Nous faisons remarquer qu'à partir d'une forme réduite $f = (a, b, c)$ (si $a > 0$ la forme est réduite primaire, sinon elle est réduite secondaire (proposition 4.2)). Il existe une seule translation qui appliquée à la forme $f.\boldsymbol{SE}$ donne une forme réduite différente de $f.\boldsymbol{SE}$, l'unique entier de cette translation est l'entier non nul h parmi $h^+_{f.\boldsymbol{SE}}$ ou $h^-_{f.\boldsymbol{SE}}$.

De la même manière il existe une seule translation qui appliquée à la forme $f.\boldsymbol{E}$ (forcément réduite classique) donne une forme réduite large différente de $f.\boldsymbol{E}$, l'unique entier de cette translation est l'entier k parmi $h^+_{f.\boldsymbol{E}}$ ou $h^-_{f.\boldsymbol{E}}$ qui n'est pas nul. On a $k = -h$ car la symétrie change le signe des racines, de plus la forme ainsi obtenue n'est pas réduite classique car son deuxième coefficient est négatif.

Proposition 4.4 *Soient une forme réelle réduite f, et un indice $n \in \mathbb{N}$. La norme de la matrice retournée par* ***NièmeVoisineDroite***(f, n) *est majorée par* $(\sqrt{\Delta_f} + 1)^n$ *et minorée par* ω^{n-2}.

Démonstration. Nous appelons $f_i = (a_i, b_i, c_i)$ les valeurs successives de f au début de la boucle "pour" de l'algorithme ***NièmeVoisineDroite***, et h_i est l'entier de la translation. Pour chaque i-itération de la boucle "pour" la matrice de passage de f à sa i-ème voisine de droite est $\boldsymbol{S}^k\mathcal{M}_{\ i}\boldsymbol{S}^l$ (théorème 4.3) avec

$$\mathcal{M}_{\ i} = \begin{pmatrix}0 & 1\\ 1 & h_0\end{pmatrix}\begin{pmatrix}0 & 1\\ 1 & h_1\end{pmatrix}\cdots\begin{pmatrix}0 & 1\\ 1 & h_{i-1}\end{pmatrix} = \begin{pmatrix}\alpha_i & \beta_i\\ \gamma_i & \delta_i\end{pmatrix}. \tag{4.4}$$

Les coefficients de $\mathcal{M}_i$ sont tous positifs, et ils satisfont la récurrence pour $i \geqslant 2$:

$$\begin{array}{lll}\gamma_{i+1} = \delta_i = h_{i-1}\delta_{i-1} + \delta_{i-2} & \text{et} & (\delta_0, \delta_1) = (1, h_1)\\ \alpha_{i+1} = \beta_i = h_{i-1}\beta_{i-1} + \beta_{i-2} & \text{et} & (\beta_0, \beta_1) = (0, 1).\end{array}$$

Puisque tous les $(h_j)_{j=0..i}$ sont plus grands que 1 (théorème 4.3), il suit que

$$\alpha_i \leqslant \min(\beta_i, \gamma_i) \leqslant \max(\beta_i, \gamma_i) \leqslant \delta_i \text{ et } \|\mathcal{M}_{\ i}\| = \delta_i \geqslant \omega^{i-2}$$

par récurrence et comparaison avec la suite de Fibonacci.
Et enfin puisque pour $i \geqslant 2$ on a $\delta_i = h_{i-1}\delta_{i-1} + \delta_{i-2}$, en utilisant le théorème 4.3 on montre la récurrence

$$\|\mathcal{M}_i\| = \delta_i < (\sqrt{\Delta_f} + 1)^i.$$

On a $\delta_0 = 1 < (1+\sqrt{\Delta_f})^0$, $\delta_1 = h_0 < (1+\sqrt{\Delta_f})^1$, $\delta_2 = 1+h_0h_1 \leqslant 1+\Delta_f < (1+\sqrt{\Delta_f})^2$ et en supposant $\delta_{i-1} < (1+\sqrt{\Delta_f})^{i-1}$ et $\delta_{i-2} < (1+\sqrt{\Delta_f})^{i-2}$ il vient

$$\delta_i = h_{i-1}\delta_{i-1} + \delta_{i-2} < \sqrt{\Delta_f}(1+\sqrt{\Delta_f})^{i-1} + (1+\sqrt{\Delta_f})^{i-2} = \frac{(\sqrt{\Delta_f}+1)^i(\Delta_f+\sqrt{\Delta_f}+1)}{(\sqrt{\Delta_f}+1)^2}$$

$$\delta_i < (\sqrt{\Delta_f}+1)^i.$$

□

4.5.1 Énumérer le cycle de réduites

Pour énumérer le cycle de réduites d'une forme réelle réduite f il suffit d'énumérer successivement toutes les voisines de droite de f en utilisant l'algorithme 6 avec l'indice 1, jusqu'à ce que une de ces voisines soit égale à f.

Algorithme 7 : ***Cycle***

Entrée : $f = (a, b, c)$ une forme réelle réduite
Sorties : le cycle de f et l'automorphisme fondamental $\mathcal{U} \in \mathrm{SL}_2$ de f

1: $C \leftarrow [f]$
2: $g, \mathcal{U} \leftarrow$ ***NièmeVoisineDroite***$(f, 1)$
3: **tantque** $f \neq g$ **faire**
4: $\quad C \leftarrow C + [g]$
5: $\quad \mathcal{U} \leftarrow \mathcal{U}\mathcal{M}$
6: $\quad g, \mathcal{M} \leftarrow$ ***NièmeVoisineDroite***$(g, 1)$
7: **fin tantque**
8: **retourner** C et $\mathcal{U}$

Il est évident que le théorème 4.3 et la proposition 4.4 peuvent être appliqués à l'algorithme ***Cycle***.

Exemple 4.1 *L'algorithme* ***Cycle*** *appliqué à la forme réelle réduite* $f = (-2, 9, 3)$*, fournit le cycle de* f *(figure 4.6), et l'automorphisme fondamental de* f *:*

$$(4, 5; -13, -16) = \boldsymbol{S}(0, 1; 1, 3)(0, 1; 1, 4)(0, 1; 1, 1)(0, 1; 1, 4)\boldsymbol{S}.$$

En comparaison l'algorithme 6.1 page 131 de [8] ne fournit ni le cycle d'une forme réelle réduite $f = (a, b, c)$ ni l'automorphisme fondamental de f : même si on peut extraire ces deux informations de l'algorithme 6.1 de [8].

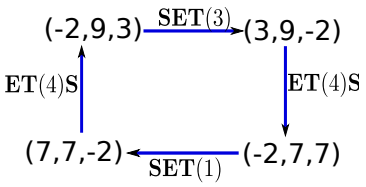


FIG. 4.6 – ***Cycle***$((-2,9,3))$

Dans [8] la fonction ***VoisineDroite***() est notée $\rho()$ alors que la forme $(-a,b,-c) = -f.\boldsymbol{S}$ est notée $\tau(f)$ et le cycle de la forme f est défini comme étant le *cycle propre* de f.

Les auteurs de [8] itèrent successivement ***VoisineDroite***$(-f.\boldsymbol{S})$ dans leur notation $\rho(\tau(f))$.

Bien sur la forme $-f.\boldsymbol{S}$ (qui est forcément réduite classique) n'est pas forcément équivalente à la forme f, mais avec une identification bien choisie, en supposant que l'automorphisme fondamental de f retourné par ***Cycle***(f) s'écrit $\boldsymbol{S}^{\ell}(\boldsymbol{ET}(h_{(1\leqslant x\leqslant n)}))^n\boldsymbol{S}^k$ avec $\{\ell,k\} \in \{0,1\}$, l'algorithme 6.1 de [8] fournit la matrice $(\boldsymbol{ET}(h_{(1\leqslant x\leqslant n)}))^n$ (les entiers $h_{(1\leqslant x\leqslant n)}$ sont égaux aux entiers $s_{(1\leqslant i\leqslant n)}$ de [8]) :

$$f.\boldsymbol{S} = (a,-b,c) \text{ est identifiée à la forme } -f.\boldsymbol{S} = (-a,b,-c).$$

Cette identification permet d'écrire $f.\boldsymbol{S} = (-a,b,-c)$ mais aussi $f.\boldsymbol{SS} = (-a,-b,-c)$ et puisque $\boldsymbol{S}^2 = \mathcal{I}_2$ il vient $f = (-a,-b,-c) = -f$.

Les auteurs de [8] peuvent alors écrire dans le cas où $a > 0$:

$$f.\boldsymbol{ET}(h) = (-a,-b,-c)\boldsymbol{ET}(h),$$

avec $h = \lfloor \zeta^+_{f.\boldsymbol{E}} \rfloor$ et leur algorithme 6.1 est une itération de

$$\boldsymbol{VoisineDroite}(-f.\boldsymbol{S}) = (-a,-b,-c)\boldsymbol{ET}(h).$$

On remarque que l'on a bien $\lfloor \zeta^+_{f.\boldsymbol{E}} \rfloor = \lfloor \zeta^+_{-f.\boldsymbol{E}} \rfloor$ puisque la représentation affine de $-f$ est la symétrie axiale (par rapport à l'axe des abscisses) de la représentation affine de f. De plus, comme la forme f est réduite de premier coefficient positif alors c est négatif et donc la forme $(-a,-b,-c)\boldsymbol{ET}(h)$ est de premier $-c$ qui est positif.

Pour finir on va montrer que ***VoisineDroite***$(-f.\boldsymbol{S}) = -$***VoisineDroite***$(f).\boldsymbol{S}$ (propriété 4.5), c'est-à-dire avec les notations de [8] que $\rho(\tau(f)) = \tau(\rho(f))$, ainsi a chaque itération dans l'algorithme 6.1 de [8] on obtient en alternance
– soit la prochaine voisine (de droite) sur le cycle de $-f.\boldsymbol{S}$
– soit la prochaine voisine (de droite) sur le cycle de f,
ce qui est illustré par les formes coloriées en rose de la figure 4.7.

Exemple 4.2 *Soit la forme réelle réduite* $f = (1,9,-6)$ *de discriminant* 105.
Le cycle de f *et le cycle de* $-f.\boldsymbol{S}$ *sont respectivement illustrés par la figure 4.7.*

Les formes f *et* $-f.\boldsymbol{S}$ *ne sont pas équivalentes. Pour calculer le cycle de* f *l'algorithme* 6.1 *page* 131 *de [8] énumère dans cet ordre les formes :*

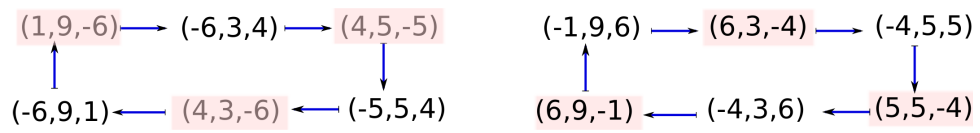


FIG. 4.7 – *Cycle de* $(1,9,-6)$ *et cycle de* $(-1,9,6)$

$$(1,9,-6)\ (6,3,-4)\ (4,5,-5)\ (5,5,-4)\ (4,3,-6)\ (6,9,-1).$$

La voisine de droite de $(-1,9,6)$ *est* $(6,3,-4)$*, la voisine de droite de* $(-6,3,4)$ *est* $(4,5,-5)$*, et ainsi de suite, jusqu'à ce que la voisine de droite de* $(-6,9,1)$ *soit* f*.*

Propriété 4.5 *Pour tout forme réelle réduite :*

$$\boldsymbol{VoisineDroite}(-f.\boldsymbol{S}) = -\boldsymbol{VoisineDroite}(f).\boldsymbol{S}.$$

Démonstration. On a $\boldsymbol{VoisineDroite}(-f.\boldsymbol{S}) = \boldsymbol{VoisineDroite}(-a,b,-c) = (-c,-b,-a).\boldsymbol{T}(\nu_{-f.\boldsymbol{E}})$ et $\nu_{-f.\boldsymbol{E}} = \nu_{f.\boldsymbol{E}} = -\nu_{f.\boldsymbol{SE}} \geqslant 1$ (car l'opposée d'une forme a les mêmes racines (seule l'orientation de la parabole change) et la symétrie change le signe des racines) donc $\boldsymbol{VoisineDroite}(-f.\boldsymbol{S}) = -f.\boldsymbol{ET}(-\nu_{f.\boldsymbol{SE}})$. Et puisque $\boldsymbol{VoisineDroite}(f) = f.\boldsymbol{ET}(-\nu_{f.\boldsymbol{SE}})\boldsymbol{S}$ l'égalité est bien vérifiée. □

4.6 Plus petite SL_2-matrice de passage

Nous voulons obtenir un critère sur la norme d'une matrice de passage de SL_2 entre deux formes équivalentes qui, s'il est vérifié, implique forcément que cette matrice est de norme minimale.

Si la matrice est une matrice de passage entre deux formes imaginaires de discriminant $\Delta < -4$, comme dans ce cas les seuls automorphismes sont $\pm\mathcal{I}_2$, alors la matrice de passage est nécessairement de norme minimale puisque cette matrice est unique au signe près (proposition 1.11).

Par contre il y a une infinité de matrices de passage de SL_2 d'une forme réelle.

4.6.1 Distance entre deux formes réelles équivalentes

On définit une notion de *distance* entre deux formes réelles équivalentes.

Définition 4.5 *La* distance *entre deux formes réelles équivalentes* $f \sim g$ *est*

$$\mathbf{dist}(f,g) = \min\{\log(\|\mathcal{M}\|),\ \mathcal{M} \in SL_2 \text{ et } f.\mathcal{M} = g\}.$$

Si f,g et h sont trois formes de $\mathfrak{F}_\Delta$, la fonction distance satisfait les trois propriétés suivantes :

1. $\mathbf{dist}(f,g) = \mathbf{dist}(g,f) \geqslant 0$
2. $\mathbf{dist}(f,g) = 0 \iff f = g$ ou $f = g.\boldsymbol{SE}$
3. $\mathbf{dist}(f,h) \leqslant \mathbf{dist}(f,g) + \mathbf{dist}(g,h)$.

Pour le voir on utilise juste les propriétés des normes induites, et le fait que seules les isométries ont une norme égale à 1.

La première et dernière propriété que satisfait notre fonction $\mathbf{dist}(.,.)$ sont deux des trois propriétés que doit satisfaire une fonction distance usuelle. Quant à sa deuxième propriété elle n'est pas usuelle car une distance nulle entre deux formes n'implique pas forcément qu'elles sont égales, mais implique qu'elles sont égales modulo l'action de $\boldsymbol{SE} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$.

On déduit le théorème important suivant :

Théorème 4.4 *Soient f et g deux formes réelles de $\mathfrak{F}_\Delta$, si $\mathcal{M} \in SL_2$ satisfait $f.\mathcal{M} = g$ et $\|\mathcal{M}\| \leqslant \sqrt{\epsilon_\Delta}$, alors* $\mathbf{dist}(f,g) = \log(\|\mathcal{M}\|)$.

Démonstration. Soit U l'automorphisme fondamental de la forme f, les valeurs propres de U sont ϵ_Δ et ϵ_Δ^{-1}.

Tout automorphisme non trivial V de f satisfait $\|V\| \geqslant \epsilon_\Delta$, car V est une puissance non nulle de U, et que son rayon spectral est une puissance positive de ϵ_Δ.

On en déduit que la matrice $\mathcal{M}$ du théorème 4.4 est nécessairement la plus petite matrice de transformation de f à g, puisque tout autre matrice $X \in \mathrm{SL}_2$ vérifiant $f.X = g$ et $\|X\| < \|\mathcal{M}\|$ produirait un automorphisme non trivial de f de norme trop petite $\|\mathcal{M}\, X^{-1}\| < \epsilon_\Delta$, ce qui est impossible. □

Un des plus grands avantages de cette distance est ce théorème 4.4 qui, en général, indique que toute matrices de transformations qui est polynomiale est nécessairement la plus petite. Cela permet de minorer efficacement la distance entre deux formes réelles (non réduites ou réduites).

Comme le montre la preuve du théorème 4.4, il est essentiel que le groupe des automorphismes de trace positive soit monogène : le théorème serait faux dans GL_2.

Dans la littérature, la notion de distance utilisée est la *distance de Shanks* [38]. Cette distance peut se modéliser à l'intérieur d'un cycle, comme étant le plus petit $k \in \mathbb{N}$ tel qu'il existe $h_1, \ldots, h_k$ tel que $\prod_{i=1}^{k} \boldsymbol{SET}(h_i)$ transforme f en g ou $g.\boldsymbol{SE}$.

Malheureusement, cette distance ne satisfait pas le théorème 4.4 : il n'existe aucun moyen efficace de vérifier qu'une distance donnée est la plus petite.

Toutes les variantes de la distance de Shanks sont basées soit sur les logarithmes des h_i soit sur des normes maximales, dans tous les cas cette distance n'est plus positive et ne respecte plus l'inégalité triangulaire.

Cela explique pourquoi nous n'utiliserons pas la distance Shanks et pourquoi nous avons introduit notre propre définition.

Conclusion du chapitre

Les algorithmes de réduction sont plus simples à étudier dans GL_2 principalement parce que nous manipulons des matrices à coefficients positifs, ce qui est plus facile à majorer.

Troisième partie

Étude des cryptosystèmes

Chapitre 5

Particularités des classes d'équivalence

5.1 Groupe de classes

Dans cette section nous décrivons comment composer deux formes de $\mathfrak{F}_\Delta$ pour obtenir une forme de $\mathfrak{F}_\Delta$. Muni de cette loi de composition interne, notée $\star$, l'ensemble des formes primitives de même discriminant $\mathfrak{F}_\Delta$ forme un groupe de classe abélien.

Cette loi de composition entre deux formes f et g de $\mathfrak{F}_\Delta$ a été introduite par Gauss dans [20], et est composée de deux étapes :

– La première étape correspond au calcul du produit des deux formes en les variables respectivement (x, y) et (z, t) : on obtient ainsi un polynôme à coefficient entier, en les monômes xz, xt, yz, yt.

– La deuxième étape correspond à un changement de variable entier

$$X = n_1xz + n_2xt + n_3yz + n_4yt$$
$$Y = m_1xz + m_2xt + m_3yz + m_4yt,$$

avec $\forall i \in \{1..4\}(n_i, m_i) \in \mathbb{Z}^2$, qui vérifie $\text{pgcd}(n_1, .., n_4, m_1, .., m_4) = 1$ et qui a pour résultat une forme $h(X, Y)\,(= f(x, y)g(z, t))$ de $\mathfrak{F}_\Delta$.

5.1.1 Identitée de Dirichlet

Donnons un exemple avec deux formes particulières de même discriminant Δ. Cet exemple a été présenté par Dirichlet dans [17]. Le produit de deux formes, s'écrivant

$$f(x, y) = ax^2 + bxy + a'ny^2 \text{ et } g(z, t) = a'z^2 + bzt + ant^2$$

pour un entier n, est la forme

$$h(X, Y) = aa'X^2 + bXY + nY^2$$

avec $X = xz - nyt$ et $Y = axt + a'yz + byt$ et l'on a $\Delta_h = \Delta$.

Définition 5.1 *L'identité de Dirichlet pour toute forme* $(a, b, a'n)$ *et* (a', b, an) *avec* $n \in \mathbb{Z}$ *s'écrit* $aa'X^2 + bXY + nY^2 = (ax^2 + bxy + a'ny^2)(a'z^2 + bzt + ant^2)$, *avec* $X = xz - nyt$ *et* $Y = axt + a'yz + byt$.

Exemple 5.1 *Le produit des deux formes* $(3, 6, 5{\cdot}2)$ *et* $(2, 6, 3{\cdot}5)$ *de* $\mathfrak{F}_{-84}$ *en les variables respectivement* (x, y) *et* (z, t), *est la forme* $(6, 6, 5)$ *en les variables* (X, Y) *de* $\mathfrak{F}_{-84}$ *(on applique l'identité de Dirichlet).*

5.1.2 Composée de deux formes de $\mathfrak{F}_\Delta$

Donnons la définition formelle de la composée de deux formes f et g de $\mathfrak{F}_\Delta$.

Définition 5.2 *La composée de deux formes* f *et* g *de* $\mathfrak{F}_\Delta$ *est une forme* $h \in \mathfrak{F}_\Delta$ *qui vérifie* $h(X, Y) = f(x, y)g(z, t)$ *pour* $X = n_1xz + n_2xt + n_3yz + n_4yt$ *et* $Y = m_1xz + m_2xt + m_3yz + m_4yt$, *avec* $\forall i \in \{1..4\}(n_i, m_i) \in \mathbb{Z}^2$ *et* $\text{pgcd}(n_1, .., n_4, m_1, ..m_4) = 1$. *On note* $h = f \star g$.

Supposons qu'une forme h, composée des deux formes f et g, existe :

$$h(X, Y) = f(x, y)g(z, t) \text{ avec } X \text{ et } Y \text{ de la définition 5.2.}$$

On peut écrire X et Y en fonction, soit de x et y, soit de z et t : pour le voir on pose

$$\mathcal{M} = \begin{pmatrix} n_1z + n_2t & n_3z + n_4t \\ m_1z + m_2t & m_3z + m_4t \end{pmatrix}, \text{ et } \mathcal{N} = \begin{pmatrix} n_1x + n_3y & n_2x + n_4y \\ m_1x + m_3y & m_2x + m_4y \end{pmatrix}.$$

Cela montre que une composée de deux formes, $h(X, Y)$, peut s'écrire

$$\text{soit } h(X, Y) = h(x, y).\mathcal{M}, \text{ soit } h(X, Y) = h(z, t).\mathcal{N}.$$

Et donc en utilisant le fait que $h(X, Y) = f(x, y)g(z, t)$, et la propriété de l'action de composition sur les discriminants, on a alors que

$$\Delta_h q^2 = \Delta_f p^2 \text{ et } \Delta_h s^2 = r^2\Delta_g$$

pour les quatre entiers $q = \det(M)$, $p = g(z, t)$, $s = \det(N)$ et $r = f(x, y)$.

Ainsi $\Delta_h = (p/q)^2\Delta_f = (r/s)^2\Delta_g$, c'est-à-dire que les trois discriminants sont égaux à un rationnel carré près.

Si on suppose que les trois discriminants sont égaux, alors on a $p^2 = q^2$ et $r^2 = s^2$, ce qui s'écrit $g(z, t)^2 = \det(\mathcal{M})^2$ et $f(x, y)^2 = \det(\mathcal{N})^2$.

Par identification et en choisissant correctement le signe de $g(z, t) = \pm\det(\mathcal{M})$ et $f(x, y) = \pm\det(\mathcal{N})$, on trouve (voir [20]) que les formes f et g doivent vérifier au signe près

$$f = (n_1m_2 - m_1n_2, (n_1m_4 - m_1n_4) - (n_2m_3 - m_2n_3), n_3m_4 - m_3n_4)$$
$$g = (n_1m_3 - m_1n_2, (n_1m_4 - m_1n_4) + (n_2m_3 - m_2n_3), n_2m_4 - m_2n_4).$$

Ce qui entraîne que h s'écrit

$$h = (m_2m_3 - m_1m_4, (m_1m_4 - n_2m_3) + (m_1n_4 - m_2n_3), n_2n_3 - n_1n_4).$$

Pour finir puisque $t = \text{pgcd}(n_1, .., n_4, m_1, ..m_4)$ divise $\text{pgcd}(h)$ et $\text{pgcd}(g)$ et $\text{pgcd}(f)$, si on suppose $\text{pgcd}(f) = \text{pgcd}(g) = 1$ alors on a forcément $t = 1$ et $\text{pgcd}(h) = 1$.

A partir de la définition 5.2, on peut donc prouver la proposition suivante :

Proposition 5.1 *La composée de deux formes f et g de $\mathfrak{F}_\Delta$ existe et c'est la forme $h \in \mathfrak{F}_\Delta$ si et seulement si $\forall i \in \{1..4\}, \exists (n_i, m_i) \in \mathbb{Z}^2$:*

$$\begin{aligned} 1 &= \text{pgcd}(n_1, .., n_4, m_1, ..m_4) \\ f &= (\delta_{12}, \delta_{14} - \delta_{23}, \delta_{34}) \\ g &= (\delta_{13}, \delta_{14} + \delta_{23}, \delta_{24}) \\ h &= (m_2m_3 - m_1m_4, (m_1m_4 - n_2m_3) + (m_1n_4 - m_2n_3), n_2n_3 - n_1n_4), \end{aligned}$$

avec δ_{ij} les déterminants $\delta_{ij} = n_im_j - m_in_j$.

Nous avons défini la composition pour des formes de $\mathfrak{F}_\Delta$, mais en fait c'est une opération sur les classes de formes de $\mathfrak{F}_\Delta$. Nous avons le théorème fondamental suivant, démontré par Gauss dans [20].

Théorème 5.1 *L'ensemble des classes de $\mathfrak{F}_\Delta$ muni de la loi interne $\star$ forme un groupe de classe abélien :*
– *l'élément neutre est : $\overline{(1, *, *)} \in \mathcal{O}_{\mathfrak{F}_\Delta}$,*
– *la classe inverse de $\overline{(a, b, c)} \in \mathcal{O}_{\mathfrak{F}_\Delta}$ est $\overline{(a, -b, c)} \in \mathcal{O}_{\mathfrak{F}_\Delta}$.*

La loi est donc associative et commutative : $\forall (f, g, h) \in \mathfrak{F}_\Delta$ on a
– $\overline{f} \star (\overline{g} \star \overline{h}) = (\overline{f} \star \overline{g}) \star \overline{h}$.
– $\overline{f} \star \overline{g} = \overline{g} \star \overline{f}$.

En se ramenant à l'identité de Dirichlet 5.1 il est aisé de vérifier les deux points du théorème. Supposons que la forme (a, b, c) appartienne à $\mathfrak{F}_\Delta$, et soit $(1, b_1, c_1)$ une forme de la classe principale de $\mathcal{O}_{\mathfrak{F}_\Delta}$.

Par l'action de $\boldsymbol{SE}$ il vient $(a, -b, c) \sim (c, b, a)$. On peut appliquer l'identité de Dirichlet aux formes (a, b, c) et (c, b, a) ce qui donne la forme $(ac, b, 1)$ de $\mathfrak{F}_\Delta$ qui est bien dans la classe principale (proposition 3.5). La classe inverse de la classe de (a, b, c) est donc bien la classe de la forme $(a, -b, c)$.

Par l'action de $\boldsymbol{T}(h')$ pour un entier h' il vient $(1, b_1, c_1) \sim (1, b_1 + 2h', *)$. Donc en prenant $h = {}^{-b_1}/_2$ modulo b ou $b/2$ (suivant la parité de b), il vient $(1, b_1, c_1) \sim (1, b, ac)$. On peut alors appliquer l'identité de Dirichlet 5.1 aux formes $(1, b, ac)$ et (a, b, c) ce qui donne la forme (a, b, c). La classe principale est donc bien l'élément neutre du groupe $(\mathcal{O}_{\mathfrak{F}_\Delta}, \star)$.

5.1.3 Algorithme de composition

Un point essentiel que l'on n'a pas encore abordé dans cette section est le calcul en pratique de la composée de deux formes de $\mathfrak{F}_\Delta$. Il n'est pas du tout évident que l'on puisse trouver en pratique les 8 entiers vérifiant le système de la proposition 5.1, ce qui nous permettrait de calculer la composée des deux formes.

Puisque la loi $\star$ est définie modulo les classes, théorème 5.1, on peut supposer que les deux formes $f = (a_f, b_f, c_f)$ et $g = (a_g, b_g, c_g)$ de $\mathfrak{F}_\Delta$ qu'on veut composer sont équivalentes à un certain type de formes, puisque cela ne change pas la classe de la composée de f et g. Finalement on va essayer de se rapprocher de l'identité de Dirichlet 5.1.

Proposition 5.2 *Pour toute $f = (a_f, b_f, c_f)$ et $g = (a_g, b_g, c_g)$ de $\mathfrak{F}_\Delta$, avec $r = \mathrm{pgcd}(a_f, a_g, {}^{(b_f+b_g)}\!/_2)$, il existe un unique entier B modulo $(2a_fa_g)/(r \cdot \mathrm{pgcd}(a_f, a_g))$ vérifiant le système :*

$$(S)\begin{cases} B & \equiv & b_f & \mod (2a_f/r) \\ B & \equiv & b_g & \mod (2a_g/r) \\ B^2 & \equiv & \Delta & \mod (4a_fa_g/r^2) \end{cases}.$$

On remarquera que si $r = 1$, cette proposition implique que les formes f et g sont respectivement équivalentes aux formes $(a_f, B, *)$ et $(a_g, B, *)$ puisque par l'action des translations le deuxième coefficient des formes est défini modulo 2 fois le premier coefficient, et que ${}^{(B^2-\Delta)}\!/_{(4a_fa_g)}$ est un entier.

Avant de démontrer cette proposition on explique pourquoi elle permet d'effectuer la composée de toute forme $f = (a_f, b_f, c_f)$ et $g = (a_g, b_g, c_g)$ de $\mathfrak{F}_\Delta$. Si l'entier B vérifiant la proposition 5.2 existe, alors le changement de variable entier :

$$X = rxz + \frac{b_g - B}{2a_g/r}xt + \frac{b_f - B}{2a_f/r}yz + \frac{\Delta + b_fb_g - B(b_f + b_g)}{4a_fa_g/r}yt$$
$$Y = \frac{a_f}{r}xt + \frac{a_g}{r}yz + \frac{b_f + b_g}{2r}zt$$

pour des variables x, y et z, t et l'entier $r = \mathrm{pgcd}(a_f, a_g, {}^{(b_f+b_g)}\!/_2)$, vérifie les conditions de la proposition 5.1. Nous en déduisons que $f(x, y)g(z, t)$ est égale à $h(X, Y) = \frac{a_fa_g}{r^2}X^2 + BXY + Cy^2$ avec $C = {}^{(B^2-\Delta)}\!/_{(4a_fa_g/r^2)}$. En d'autre termes la composée de f et g est $h = f \star g$.

Nous avons donc le théorème suivant :

Théorème 5.2 *La composée de deux formes $f = (a_f, b_f, c_f)$ et $g = (a_g, b_g, c_g)$ de $\mathfrak{F}_\Delta$, avec $r = \mathrm{pgcd}(a_f, a_g, {}^{(b_f+b_g)}\!/_2)$, B l'entier de la proposition 5.2 et l'entier $C = {}^{(B^2-\Delta)}\!/_{(4a_fa_g/r^2)}$ est la forme $\left(\frac{a_fa_g}{r^2}, B, C\right)$.*

Démontrons la proposition 5.2.

Démonstration. Les solutions de $B \equiv b_f \mod (2a_f/r)$ s'écrivent

$$B = (2a_f/r)h_f + b_f,\ \forall h_f \in \mathbb{Z}.$$

A partir de ces solutions, on peut résoudre

$$B \equiv b_g \mod (2a_g/r)$$

si et seulement si $-(a_f/r)h_f \equiv {}^{(b_f-b_g)}\!/_2 \pmod{(a_g/r)}$, et cette dernière congruence est vraie si et seulement si ${}^{(\mathrm{pgcd}(a_f,a_g))}\!/_r$ divise ${}^{(b_f-b_g)}\!/_2$.

On montre maintenant que ${}^{(\mathrm{pgcd}(a_f,a_g))}\!/_r$ divise ${}^{(b_f-b_g)}\!/_2$. Comme $\Delta_f = \Delta_g$ on a $({}^{(b_f-b_g)}\!/_2)\,({}^{(b_f+b_g)}\!/_2) = a_f c_f - a_g c_g$. On en déduit que ${}^{(\mathrm{pgcd}(a_f,a_g))}\!/_r$ divise ${}^{(b_f-b_g)}\!/_2$ puisque par définition de r on a $({}^{(b_f-b_g)}\!/_2)\,({}^{(b_f+b_g)}\!/_{2r}) = (a_f/r)c_f - (a_g/r)c_g$ et $\mathrm{pgcd}({}^{(b_f+b_g)}\!/_{2r}, {}^{(\mathrm{pgcd}(a_f,a_g))}\!/_r) = 1$.

Donc il existe un entier B_0 vérifiant $B_0 \equiv b_f \mod (2a_f/r)$ et $B_0 \equiv b_g \mod (2a_g/r)$. Finalement l'ensemble des entiers satisfaisant les deux premières congruences du système (S) s'écrit de façon unique (modulo $\mathrm{ppcm}(2a_f, 2a_g)$)

$$B = ((2a_f a_g)/(r \cdot \mathrm{pgcd}(a_f, a_g))k + B_0$$

pour un entier k.

Il reste donc à montrer qu'il existe une valeur de k pour laquelle B satisfait la troisième congruence du système (S) :

$$B^2 \equiv \Delta \mod (4a_f a_g/r^2).$$

En remplaçant B, on voit que l'équation $B^2 \equiv \Delta \mod (4a_f a_g/r^2)$ est équivalente à $kB_0 \equiv \frac{(r\cdot\mathrm{pgcd}(a_f,a_g))(\Delta - B_0^2)}{4a_f a_g} \pmod{(\mathrm{pgcd}(a_f, a_g)/r)}$.
On remarquera que par définition de B_0 et de Δ, la fraction $\frac{(r\cdot\mathrm{pgcd}(a_f,a_g))(\Delta - B_0^2)}{4a_f a_g}$ est bien un entier puisque $\Delta - B_0^2 \equiv 0 \pmod{2a_f/r}$ et $\Delta - B_0^2 \equiv 0 \pmod{2a_g/r}$.
Si $\mathrm{pgcd}(a_f, a_g)/r$ divise B_0 alors il divise aussi b_f et b_g et donc $\mathrm{pgcd}(a_f, a_g)/r$ diviserait aussi ${}^{(b_f+b_g)}\!/_2$, ce qui n'est pas possible par définition de r. Donc B_0 est bien inversible modulo $\mathrm{pgcd}(a_f, a_g)/r$, et l'entier k vérifiant que $B^2 \equiv \Delta \mod (4a_f a_g/r^2)$ existe bien. □

Composer deux formes f et g de $\mathfrak{F}_\Delta$ revient donc à calculer l'entier B de la proposition 5.2. Ce calcul n'est pas unique puisque B est défini modulo $(2a_f a_g)/(r \cdot \mathrm{pgcd}(a_f, a_g))$ et un moyen de le calculer est donné dans l'algorithme 8 de composition de deux formes de $\mathfrak{F}_\Delta$.

Algorithme 8 : ***CompositionGauss***

Entrée : $f = (a_f, b_f, c_f)$ et $g = (a_g, b_g, c_g)$ de $\mathfrak{F}_\Delta$
Sorties : $h = f \star g$ de $\mathfrak{F}_\Delta$

1: $r \leftarrow \text{pgcd}(a_f, a_g, {}^{(b_f+b_g)}\!/_2)$
2: Calculer trois entiers x, y et z vérifiant $a_g x + a_f y + ({}^{(b_f+b_g)}\!/_2)z = r$
3: $B \leftarrow ({}^{b_f a_g}\!/_r)x + ({}^{b_g a_f}\!/_r)y + ({}^{(\Delta+b_f b_g)}\!/_{2r})z \mod {}^{2a_f a_g}\!/_{(r\cdot\text{pgcd}(a_f,a_g))}$
4: $C \leftarrow \frac{B^2-\Delta}{4a_f a_g/r^2}$
5: **retourner** $({}^{a_f a_g}\!/_{r^2}, B, C)$

Pour montrer que l'algorithme 8 est correct, on montre que l'entier B de l'algorithme vérifie le système (S) de la proposition 5.2. Car dans ce cas, d'après le théorème 5.2 l'algorithme 8 est correct. On notera que la définition des entiers x, y, z de l'algorithme 8 implique que $b_f a_g x + b_g a_f y + ({}^{(\Delta+b_f b_g)}\!/_2)z$ est divisible par r.

5.2 Relation entre les classes de $\mathfrak{F}_{\Delta_1}$ et les classes de $\mathfrak{F}_{\Delta_\mathbf{q}=\mathbf{q}^2\Delta_1}$

Dans ce chapitre nous énonçons les relations entre les ensembles $\mathfrak{F}_{\Delta_1}$ et $\mathfrak{F}_{\Delta_q}$ des formes primitives respectivement de discriminants, Δ_1, un discriminant fondamental, et un discriminant $\Delta_q = q^2\Delta_1$ pour un nombre premier impair q Les références de cette partie sont : [9, 8, 13].

5.2.1 Matrices à coefficients entiers de déterminant q

Nous définissons une relation d'équivalence entre deux matrices de $\mathcal{M}_2(\mathbb{Z})$.

Définition 5.3 *Deux matrices $\mathcal{A}$ et $\mathcal{B}$ de $\mathcal{M}_2(\mathbb{Z})$ sont dites équivalentes si*

$$\mathcal{A} \equiv \mathcal{B} \iff \exists \mathcal{M} \in SL_2,\ \mathcal{A}\mathcal{M} = \mathcal{B}.$$

On note $\overline{\mathcal{A}}$ la classe d'équivalence de $\mathcal{A}$:

$$\overline{\mathcal{A}} = \{\mathcal{A}\mathcal{M} : \mathcal{M} \in \text{SL}_2\}.$$

On peut montrer que l'équivalence est compatible avec la multiplication à gauche par une matrice de SL_2 :

$$\forall \mathcal{N} \in \text{SL}_2 : \overline{\mathcal{N}\mathcal{A}} = \mathcal{N}\overline{\mathcal{A}}.$$

L'introduction de cette notion d'équivalence entre les matrices nous a permis de faire une nouvelle démonstration des théorèmes principaux (comme le théorème 5.4) concernant les relations entre $\mathfrak{F}_{\Delta_1}$ et $\mathfrak{F}_{\Delta_q}$.

Proposition 5.3 *Les matrices de $\mathcal{M}_2(\mathbb{Z})$ de déterminant q ont exactement $q+1$ classes d'équivalences. Et chaque classe contient une unique forme normale d'Hermite (représentante) :*

$$Q_k = \begin{pmatrix} q & k \\ 0 & 1 \end{pmatrix},\ k \in \{0, \ldots, q-1\}$$
$$Q_\infty = \begin{pmatrix} 1 & 0 \\ 0 & q \end{pmatrix}.$$

Démonstration. Si $\mathcal{Q} = \begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix}$ est une matrice de déterminant q, alors il n'y a que deux cas possibles : soit $\text{pgcd}(\gamma, \delta) = q$, soit $\text{pgcd}(\gamma, \delta) = 1$.
Commençons par le cas où $\text{pgcd}(\gamma, \delta) = 1$. Dans ce cas il existe deux entiers u et v vérifiant $\gamma u + \delta v = 1$ et on a $\mathcal{Q}\begin{pmatrix} \delta & u \\ -\gamma & v \end{pmatrix} = \begin{pmatrix} q & \alpha u + \beta v \\ 0 & 1 \end{pmatrix}$. De plus comme $\delta(\alpha u + \beta v) = qu + \beta$ il vient que $\alpha u + \beta v = k + qh$ pour un entier $k \in \{0, \cdots, q-1\}$ et un entier h. En multipliant la matrice précédente par la SL_2-matrice $\begin{pmatrix} 1 & -h \\ 0 & 1 \end{pmatrix}$ à droite il vient $\mathcal{Q} \equiv Q_k$.
Dans le cas où $\text{pgcd}(\gamma, \delta) = q$, il existe deux entiers r et s vérifiant $\gamma r + \delta s = q$, et on a $\mathcal{Q}\begin{pmatrix} \delta/q & r \\ -\gamma/q & s \end{pmatrix} = \begin{pmatrix} 1 & n \\ 0 & q \end{pmatrix}$ où $n = \alpha r + \beta s$. En multipliant la matrice précédente par la SL_2-matrice $\begin{pmatrix} 1 & -n \\ 0 & 1 \end{pmatrix}$ à droite il vient $\mathcal{Q} \equiv Q_\infty$.
Le caractère unique vient du fait que les matrices Q_ℓ pour $\ell \in \{0, .., q-1, \infty\}$ sont des formes normales d'Hermite : pour toute matrice $\mathcal{Q}$ donnée il existe un unique indice $\ell \in \{0, .., q-1, \infty\}$ vérifiant $\mathcal{Q}\mathcal{M} = \mathcal{Q}_\ell$ pour $\mathcal{M} \in \text{SL}_2$ ([13] section 2.4.2). □

Une conséquence immédiate de cette proposition est que pour toute SL_2 matrice $\mathcal{N}$ et $h \in \{0, .., q-1, \infty\}$ il existe un unique entier $\ell \in \{0, .., q-1, \infty\}$ tel que $\mathcal{N}Q_h \equiv Q_\ell$. On a donc $\mathcal{N}\overline{Q_h} = \overline{Q_\ell}$.

5.2.2 Action de composition de Q_∞ et Q_k avec $k \in \{0, .., q-1\}$

Avant d'expliciter l'action de composition de ces matrices, on rappelle que d'après la proposition 1.6 toute forme de $\mathfrak{F}_{\Delta_1}$ est équivalente à une forme dont le premier coefficient est premier à q. Ce sont généralement ces formes, de premier coefficient premier à q, que l'on prendra en considération.

L'action de composition des matrices Q_∞ et Q_k avec $k \in \{0, .., q-1\}$, sur une forme $f = (a, b, c) \in \mathfrak{F}_{\Delta_1}$ vérifiant $\text{pgcd}(a, q) = 1$, engendre les formes :

$$f.Q_k = (aq^2, q(b + 2ak), ak^2 + bk + c)$$
$$f.Q_\infty = (a, bq, cq^2)$$

de déterminant Δ_q. On a simplement appliqué la définition de l'action de composition 1.15 et la propriété 1.4.

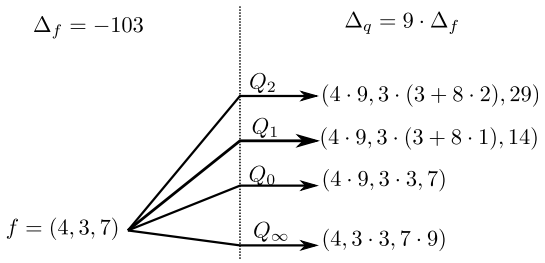


FIG. 5.1 – *Représentation de l'action de Q_∞ et Q_k sur une forme imaginaire f*

Cette figure illustre l'action de composition des matrices Q_∞ et Q_k pour $q = 3$, sur la forme imaginaire primitive $f = (4, 3, 7)$ de discriminant -103.

Les formes ainsi obtenues peuvent facilement être prouvées primitives ou non.

Dans un premier cas, la forme $f.Q_\infty$ est primitive puisque f est primitive et $\text{pgcd}(a, q) = 1$.

Dans un deuxième cas, on déduit la propriété suivante :

Propriété 5.1 *Une forme s'écrivant $f.Q_k$ avec $k \in \{0, .., q-1\}$ avec $f = (a, b, c) \in \mathfrak{F}_{\Delta_1}$ et $\text{pgcd}(a, q) = 1$, est primitive si et seulement si $(b + 2ak)^2 \neq \Delta_1 \pmod q$.*

Démonstration. Les formes $f.Q_k$ sont primitives si et seulement si $ak^2 + bk + c = \frac{1}{4a}\left((b + 2ak)^2 - \Delta_1\right) \neq 0 \pmod q$ ce qui est équivalent à la condition $(b + 2ak)^2 \neq \Delta_1 \pmod q$ quand $q \neq 2$. □

Le symbole de Legendre est noté $\left(\cdot/\cdot\right)$. Pour un nombre premier p et un entier n, le symbole de Legendre $\left(n/p\right)$ est égale à, 0 si p divise n, 1 si n est un résidu quadratique modulo p ($\exists k \in \mathbb{Z} : k^2 \equiv n \pmod p$), et -1 si n n'est pas un résidu quadratique modulo p.

Finalement on a montré la proposition suivante :

Proposition 5.4 *Pour toute forme $f = (a, b, c) \in \mathfrak{F}_{\Delta_1}$ avec $\text{pgcd}(a, q) = 1$, il y a exactement $\boldsymbol{L_q} = q - \left(\Delta_1/q\right)$ formes primitives (dans $\mathfrak{F}_{\Delta_q}$) parmi $\{f.Q_0, \ldots, f.Q_{q-1}, f.Q_\infty\}$.*

Démonstration. On pose $R = \{f.Q_0, \ldots, f.Q_{q-1}, f.Q_\infty\}$.
Si q divise Δ_1 l'équation $(b+2ak)^2 = 0 \pmod q$ admet exactement une solution, donc la propriété 5.1 implique que R contient exactement q formes primitives. Sinon si l'equation

$(b+2ak)^2 = \Delta_1 \pmod q$ admet une solution alors elle en admet exactement deux, donc comme précédemment on en déduit que si $\left(\Delta_1/q\right) = 1$ alors R contient exactement $q-1$ formes primitives, sinon il en contient exactement $q+1$. □

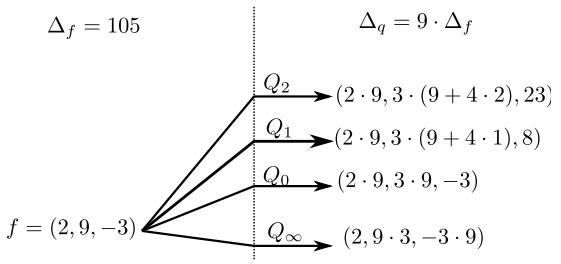


FIG. 5.2 – *Représentation de l'action de Q_∞ et Q_k sur une forme réelle*

Cette figure illustre l'action de composition des matrices Q_∞ et Q_k pour $q = 3$, sur la forme réelle primitive $(2, 9, -3)$ de discriminant 105.
On remarquera que $f.Q_0$ n'est pas primitive et $\left(\Delta_f/q\right) = 0$.
Dans cet exemple on a donc $\mathfrak{I}_f = \{1, 2, \infty\}$ et $\boldsymbol{L}_q = q$.

Nous définissons l'ensemble des indices de $\ell \in \{0, \ldots, q-1, \infty\}$ qui vérifie que $f.Q_\ell$ est primitive. Cet ensemble est donc de cardinal $\mathbf{L}_q$ d'après la proposition ci-dessus.

Définition 5.4 *L'ensemble $\mathfrak{I}_f$ de cardinal $\boldsymbol{L}_q$ pour une forme $f \in \mathfrak{F}_{\Delta_1}$ avec* $\gcd(a, q) = 1$*, est l'ensemble des indices $\ell \in \{0, \ldots, q-1, \infty\}$ qui vérifient que $f.Q_\ell$ est primitive.*

En résumé
- l'action d'une matrice de $\overline{Q_\ell}$ avec $\ell \in \{0, \ldots, q-1, \infty\}$ sur une forme f est une forme équivalente à $f.Q_\ell$
- à partir de chaque représentante de $\mathfrak{F}_{\Delta_1}$ on obtient $\mathbf{L}_q$ formes de $\mathfrak{F}_{\Delta_q}$. Les formes ainsi obtenues peuvent être équivalentes entre elles, c'est ce que l'on étudie dans la section 5.2.4.

5.2.3 Fonction *Lift* $\varphi : \mathfrak{F}_{\Delta_q} \to \mathfrak{F}_{\Delta_1}$

La fonction ***Lift*** est une application particulière de $\mathfrak{F}_{\Delta_q}$ dans $\mathfrak{F}_{\Delta_1}$.

On commence par montrer que toute forme de $\mathfrak{F}_{\Delta_q}$ est le résultat de l'action d'une matrice de $\mathcal{M}_2(\mathbb{Z})$ de déterminant q sur une forme de $\mathfrak{F}_{\Delta_1}$.

Proposition 5.5 *Pour toute forme $f \in \mathfrak{F}_{\Delta_q}$, il existe une forme $g \in \mathfrak{F}_{\Delta_1}$ telle que $g.\mathcal{Q} = f$ pour une matrice $\mathcal{Q} \in \mathcal{M}_2(\mathbb{Z})$ de déterminant q.*

Démonstration. Puisque $f = (a, b, c)$ est primitive de discriminant Δ_q, a ou c est nécessairement premier à q, f est donc équivalente par l'action d'une matrice $\mathcal{N} \in \mathrm{SL}_2$ à une forme h de premier coefficient premier à q. Ensuite en appliquant une translation $\boldsymbol{T}(n)$ pour un entier n bien choisi (cela ne change pas le premier coefficient) sur h, on peut obtenir une forme de deuxième coefficient divisible par q (voir la définition 5.5), et comme q est impair, de troisième coefficient divisible par q^2 : soit $g' = (a', b'q, c'q^2)$ cette forme. On peut donc conclure qu'il existe une forme $g = (a', b', c') \in \mathfrak{F}_{\Delta_1}$ (sinon g' ne serait pas primitive donc f non plus (car g' est équivalente à f)) et une matrice $\mathcal{Q} = Q_\infty \boldsymbol{T}(-n)\mathcal{N}^{-1} \in \mathcal{M}_2(\mathbb{Z})$ de déterminant q vérifiant $g.\mathcal{Q} = f$. □

La proposition suivante nous donne une relation entre les formes équivalentes de $\mathfrak{F}_{\Delta_q}$ et les formes de $\mathfrak{F}_{\Delta_1}$.

Proposition 5.6 *Soient deux formes f et h dans $\mathfrak{F}_{\Delta_q}$.*
Si $f \sim h$ alors il existe une forme $g \in \mathfrak{F}_{\Delta_1}$ de premier coefficient premier à q et telle que $g.Q_\infty$ est équivalente à f (et donc aussi à h).

Démonstration. D'après la proposition 1.6 la forme f est équivalente à une forme de premier coefficient premier à q : $f \sim (a, b, c)$ avec $\mathrm{pgcd}(a, q) = 1$. Elle est aussi équivalente à une forme qui s'écrit (a, Bq, Cq^2) : par l'action de la translation par $-b/2a \pmod q$, et en utilisant le fait que $\mathrm{pgcd}(a, q) = 1$ et que q est impair. On peut conclure puisque $(a, B, C).\mathcal{Q}_\infty = (a, Bq, Cq^2) \sim f \sim h$ et $(a, B, C) \in \mathfrak{F}_{\Delta_1}$ (car Δ_1 est fondamental). □

Pour compléter les relations entre les formes équivalentes de $\mathfrak{F}_{\Delta_q}$ et les formes de $\mathfrak{F}_{\Delta_1}$ nous démontrons le lemme suivant.

Lemme 5.1 *Toute matrice qui s'écrit $Q_{\ell\in\{0,..,q-1,\infty\}}\mathcal{M}Q_\infty^{-1}$ pour une matrice $\mathcal{M} \in SL_2$, et qui vérifie $g.Q_\ell\mathcal{M}Q_\infty^{-1} = g'$ pour des formes $g' \in \mathfrak{F}_{\Delta_1}$ et $g \in \mathfrak{F}_{\Delta_1}$ qui sont de premier coefficient premier à q, est forcément une matrice de SL_2 (de coefficients entiers).*

Démonstration. On pose $\mathcal{N} = Q_{\ell\in\{0,..,q-1,\infty\}}\mathcal{M}Q_\infty^{-1}$, $M = (\alpha, \beta; \gamma, \delta)$, $g = (a, b, c)$ et $g' = (A, B, C)$. On a deux cas : soit $\mathcal{N} = (\alpha, \beta/q; q\gamma, \delta)$ $(\ell = \infty)$, soit $\mathcal{N} = (q\alpha + \ell\gamma, \beta + \frac{\delta\ell}{q}; \gamma, \frac{\delta}{q})$ $(\ell \in \{0, .., q-1\})$. Dans les deux cas il vient $\mathcal{N} \in \mathcal{M}_2(\mathbb{Q})$. De plus le déterminant de $\mathcal{N}$ est égal à $\det(Q_\ell\mathcal{M}Q_\infty^{-1}) = q \cdot 1 \cdot 1/q = 1$. Ainsi pour démontrer le lemme il faut montrer que les coefficients de $\mathcal{N}$ sont entiers.
Étudions les deux cas :

- Dans le premier cas puisque $g.\mathcal{N} = g'$ et que g' est une forme (donc de coefficients entiers) on a $C = \frac{1}{q^2}(a\beta^2 + b\beta\delta q + c\delta^2q^2) \in \mathbb{Z}$. Ce qui implique $a\beta^2 \equiv 0 \pmod q$ et donc $\beta \equiv 0 \pmod q$ car $\mathrm{pgcd}(a, q) = 1$. On a montré que $\mathcal{N} \in \mathrm{SL}_2$.
- Dans le deuxième cas il vient $C = \frac{1}{q^2}(a\beta^2q^2 + 2a\delta\beta\ell q + a\delta^2\ell^2 + b\beta\delta q + b\delta^2\ell + c\delta^2) \in \mathbb{Z}$. Ce qui implique $\delta^2(a\ell^2 + b\ell + c) \equiv 0 \pmod q$ et donc $\delta \equiv 0 \pmod q$ car $g.Q_\ell (= g'.Q_\infty\mathcal{M}^{-1})$ est une forme primitive et donc $a\ell^2 + b\ell + c \not\equiv 0 \pmod q$ (comme g' est primitive et de premier coefficient premier à q, alors la forme $g'.Q_\infty$ est primitive et $g'.Q_\infty\mathcal{M}^{-1}$ l'est aussi) . On a montré que $\mathcal{N} \in \mathrm{SL}_2$.

□

On en déduit la proposition suivante.

Proposition 5.7 *Les deux ensembles de formes qui sont issus de l'action des classes de Q_∞ et Q_k avec $k \in \{0,..,q-1\}$ sur respectivement deux formes t et p non équivalentes de $\mathfrak{F}_{\Delta_1}$ et de premier coefficient premier à q, vérifient qu'il n'existe pas deux formes, une dans chacun de ces ensembles, qui sont équivalentes entre elles.*

Démonstration. On raisonne par contradiction : supposons qu'il existe deux formes f et f', une dans chacun des ensembles, qui sont équivalentes entre elles. D'après la proposition 5.6 il existe une forme $g \in \mathfrak{F}_{\Delta_1}$ de premier coefficient premier à q qui vérifie $g.Q_\infty \sim f \sim f'$. Par hypothèse on a $t.Q_{i\in\{0,..,q-1,\infty\}} \sim f$ et $p.Q_{j\in\{0,..,q-1,\infty\}} \sim f'$. On en déduit qu'il existe deux matrices $\mathcal{M}$ et $\mathcal{N}$ de SL_2 qui vérifient $t.Q_i\mathcal{M}Q_\infty^{-1} = g$ et $p.Q_j\mathcal{N}Q_\infty^{-1} = g$. On peut appliquer le lemme 5.1 qui implique que $t \sim g \sim p$ ce qui contredit que t et p ne sont pas équivalentes. □

Lorsque la forme $f = (a,b,c) \in \mathfrak{F}_{\Delta_q}$ vérifie aussi $\mathrm{pgcd}(a,q) = 1$, on définit la fonction ***Lift*** qui calcule une forme $g \in \mathfrak{F}_{\Delta_1}$ vérifiant que $g.Q_\infty \sim f$.

Définition 5.5 *La fonction particulière φ (aussi appelée **Lift**) est définit pour toute forme $f = (a,b,c) \in \mathfrak{F}_{\Delta_q}$ vérifiant* $\mathrm{pgcd}(a,q) = 1$*, et l'image de f par φ est la forme :*

$$\varphi(f) = \left(a, \frac{b+2ah}{q}, \frac{ah^2+bh+c}{q^2}\right)$$

avec $h \in [1-q,\ldots,0]$ et $h = {}^{-b}\!/_{2a} \mod q$.

On peut remarquer que toutes les divisions sont exactes :

l'entier h de la définition du ***Lift*** 5.5 implique que le deuxième coefficient de $f.\boldsymbol{T}(h) = (a, b+2ah, C)$ est divisible par q, et comme f est de discriminant Δ_q cela entraîne $4aC \equiv 0 \pmod{q^2}$, enfin, puisque q est impair et premier avec a, on en déduit que C est divisible par q^2.

On montre que $\varphi(f)$ est une forme primitive : si $\varphi(f)$ n'est pas une forme primitive alors a, b et c ont un facteur commun différent de 1 et donc f n'est pas primitive, ce qui est en contradiction avec l'hypothèse $f \in \mathfrak{F}_{\Delta_q}$.

Il est important de noter que le lift préserve le premier coefficient de la forme.

Exemple 5.2 *Le lift de la forme $(-10,25,8) \in \mathfrak{F}_{9\cdot 105}$ (q=3) est par définition la forme*

$$\left(-10, \frac{25-20h}{3}, \frac{-10h^2+25h+8}{9}\right) \text{ avec } h = 2(= 25*2 \pmod 3)$$

c'est donc la forme $\varphi((-10,25,8)) = (-10,-5,2)$ de discriminant 105.

Propriété 5.2 *Soient deux formes f et f' de $\mathfrak{F}_{\Delta_q}$ vérifiant que leur premier coefficient est premier à q :*
– Si f est normalisée primaire, alors $\varphi(f)$ l'est aussi

Démonstration. Appliquer la fonction φ sur la forme f, entraîne que l'action sur les racines de f est une translation par $-h \in [0, q-1]$ suivie d'une division par q, ce qui stabilise l'intervalle $]0,1[$ de la plus grande des racines.

□

Cette propriété 5.2 signifie que le ***Lift*** préserve la normalisation primaire.

Propriété 5.3 *Soient deux formes f et f' de $\mathfrak{F}_{\Delta_q}$ vérifiant que leur premier coefficient est premier à q :*
– Si $f \sim f'$ alors $\varphi(f) \sim \varphi(f')$ (la réciproque est en général fausse).

Démonstration. Comme $f \sim f'$ la proposition 5.6 implique qu'il existe une forme $g \in \mathfrak{F}_{\Delta_1}$ de premier coefficient premier à q qui vérifie $g.\mathcal{Q}_\infty \sim f \sim f'$. On pose $f = (a,b,c)$ et $f' = (a',b,c')$. Par hypothèse on a $\varphi(f).\mathcal{Q}_\infty \boldsymbol{T}(b/2a \pmod q) = f$ et $\varphi(f').\mathcal{Q}_\infty \boldsymbol{T}(b'/2a' \pmod q) = f'$. On en déduit qu'il existe deux matrices $\mathcal{M}$ et $\mathcal{N}$ de SL_2 qui vérifient $\varphi(f).\mathcal{Q}_\infty \boldsymbol{T}(b/2a \pmod q)\mathcal{M}\mathcal{Q}_\infty^{-1} = g$ et $\varphi(f').\mathcal{Q}_\infty \boldsymbol{T}(b'/2a' \pmod q)\mathcal{N}\mathcal{Q}_\infty^{-1} = g$. On peut appliquer le lemme 5.1 qui implique que $\varphi(f) \sim \varphi(f')$. □

Cette propriété 5.3 signifie que le ***Lift*** préserve aussi l'équivalence.

La réciproque de la proposition 5.3 est en général fausse : l'action des matrices Q_∞ et Q_k avec $k \in \{0,..,q-1\}$ sur deux formes équivalentes de $\mathfrak{F}_1$ ne sont pas forcément des formes équivalentes entre elles : nous donnons l'exemple 5.3.

Exemple 5.3 *Les deux formes $(2,9,-3)$ et $(-3,9,2)$ de $\mathfrak{F}_{105}$ sont équivalentes entre elles (figure 3.5).*

Pourtant la forme $(2,9,-3).Q_\infty$ appartient au cycle :

$$((2,27,-27)(-27,27,2)(2,29,-13)(-13,23,8)(8,25,-10)(-10,15,18)$$
$$(18,21,-7)(-7,21,18)(18,15,-10)(-10,25,8)(8,23,-13)(-13,29,2)),$$

alors que la forme $(-3,9,2).Q_\infty$ appartient au cycle :

$$((-3,27,18)(18,9,-12)(-12,15,15)(15,15,-12)(-12,9,18)(18,27,-3)).$$

L'exemple 5.4 suivant a pour objectif d'illustrer le fait que la fonction ***Lift*** n'est pas une transformation inverse des matrices Q_∞ et Q_k avec $k \in \{0,..,q-1\}$. On pourra remarquer que la fonction ***Lift*** introduit une matrice de transformation de déterminant $1/q$: si on étend l'action de composition aux matrices de $\mathcal{M}_2(\mathbb{R})$ on peut écrire $\varphi(f) = f.Tr(h)Q_\infty^{-1}$.

Exemple 5.4 *La réduite classique de $(2,9,-3).Q_2$ (figure 5.2) est $(-10,25,8)$ et la réduite classique de $\varphi((-10,25,8))$ (exemple 5.2) est la forme $(2,7,-7)$ qui appartient au cycle de $(2,9,-3)$ (figure 3.5).*

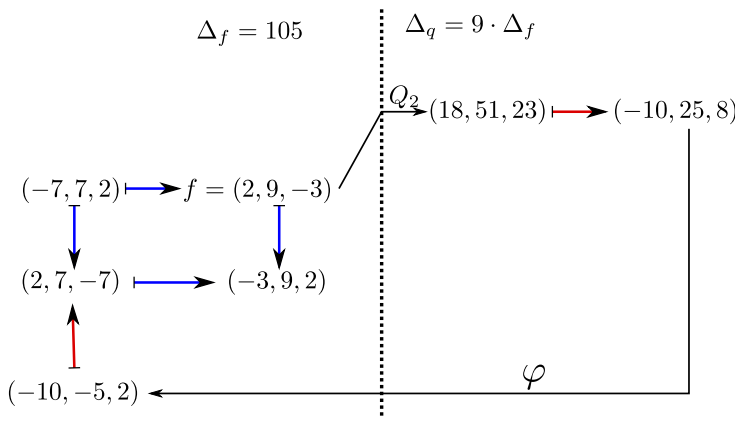


FIG. 5.3 – *Représentation de la fonction* ***Lift***

Cette figure illustre la fonction lift sur une forme de $\mathfrak{F}_{9*105}$.

Nous concluons cette section par le théorème fondamental qui lie la fonction ***Lift*** au groupe de classe $(\mathcal{O}_{\mathfrak{F}_{\Delta_1}}, \star)$ et $(\mathcal{O}_{\mathfrak{F}_{\Delta_q}}, \star)$ (voir [20]).

Théorème 5.3 *La fonction* ***Lift*** *est un morphisme de* $(\mathcal{O}_{\mathfrak{F}_{\Delta_q}}, \star)$ *dans* $(\mathcal{O}_{\mathfrak{F}_{\Delta_1}}, \star)$.

5.2.4 Nombre de classes de $\mathfrak{F}_{\Delta_q}$

La proposition suivante met en relation les formes de $\mathfrak{F}_{\Delta_q}$ issues d'une action sur la même forme de $\mathfrak{F}_{\Delta_1}$ qui sont équivalentes, et la solution fondamentale de l'équation de Pell de Δ_1.

Proposition 5.8 *Soient une forme* $g = (a, b, c) \in \mathfrak{F}_{\Delta_1}$ *avec* $\text{pgcd}(a, q) = 1$, *et deux indices distincts* i *et* j *dans* $\mathfrak{I}_g$. *Si* $\Delta_1 > 0$ *on pose* $\mathcal{N} = \mathcal{U}_g$ *l'automorphisme fondamental de* g, *et si* $\Delta_1 < -4$ *on pose* $\mathcal{N} = \pm\mathcal{I}_2$.
Les formes $g.Q_i$ *et* $g.Q_j$ *sont équivalentes si et seulement si*

$$\exists r \in \mathbb{Z} \text{ tel que } Q_i \equiv \mathcal{N}^r Q_j.$$

Pour prouver cette proposition nous avons besoin du lemme ci-dessous.

Lemme 5.2 *Tout matrice* $\mathcal{N} \in \mathcal{M}_2(\mathbb{Z})$ *de déterminant* q^2 *vérifiant* $g.\mathcal{N} = q^2 \cdot g$ *pour une forme primitive* $g = (a, b, c) \in \mathfrak{F}_{\Delta_1}$ *s'écrit*

$$\begin{pmatrix} (x - by)/2 & -cy \\ ay & (x + by)/2 \end{pmatrix}$$

avec $(x, y) \in \mathbb{Z}^2$ *une solution de* $x^2 - y^2\Delta_1 = (2q)^2$.

Démonstration. Soit une matrice $\mathcal{N} = (\alpha, \beta; \gamma, \delta) \in \mathcal{M}_2(\mathbb{Z})$ de déterminant $\alpha\delta - \gamma\beta = q^2$. Remarquons que les racines de $q^2 \cdot g$ sont exactement les racines de g ($q \neq 0$).

Comme $g.\mathcal{N} = q^2 \cdot g$ alors d'après la proposition 1.4 (sur l'action de composition des matrices de $\mathcal{M}_2(\mathbb{Z})$ sur les formes) et la remarque précédente, il existe une racine de g, que l'on note ζ_g, qui vérifie $\zeta_g = \frac{\alpha\zeta_g+\beta}{\gamma\zeta_g+\delta}$. Ce qui s'écrit aussi

$$\gamma(\zeta_g)^2 + (\delta - \alpha)\zeta_g - \beta = 0.$$

Or on a déjà que ζ_g est une racine de g :

$$a(\zeta_g)^2 + b\zeta_g + c = 0.$$

Et comme g est primitive on en déduit qu'il existe un entier k vérifiant

$$ka = \gamma,\ kb = (\delta - \alpha) \text{ et } kc = -\beta.$$

En remplaçant ces valeurs dans la formule du discriminant $(kb)^2 - 4(ka)(kc) = k^2\Delta_1$:

$$\begin{aligned}
(\delta - \alpha)^2 - 4(\gamma)(-\beta) &= k^2\Delta_1 \\
\delta^2 + \alpha^2 - 2\delta\alpha + 4\gamma\beta - k^2\Delta_1 &= 0 \\
\delta^2 + \alpha^2 + 2\delta\alpha - 4\delta\alpha + 4\gamma\beta - k^2\Delta_1 &= 0 \\
(\delta + \alpha)^2 - k^2\Delta_1 &= 4(-\delta\alpha + \gamma\beta) \\
(\delta + \alpha)^2 - k^2\Delta_1 &= 4q^2
\end{aligned}$$

on en déduit que $x = \delta + \alpha$ et $y = k$ sont des entiers solution de $x^2 - y^2\Delta_1 = (2q)^2$.

Maintenant si x et y sont des entiers solution de $x^2 - y^2\Delta_1 = (2q)^2$ on en déduit que soit $b \equiv 0 \pmod 2$ et $x \equiv 0 \pmod 2$ (si $\Delta_1 \equiv 0 \pmod 4$ alors $x^2 - 0 = 0 \pmod 2$) soit $b \equiv 1 \pmod 2$ et x et y ont la même parité (si $\Delta_1 \equiv 1 \pmod 4$ alors $x^2 - y^2 = 0 \pmod 2$). On rappelle que le deuxième coefficient des formes est de même parité que le discriminant (voir la définition 1.3 du discriminant). On en déduit aussi que $x^2 - y^2b^2 = 0 \pmod 4$. Finalement on a montré que $\frac{x-yb}{2}$ et $\frac{x+yb}{2}$ sont des entiers. Et on vérifie que $(\frac{x-yb}{2}, -yc; ya, \frac{x+yb}{2})$ est une matrice qui vérifie le lemme : elle est dans $\mathcal{M}_2(\mathbb{Z})$, de déterminant q^2, et en utilisant $x^2 - y^2(b^2 - 4ac) = (2q)^2$ il vient que $g.\mathcal{N} = (A, B, C)$

vérifie $g.\mathcal{N} = q^2 \cdot g$ car :

$$\begin{aligned}
A &= g\left(\frac{x-yb}{2}, ya\right) = a\left(\left(\frac{x-yb}{2}\right)^2 + by\left(\frac{x-yb}{2}\right) + acy^2\right)\\
&= a\left(\left(\frac{x-yb}{2}\right)\left(\frac{x+yb}{2}\right) + acy^2\right) = aq^2\\
C &= g\left(-yc, \frac{x+yb}{2}\right) = c\left(acy^2 - by\left(\frac{x+yb}{2}\right) + \left(\frac{x+yb}{2}\right)^2\right)\\
&= c\left(acy^2 + \left(\frac{x-yb}{2}\right)\left(\frac{x+yb}{2}\right)\right) = cq^2\\
B &= b\left(\frac{x^2-y^2b^2}{4} + y^2ac\right) + 2\left(a(-yc)\frac{x-yb}{2} + c(ay)\frac{x+yb}{2}\right)\\
&= b\left(-acy^2 + q^2\right) + acy(yb) = q^2b
\end{aligned}$$

□

Démontrons la proposition 5.8.

Démonstration. Le premier sens de l'équivalence est évident : si on suppose que $Q_i \equiv \mathcal{N}^t Q_j$, on a bien $g.Q_i \sim g.\mathcal{N}^t Q_j = g.Q_j$ par définition de $\equiv$ et de $\mathcal{N}$ (si $\Delta_1 > 0$ on a $\mathcal{N} = \mathcal{U}_g$, si $\Delta_1 < -4$ on a $\mathcal{N} = \pm\mathcal{I}_2$).

Dans le deuxième sens de l'équivalence, on suppose que les formes $g.Q_i$ et $g.Q_j$ sont équivalentes, il existe donc une matrice $\mathcal{M} = (\alpha, \beta; \gamma, \delta) \in \mathrm{SL}_2$ vérifiant $g.Q_i\mathcal{M} = g.Q_j$. Il vient alors l'égalité $g.Q_i\mathcal{M}Q_j^{-1} = g$ mais on ne peut pas affirmer que $Q_i\mathcal{M}Q_j^{-1} \in \mathrm{SL}_2$ (rien ne dit que ses coefficients sont entiers).
On va montrer que les matrices $Q_i\mathcal{M}Q_j^{-1}$ (de déterminant 1) qui vérifient $g.Q_i\mathcal{M}Q_j^{-1} = g$ pour une forme g primitive sont forcement à coefficients entiers, ou de façon équivalente, sont forcement des automorphismes de g (si $\Delta_1 > 0$ ces matrices sont des puissances entière de l'automorphisme fondamental $\mathcal{U}_g$, si $\Delta_1 < -4$ ces matrices sont l'identité au signe près). On aura alors prouvé la proposition.

Soient deux indices distincts h et ℓ de $\mathfrak{I}_g$, il vient :

$$\begin{aligned}
Q_\infty\mathcal{M}Q_h^{-1} &= \begin{pmatrix} \alpha/q & \beta - h\alpha/q \\ \gamma & -h\gamma + q\delta \end{pmatrix}\\
Q_h\mathcal{M}Q_\infty^{-1} &= \begin{pmatrix} q\alpha + h\gamma & \beta + h\delta/q \\ \gamma & \delta/q \end{pmatrix}\\
Q_h\mathcal{M}Q_\ell^{-1} &= \begin{pmatrix} \alpha + h\gamma/q & -l\alpha + q\beta + h\delta - \ell h\gamma/q \\ \gamma/q & \delta + -\ell\gamma/q \end{pmatrix}.
\end{aligned}$$

On pose $\mathcal{N} = Q_i\mathcal{M}Q_j^{-1}$, et d'après l'étude de cas ci-dessus la matrice $q \cdot \mathcal{N}$ est une matrice entière de déterminant q^2 vérifiant $g.(q \cdot \mathcal{N}) = q^2 \cdot (g.\mathcal{N}) = q^2 \cdot g$ car $g.\mathcal{N} = g$. On peut donc appliquer le lemme 5.2.

Ainsi il existe 3 couples d'entiers $\{(r,s),(t,u),(v,w)\}$ qui chacun vérifie l'équation $x^2 - y^2\Delta_1 = 4q^2$ et

$$Q_\infty \mathcal{M} Q_h^{-1} = \begin{pmatrix} \frac{r-sb}{2q} & -\frac{sc}{q} \\ \frac{sa}{q} & \frac{r+sb}{2q} \end{pmatrix}$$
$$Q_h \mathcal{M} Q_\infty^{-1} = \begin{pmatrix} \frac{t-ub}{2q} & -\frac{uc}{q} \\ \frac{ua}{q} & \frac{t+ub}{2q} \end{pmatrix}$$
$$Q_h \mathcal{M} Q_\ell^{-1} = \begin{pmatrix} \frac{v-wb}{2q} & -\frac{wc}{q} \\ \frac{wa}{q} & \frac{v+wb}{2q} \end{pmatrix}.$$

Nous allons montrer que ces trois égalités entraînent que la matrice du membre gauche est dans SL_2 : ce qui conclut la preuve.

Des deux première égalités impliquent $\gamma = {}^{sa}/_{q} = {}^{ua}/_{q}$ ce qui entraîne (puisque $\mathrm{pgcd}(a,q) = 1$ et $\mathcal{M} \in \mathrm{SL}_2$) que u et s sont divisibles par q. Et comme chacun des couples $\{(r,s),(t,u)\}$ vérifie l'équation $x^2 - y^2\Delta_1 = 4q^2$ cela entraîne que r et t sont aussi divisibles par q. Donc les matrices $Q_\infty \mathcal{M} Q_h^{-1}$ et $Q_h \mathcal{M} Q_\infty^{-1}$ sont bien des matrices de SL_2.

La troisième égalité implique $\alpha + \frac{h\gamma}{q} = \frac{v-wb}{2q}$ et $\frac{\gamma}{q} = \frac{wa}{q}$ ce qui est équivalent à $\alpha + \frac{hwa}{q} = \frac{v-wb}{2q}$ ou encore à $\alpha = \frac{v-wb}{2q} - \frac{2hwa}{2q} = \frac{v-w(b+2ah)}{2q}$. Et comme $\mathcal{M} \in \mathrm{SL}_2$ et que q est impair on a forcément $w(b+2ah) \equiv v \pmod q$.
Raisonnons par contradiction : si q ne divise pas w alors cette congruence est équivalente à $(b+2ah) \equiv (v/w) \pmod q$ ce qui implique $(b+2ah)^2 \equiv (v/w)^2 = \Delta_1 \pmod q$ (car $v^2 - w^2\Delta_1 = 4q^2$), ce qui n'est pas possible car g est primitive.
On en déduit que w est divisible pas q, et donc v aussi (car $v^2 - w^2\Delta_1 = 4q^2$). Donc la matrice $Q_h \mathcal{M} Q_\ell^{-1}$ est bien une matrice de SL_2.

□

Le théorème suivant permet de structurer les formes qui sont le résultat de l'action des matrices Q_∞ et Q_k avec $k \in \{0,..,q-1\}$ sur une même forme : par "structurer ces formes" on entend rassembler les formes primitives qui sont équivalentes entre elles.

Nous avons fait la démonstration de ce théorème en utilisant la notion d'équivalence entre les matrices que l'on a définit en début de chapitre. De cette manière elle contient essentiellement les notions vu dans ce manuscrit, à l'inverse des démonstrations que l'on peut trouver dans la littérature classique sur les formes.

Théorème 5.4 *Pour toute forme $g = (a,b,c) \in \mathfrak{F}_{\Delta_1}$ avec $\mathrm{pgcd}(a,q) = 1$. Si $\Delta_1 > 0$ on pose $\mathcal{N} = \mathcal{U}_g$ l'automorphisme fondamental de g, et si $\Delta_1 < -4$ on pose $\mathcal{N} = \mathcal{I}_2$.*
L'ensemble des formes $\{g.Q_i, i \in \mathfrak{J}_g\}$ se répartit en $\boldsymbol{L}_q/\boldsymbol{s}_q$ ensembles de $\boldsymbol{s}_q$ formes équivalentes, où $\boldsymbol{s}_q$ est le plus petit entier > 0 tel que $\mathcal{N}^{\boldsymbol{s}_q} = \pm\mathcal{I}_2 \pmod q$.

Démonstration. Nous définissons la fonction Ψ qui à chaque représentant Q_i avec $i \in \mathfrak{I}_g$ associe le représentant de $\mathcal{U}_g Q_i$:

$$\Psi : \begin{array}{ccc} \{\overline{Q_i}, i \in \mathfrak{I}_g\} & \to & \{\overline{Q_i}, i \in \mathfrak{I}_g\} \\ \overline{Q_i} & \to & \mathcal{N}\overline{Q_i} \end{array} .$$

Cette fonction est une permutation de $\{Q_i, i \in \mathfrak{I}_g\}$.

– En effet elle est bien définie. Soit $h \in [0, \ldots, q-1, , \infty]$ tel que $g.Q_h$ est primitive, et l'unique $\ell \in [0, \ldots, q-1, \infty]$ tel que $\Psi(Q_h) = Q_\ell$. On a donc $\mathcal{N}Q_h \equiv Q_\ell$ ce qui est équivalent à $g.Q_\ell \sim g.\mathcal{N}Q_h = g.Q_h$ donc ℓ appartient bien à l'ensemble $\mathfrak{I}_g$.
– Et elle est bijective car on vérifie que sa bijection réciproque est la fonction Ψ^{-1} qui à chaque représentant Q_i avec $i \in \mathfrak{I}_g$ associe le représentant de $\mathcal{N}^{-1}Q_i$.

Pour tout entier n, $\Psi^n(Q_i)$ est le représentant de $\mathcal{N}^n Q_i$. Les formes $g.Q_h$ et $g.Q_\ell$ sont donc équivalentes si et seulement si $\exists t \in \mathbb{Z}$ tel que $\Psi^t(Q_\ell) = Q_h$ (simple réécriture de la proposition 5.8), et donc si et seulement si Q_h et Q_ℓ sont dans le même cycle par Ψ.

Dans la cas où $\Delta_1 < -4$ on a donc $\mathbf{s}_q = 1$, et le théorème est bien prouvé.

Dans le cas où $\Delta_1 > 0$, on va montrer que la taille des cycles est la même, et que c'est l'ordre de $\mathcal{N} = \mathcal{U}_g$ modulo q que l'on note $\mathbf{s}_q$.

Commençons par chercher la longueur s_∞ du cycle de Q_∞ par Ψ. Pour tout entier k on a $\Psi^k(Q_\infty) = Q_\infty$ lorsque $\mathcal{U}_g^k Q_\infty \equiv Q_\infty$ ce qui est équivalent à la condition $Q_\infty^{-1}\mathcal{U}_g^k Q_\infty \in \mathrm{SL}_2$. Et comme $\mathrm{pgcd}(a, q) = 1$ cette dernière condition est équivalente à $y = 0 \pmod q$ pour la solution $\epsilon_{\Delta_1}^k = (x, y) \in \mathcal{P}_{\Delta_1}$, ce qui revient à $\mathcal{U}_g^k = \pm\mathcal{I}_2 \pmod q$. Dans l'autre sens si k est un entier vérifiant $\mathcal{U}_g^k = \pm\mathcal{I}_2 \pmod q$ alors $\Psi^k(\overline{Q_\infty}) = \overline{Q_\infty}$ (le pgcd des coefficients de la ligne du bas de la matrice $\mathcal{U}_g^k Q_\infty$ est encore égal à q). La longueur s_∞ du cycle de Q_∞ par Ψ est donc le plus petit entier $k > 0$ tel que $\mathcal{U}_g^k = \pm\mathcal{I}_2 \pmod q$.

Maintenant, soit $\ell \neq \infty \in \mathfrak{I}_g$ un autre indice. On cherche la longueur s_ℓ du cycle de Q_ℓ par Ψ. Comme Ψ^{s_∞} est la fonction identité on sait déjà que s_ℓ divise s_∞, on va montrer l'égalité $s_\ell = s_\infty$. Pour tout entier k on a $\Psi^k(Q_\ell) = Q_\ell$ lorsque $\mathcal{U}_g^k Q_\ell \equiv Q_\ell$ ce qui est équivalent à la condition $Q_\ell^{-1}\mathcal{U}_g^k Q_\ell \in \mathrm{SL}_2$. On remarque que

$$Q_\ell = \begin{pmatrix} \ell & -1 \\ 1 & 0 \end{pmatrix} Q_\infty \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

ce qui entraîne

$$Q_\ell^{-1}\mathcal{U}_g^k Q_\ell = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} Q_\infty^{-1} \begin{pmatrix} 0 & 1 \\ -1 & \ell \end{pmatrix} \mathcal{U}_g^k \begin{pmatrix} \ell & -1 \\ 1 & 0 \end{pmatrix} Q_\infty \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}.$$

La dernière condition est donc équivalente à

$$Q_\infty^{-1} \begin{pmatrix} 0 & 1 \\ -1 & \ell \end{pmatrix} \mathcal{U}_g^k \begin{pmatrix} \ell & -1 \\ 1 & 0 \end{pmatrix} Q_\infty \in \mathrm{SL}_2$$

ce qui est équivalent à ce que le coefficient en bas à gauche de

$$\begin{pmatrix} 0 & 1 \\ -1 & \ell \end{pmatrix} \mathcal{U}_g^k \begin{pmatrix} \ell & -1 \\ 1 & 0 \end{pmatrix}$$

soit divisible par q, ce qui s'écrit $y(a\ell^2 + b\ell + c) = 0 \pmod q$ pour la solution $\epsilon_{\Delta_1}^k = (x, y) \in \mathcal{P}_{\Delta_1}$ et comme la forme $g.Q_\ell$ est primitive, c'est équivalent à $y = 0 \pmod q$. Donc la longueur du cycle de Q_ℓ par Ψ est encore le plus petit entier $k > 0$ tel que $\mathcal{U}_g^k = \pm\mathcal{I}_2 \pmod q$. Et finalement on a montré que pour tout $\ell \neq \infty \in \mathfrak{I}_g$ la longueur du cycle de Q_ℓ par Ψ est $s_\ell = s_\infty = \mathbf{s}_q$.

Pour conclure, puisque tous les cycles sont de longueur $\mathbf{s}_q$, on sait que $\mathbf{s}_q$ divise $\mathbf{L}_q$ (le cardinal de $\mathfrak{I}_g$). Il y a donc en tout $\mathbf{L}_q/\mathbf{s}_q$ cycles de longueur $\mathbf{s}_q$. Et comme deux formes dans $\{g.Q_i, i \in \mathfrak{I}_g\}$ sont équivalentes si et seulement si elles sont dans le même cycle par Ψ, il y a autant de formes dans $\{Q_i, i \in \mathfrak{I}\}$ non équivalentes que de cycle de Ψ. □

Pour résumer :
- les $\mathbf{L}_q/\mathbf{s}_q$ classes d'équivalence différentes du théorème sont les seules telles que leurs images par la fonction ***Lift*** sont dans la classe de g,
- l'unité fondamentale ϵ_{Δ_q} est égale à la puissance $(\epsilon_{\Delta_1})^{\mathbf{s}_q}$ (simple utilisation de la démonstration du théorème 1.3),
- l'isomorphisme de la proposition 1.7 ($Aut(g) \cong \mathcal{P}_{\Delta_1}$ pour une forme primitive $g \in \mathfrak{F}_{\Delta_1}$), et le théorème 1.3 impliquent que l'entier $\mathbf{s}_q$ dépend uniquement des entiers ϵ_{Δ_1} et q.

On déduit en déduit le nombre de classes de $\mathfrak{F}_{\Delta_q} = q^2\Delta_1$.

Corollaire 5.1 *Le nombre de classes de $\mathfrak{F}_{\Delta_q}$, lorsque $\Delta_1 < -4$ ou $\Delta_1 > 0$, est*

$$\mathrm{B}_{\Delta_q} = \mathrm{B}_{\Delta_1}(\boldsymbol{L}_q/\boldsymbol{s}_q).$$

Les classes de $\mathfrak{F}_{\Delta_q}$ peuvent se construire à partir des classes de $\mathfrak{F}_{\Delta_1}$. Pour chacune des B_{Δ_1} représentantes de $\mathfrak{F}_{\Delta_1}$ on effectue la procédure suivante :
- s'assurer que le premier coefficient de la représentante a un premier coefficient premier à q (si ce n'est pas le cas appliquer la proposition 1.6)
- faire agir sur cette forme les matrices Q_∞ et Q_k avec $k \in \{0, .., q-1\}$
- parmi les q formes obtenues ne conserver que les $\mathbf{L}_q$ formes qui sont primitives
- rassembler ces $\mathbf{L}_q$ formes en $\mathbf{L}_q/\mathbf{s}_q$ classes d'équivalence différentes.

5.2.5 La q-ceinture

Nous définissons un certain ensemble de formes normalisées primaires comme étant une *q-ceinture* de discriminant Δ_q. La notion de q-ceinture n'est pas définie dans la littérature, nous l'introduisons pour étudier correctement les cryptosystèmes utilisant les relations entre $\mathfrak{F}_{\Delta_1}$ et $\mathfrak{F}_{\Delta_q}$.

Définition 5.6 *Une q-ceinture de discriminant $\Delta_q = q^2\Delta_1$ est un ensemble de formes normalisées primaires $(q^2, \ell q, *)$ de la classe principale de $\mathfrak{F}_{\Delta_q}$.*

On voit facilement que dans ce cas nécessairement : $\ell \in [\sqrt{\Delta_1} - 2q, \sqrt{\Delta_1}]$.

Nous allons montrer que une *q-ceinture* de discriminant Δ_q possible est l'ensemble des formes équivalentes à la forme $(1, b, *).Q_\infty$, et qui s'écrivent $(1, b, *).Q_k$ avec $(1, b, *)$ la forme principale de $\mathfrak{F}_{\Delta_1}$ et $k \in \{1-q, \ldots, 0\}$.
On appelle cette q-ceinture la q-ceinture de Δ_q

Proposition 5.9 *La q-ceinture de Δ_q est l'ensemble*

$$\{g_1 = f.Q_{k_1}, \ldots, g_{s_q-1} = f.Q_{k_{s_q-1}}\},$$

avec
– *f la forme principale de $\mathfrak{F}_{\Delta_1}$*
– *$\mathcal{U}_f$ l'automorphisme fondamental de f.*
La suite k_i pour $1 \leqslant i \leqslant \boldsymbol{s}_q - 1$ est définie par récurrence par :
– *$k_0 = \infty$,*
– *k_i l'unique entier $\in [1-q, \ldots, 0]$ tel que $\mathcal{U}_f Q_{k_{i-1}} \equiv Q_{k_i}$ pour $i \geqslant 1$.*

Démonstration. On montre que ainsi définit la q-ceinture de Δ_q est une q-ceinture.
On pose g_0 la forme principale de $\mathfrak{F}_{\Delta_q}$, alors nécessairement $f = \varphi(g_0)$ est équivalente à la forme principale de $\mathfrak{F}_{\Delta_1}$ car ***Lift*** est un morphisme (théorème 5.3).
Par définition de la suite k_i on peut remarquer que $Q_{k_i} \equiv \mathcal{U}_f^i Q_{k_0}$, or l'ordre de $\mathcal{U}_f$ mod q est $\mathbf{s}_q$, donc la suite (k_i) est périodique : $k_{\mathbf{s}_q} = k_0 = \infty$, et il y a exactement $\mathbf{s}_q - 1$ formes dans la q-ceinture de Δ_q. Finalement la q-ceinture de Δ_q est l'ensemble $\{g_1 = f.Q_{k_1}, \ldots, g_k = f.Q_{k_{\mathbf{s}_q-1}}\}$: ces formes sont normalisées primaires et équivalentes par construction. Une matrice de transformation de g_i à g_{i-1} pour $i \geqslant 2$ est par construction $Q_{k_i}^{-1} \mathcal{U}_f Q_{k_{i-1}} \in \mathrm{SL}_2$, car $\mathcal{U}_f Q_{k_{i-1}} \equiv Q_{k_i}$. □

Exemple 5.5 *La solution fondamentale de $\mathcal{P}_{105}$ est le couple $(82, 8)$. La forme principale $(1, 9, -6) \in \mathfrak{F}_{105}$ a donc pour automorphisme fondamental $\mathcal{U}_f = \begin{pmatrix} 5 & 48 \\ 8 & 77 \end{pmatrix}$.*

L'ordre de $\mathcal{U}_f$ modulo $q = 3$ est $\boldsymbol{s}_q = 3$ (ne dépend que de la solution fondamentale). Puisque $\left(105/3\right) = 0$, 3 formes parmi $\{f.Q_0, f.Q_{-1}, f.Q_{-2}, f.Q_\infty\}$ sont primitives, et puisque $\boldsymbol{s}_q = 3$ ces formes sont toutes équivalentes entre elles (mais ce n'est pas toujours le cas).

Comme le montre la figure 5.4 les formes $f.Q_{-1}$ et $f.Q_{-2}$ forment la 3-ceinture de discriminant $9 \cdot 105$.

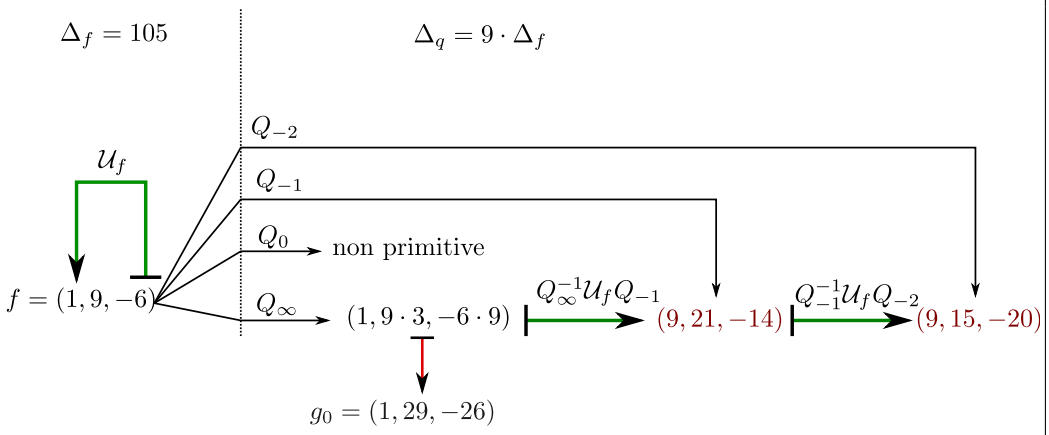


FIG. 5.4 – *Représentation de la* 3*-ceinture de discriminant* $9 \cdot 105$

Cette figure illustre la 3*-ceinture de discriminant* $9 \cdot 105$*, celle-ci est formée des formes* $g_2 = (9, 21, -14)$ *et* $g_1 = (9, 15, -20)$*.*
Ces formes sont normalisées primaires.

Chapitre 6

Cryptosystèmes *NICE*

Dans ce chapitre on étudie les schémas de chiffrement *NICE* (New Ideal Coset Encryption) ces schémas utilisent des formes de $\mathfrak{F}_N$ avec $N = \pm pq^2$ pour p et q deux nombre premiers particuliers pour chiffrer un message. Plus précisément ils utilisent les relations entre $\mathfrak{F}_N$ et $\mathfrak{F}_p$ vues dans la section 5.2, et le fait que seul celui qui connait l'entier q peut appliquer la fonction ***Lift*** sur une forme de $\mathfrak{F}_N$.

A l'origine il y a deux variantes du cryptosystème *NICE*, par ordre chronologique ce sont *NICE* imaginaire [22] (utilisant des formes imaginaires), et *NICE* réel [24] (utilisant des formes réelles). La sécurité de ces schémas de chiffrement repose sur la difficulté de factoriser le discriminant (voir [6]) qui est public.

Ces schémas sont attractifs car ils ont un niveau de sécurité apparemment identique au niveau de sécurité d'un chiffrement RSA ("apparemment" puisque le but du chapitre est de démontrer justement qu'ils sont moins sûrs) mais avec un temps de déchiffrement plus rapide (voir [35]), puisque le temps de déchiffrement d'un cryptosystème *NICE* est de complexité quadratique au lieu de cubique pour un déchiffrement RSA.

Ces deux cryptosystèmes sont considérés comme cassés. L'attaque arithmétique [11] trouve la factorisation du discriminant des formes imaginaires utilisées dans le cyptosystème *NICE* imaginaire en temps polynomial. Cependant, l'attaque la plus générale [10] qui fonctionne sur les deux versions de *NICE* et qui utilise la réduction de réseaux, reste expérimentale et liée à deux hypothèses heuristiques.

Dans ce chapitre, nous proposons d'appliquer les résultats du chapitre 4 à la cryptanalyse des cryptosystèmes *NICE*.

Nous fournissons un point de vue alternatif sur l'attaque par réduction de réseaux, ce qui nous permet de ne pas avoir besoin d'utiliser ces hypothèses heuristiques, et aussi de prouver entièrement l'attaque par réseaux.

Les deux variantes de *NICE* (réel et imaginaire) ont été décrites, à l'origine, en terme d'idéaux d'ordre quadratique.

Cette notion n'est pas nécessaire pour comprendre l'attaque par réseaux, c'est pourquoi nous allons donner une description simplement en terme de formes.

Ce travail a été effectué en collaboration avec Nicolas Gama, Maître de conférences à l'Université de Versailles Saint-Quentin-en-Yvelines. Il a donné lieu à une publication dans la conférence internationale avec comité de relecture, Algorithmic Number Theory Symposium, en juillet 2010 [2].

6.1 Schéma cryptographique *NICE* réel

Nous décrivons le schéma de chiffrement *NICE* réel.

6.1.1 Génération des clés

La clé publique est l'entier $N = pq^2$ et la clé secrète est (p, q) avec p et q deux nombres premiers distincts de même taille, qui satisfont deux conditions.

La première condition est que p doit être un nombre premier de Schinzel [37, 42] : c'est un entier positif sans facteur carré de la forme $p = A^2x^2 + 2Bx + C$ avec $(A, B, C, x) \in \mathbb{Z}^4$, $A \neq 0$ et $B^2 - 4AC$ qui divise $4\text{pgcd}(A^2, B)^2$.

Ces nombres premiers spéciaux impliquent qu'il y a vraiment très peu de formes réduites dans chaque classe.

Théorème 6.1 *Si p est un nombre premier de Schinzel alors il y a $O(\log(p))$ formes réduites dans chaque classe d'équivalence de $\mathfrak{F}_p$.*

La démonstration se trouve dans [12] et [44, Theorem 5.8, p. 52].

Ce théorème implique qu'il est faisable en pratique d'énumérer toutes les formes réduites équivalentes à une forme de $\mathfrak{F}_p$ donnée.

On rappelle qu'avec un discriminant générique le nombre de formes réduites par cycle serait exponentiel : environ $O(\sqrt{p})$) (voir la proposition 2.3).

Pour éviter la confusion on précise que, même pour un nombre premier de Schinzel, le nombre de classes dans $\mathfrak{F}_p$ reste exponentiel.

La deuxième condition est que q doit être tel que l'entier $\mathbf{s}_q$ défini dans le théorème 5.4 est linéaire en q : d'après la section 2.4.2 cela se traduit par $\mathbf{s}_q = O(q)$.

Cette deuxième condition combinée à la proposition 2.3 implique que le nombre de formes réduites de discriminant $N = q^2p$ dans chaque classe d'équivalence (que l'on abrégera

par *taille des cycles* de $\mathfrak{F}_N$) est majoré par $O(q\log(p))$ qui est exponentiel (on a vu dans la sous-section 5.2.4 l'égalité $\epsilon_N = (\epsilon_p)^{\mathbf{s}_q}$).

Ces deux conditions sont illustrées dans la figure 6.1.

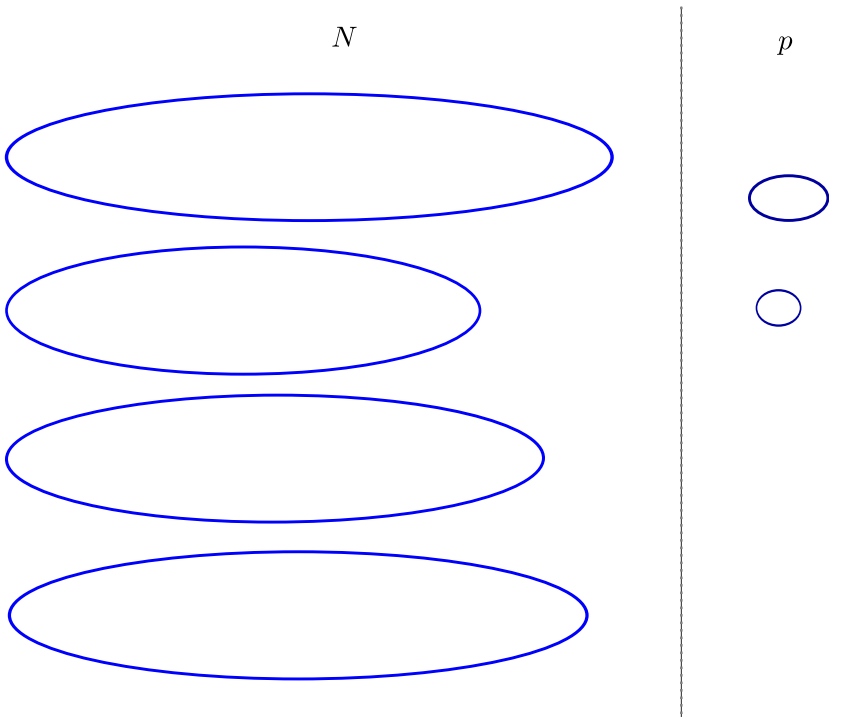


FIG. 6.1 – *Clés $N = q^2p$ et p d'un cryptosytème NICE réel*

Cette figure illustre la différence entre la taille des cycles des formes de $\mathfrak{F}_N$ qui est $O(q\log(p))$ et la taille des cycles des formes de $\mathfrak{F}_p$ qui est $O(\log(p))$, lorsque N et p sont les clés respectivement publique et privée d'un cryptosystème NICE réel.

6.1.2 Chiffrement

Le chiffrement d'un message m se fait de la manière suivante :

Le message m est intégré dans un entier (habituellement un nombre premier) $a \leqslant \sqrt{p}/2$ qui satisfait un certain motif de faible probabilité : c'est à dire que la probabilité d'avoir ce motif est faible et donc on peut supposer qu'une seul forme possède un premier coefficient qui satisfait ce motif. De plus, on impose que q^2p est un carré modulo a.

Cet entier est développé en une forme quadratique normalisée $f_s = (a, b', c')$ (qui n'est pas publique) de discriminant $N = q^2 p$.

Comme la forme f_s est normalisée et que son premier coefficient vérifie $a \leqslant \sqrt{p}/2 \leqslant \sqrt{N}/2$ on peut appliquer la proposition 3.4 qui indique que f_s est une forme réduite.

Le chiffré est une forme aléatoire réduite f_c équivalent à f_s (il y en a un nombre exponentiel). Il peut être généré à partir de f_s par une succession d'actions par des matrices unimodulaires aléatoires de taille polynomiale et des réductions.

Ce schéma de chiffrement est illustré dans la figure 6.2.

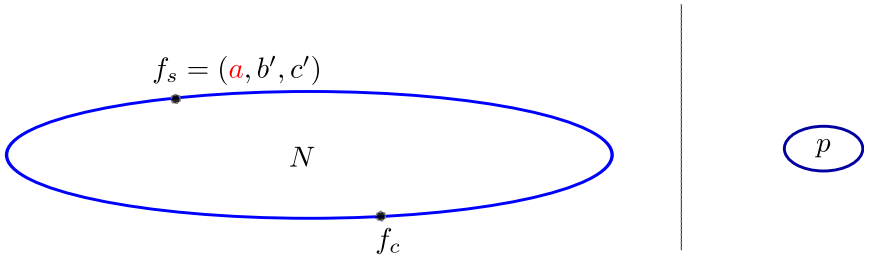


FIG. 6.2 – *Chiffrement d'un message par le cryptosytème NICE réel*

Cette figure illustre le chiffrement d'un message m par le cryptosytème NICE réel. Le message m est représenté par la forme f_s qui reste secrète, alors que le chiffré est la forme f_c.

6.1.3 Déchiffrement

L'algorithme de déchiffrement applique la fonction ***Lift*** sur le chiffré f_c (on a $\varphi(f_c) \in \mathfrak{F}_p$) et énumère toutes les formes réduites équivalentes à $\varphi(f_c)$, à la recherche d'une forme qui satisfait le motif.

Ce schéma de déchiffrement est illustré dans la figure 6.3.

Bien sûr, la connaissance de q est nécessaire pour calculer le ***Lift*** φ.

Il y a seulement $O(\log(p))$ formes réduites équivalentes à $\varphi(f_c)$ (théorème 6.1).

Cet algorithme de déchiffrement trouvera nécessairement la forme qui satisfait le motif.

Pour le montrer on note en premier que le lift de f_c est une forme équivalente à $\varphi(f_s)$: comme $f_c \sim f_s$ la propriété 5.3 implique $\varphi(f_c) \sim \varphi(f_s)$. De plus on remarque que la forme $\varphi(f_s)$ est de premier coefficient a (le premier coefficient de f_s puisque le ***Lift*** ne modifie pas le premier coefficient), et que sa normalisation $(a, *, *)$ est réduite en raison

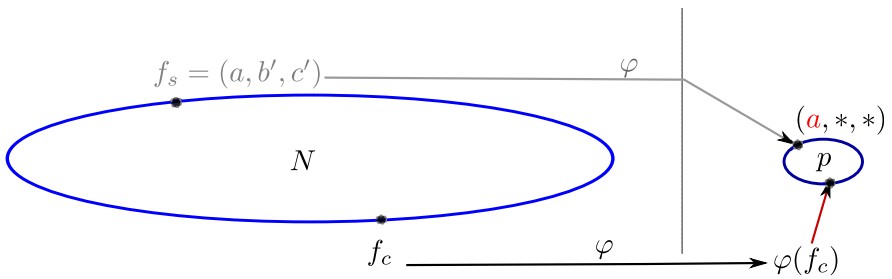


FIG. 6.3 – *Déchiffrement d'un message par le cryptosytème NICE réel*

Cette figure illustre le déchiffrement d'un message m par le cryptosytème NICE réel. La chiffré f_c est tout d'abord lifté dans $\mathfrak{F}_p$ puis la forme obtenue est réduite et l'on parcourt son cycle juqu'à trouver une forme dont le premier coefficient satisfait le motif : à partir de ce premier coefficient a on extrait le message m.

de la petite taille de a (voir la proposition 3.4). Enfin par construction a satisfait le motif.

En raison du petit nombre de formes réduites, il n'y a probablement qu'une seule forme dans le cycle de réduites de $\varphi(f_c)$ qui satisfait le motif, et le message m est finalement extrait de a.

6.2 Cryptanalyse de *NICE* réel

La cryptanalyse de *NICE* Réel présenté dans [10] fonctionne comme suit.

Les auteurs présentent un algorithme inspiré des méthodes de Coppersmith (voir [14, 32]), qui résout en temps polynomial l'équation

$$au^2 + buv + xv^2 = 0 \mod q^2$$

en les variables (u, v, q) où $N = pq^2$ est connu, et $\max(|u|, |v|) = O(N^{1/9})$. Ils appellent cet algorithme HomogeneousCoppersmith dans [10].

Leur cryptanalyse de *NICE* Réel de clé publique $N = q^2p$ et de clé secrète (p, q) s'organise d'après le plan suivant :

1. générer aléatoirement une forme g du cycle principal $\mathbb{1}_N$
2. essayer de résoudre l'équation $g(u, v) = 0 \mod q^2$ avec HomogeneousCoppersmith
3. répéter les étapes 1 et 2 jusqu'à ce qu'on trouve une solution (u, v, q)
4. retourner la clé privée $(N/q^2, q)$.

On calcule la forme principale de $\mathfrak{F}_N$ à l'aide de la clé publique N. Pour générer une forme aléatoirement dans le cycle principal on peut appliquer à la forme principale (ou bien à une forme réduite du cycle principal) une succession d'actions par des matrices unimodulaires aléatoires de taille polynomiale et des réductions .

Au lieu de prendre une forme aléatoire g, les auteurs de [10] énumèrent les formes de $\mathbb{1}_N$ de façon séquentielle en partant de la forme principale jusqu'à obtenir une forme qui fournisse la clé privée.

Ils ont alors besoin de prendre en considération : non seulement le grand nombre des formes à partir desquelles on retrouve la clé privée, mais aussi la répartition régulière sur le cycle $\mathbb{1}_N$ de ces formes.

Choisir une forme g aléatoirement évite d'avoir à prouver l'hypothèse sur la répartition régulière des formes à partir desquelles on peut retrouver la clé privée (hypothèse 2 dans [10]), il nous suffit de montrer que le nombre de formes sur le cycle à partir desquelles on peut retrouver la clé privée est grand (ce résultat est démontré dans le théorème 6.2). La forme g se trouve n'importe où sur le cycle mais ses coefficients restent de taille polynomiale (puisqu'elle est réduite),

La preuve de l'attaque de [10] repose sur l'hypothèse heuristique :

Hypothèse 6.1 *Le cardinal de l'ensemble*

$$\mathcal{A} = \{g \in \mathbb{1}_N,\ \exists (u,v)\ \max(|u|,|v|) \leqslant O(N^{\frac{1}{9}})\ \textit{et}\ g(u,v) = 0 \mod q^2\}$$

est linéaire en $\boldsymbol{s}_q$.

Les auteurs de [10] vérifient expérimentalement cette hypothèse.

A savoir, si $\bar{g}_k$ désigne la réduction de la forme $g_k = (q^2, *, *)$ de la q-ceinture (la q-ceinture est définie dans la section 5.2.5) par la réduction classique de Gauss : soit $(\alpha, \beta; \gamma, \delta) \in \mathrm{SL}_2$ la matrice de réduction de g_k, on a donc $g_k.(\alpha, \beta; \gamma, \delta) = \bar{g_k}$. Par définition de l'action des matrices de réduction : les deux coefficients du bas de la matrice de réduction satisfont $\bar{g}_k(\delta, -\gamma) = q^2$ puisque on a $g_k = \bar{g_k}.(\alpha, \beta; \gamma, \delta)^{-1} = \bar{g_k}.(\delta, -\beta; -\gamma, \alpha)$.

Alors HomogeneousCoppersmith récupère expérimentalement $(\delta, -\gamma)$ pour la plupart des $\bar{g}_k$ et même quelques-unes de leurs voisines directes à gauche ou à droite sur le cycle principal $\mathbb{1}_N$. Cela indique que la norme de la matrice de réduction est en général majorée par $O(N^{1/9})$.

L'analyse dans [4] de l'algorithme classique de réduction des formes g_k indique seulement que la norme de la matrice de réduction fournie par l'algorithme est majorée par $O(N^{2/3}) = O(\|g_k\|)$:
– les formes g_k sont normalisées et par définition de la q-ceinture on a donc $\|g_k\| = q^2 \approx N^{2/3}$ (p et q étant de même taille on peut écrire $N^{1/3} \approx q$).

Les auteurs de [10] ne peuvent donc pas prouver que HomogeneousCoppersmith trouve une solution pour chaque forme $\bar{g}_k$ en temps polynomial.

Nous appelons les formes $\bar{g}_k$ qui vérifient que la norme de la matrice de réduction est plus grande que $N^{1/9}$ des formes *déséquilibrées*, car elles ont en général un premier ou dernier coefficient de taille extrêmement petite.

Les trois principales difficultés qui ont empêché les auteurs de [10] de prouver l'hypothèse 6.1 sont :
– justifier que la proportion de formes déséquilibrées est négligeable parmi l'ensemble des $\{\bar{g}_k\}$,
– prouver que pour une majoritée des formes g_k il existe une matrice de réduction dont la norme est majorée par $O(N^{1/9})$,
– montrer que la réduction classique de Gauss est injective pour un assez grand sous-ensemble de formes de la q-ceinture, ce qui empêche que $\{\bar{g}_k\}$ ne soit trop petit.

6.2.1 Preuve et amélioration de la Cryptanalyse

Notre première amélioration dans leur analyse est de remplacer la réduction classique de Gauss par notre algorithme ***RéductionRéelle*GL2**. Cela permet de passer à la racine carrée la majoration de la norme des matrices de réduction (théorème 4.2).

Ainsi, nous définissons $\hat{g}_k$ comme étant la réduction par ***RéductionRéelle*GL2** des formes g_k de la q-ceinture pour chaque k.

Nous nous assurons que $\hat{g}_k$ est réduite de façon classique et que la matrice de réduction a un déterminant égal à $+1$ à l'aide de lemme 4.2.

Le premier point du théorème 4.2 implique que la norme de la matrice de réduction est en $O(N^{1/9})$ dès que le plus petit coefficient de $\hat{g}_k$ est supérieur à $N^{4/9}$.

Nous pouvons soit prouver que cette condition est remplie par une grande partie des formes g_k soit nous pouvons aussi contourner cette limitation en utilisant le deuxième point du théorème 4.2, qui indique que la taille du produit $|uv|$ est toujours majoré par $O(N^{1/6})$.

Nous avons donc amélioré l'algorithme HomogeneousCoppersmith de façon à ce qu'il trouve également des solutions déséquilibrées : à savoir, nous concevons une variante rationnelle de l'algorithme Boneh-Durfee-HowgraveGraham [5] qui résout en particulier

$$g(u,v) = au^2 + buv + cv^2 = 0 \mod q^2$$

en (u,v,q) dès que le produit $|uv|$ est en $O(N^{2/9})$.

Notre nouvelle attaque polynomiale sur *NICE* réel suit le schéma suivant :

1. choisir au hasard une forme g sur le principal cycle $\mathbb{1}_q$
2. essayer de résoudre $g(u,v) = 0 \mod q^2$ en (u,v,q) en utilisant ***Rationel Boneh-Durfee-HowgraveGraham***,
3. répéter les étapes 1 et 2 jusqu'à ce qu'on trouve une solution (u,v,q),
4. retourner la clé privée $(N/q^2, q)$.

La preuve de cette attaque fonctionne en deux étapes.

D'abord, nous montrons (dans le théorème 6.2) que les formes $\hat{g}_k$ définies ci-dessus représentent une proportion non négligeable du cycle principal.

Ensuite on prouve (dans le théorème 6.3 de la section 6.2.3) que ***Rationel Boneh-Durfee-HowgraveGraham*** trouve q à partir d'une forme $\hat{g}_k$ en temps polynomial.

6.2.2 Démonstration du théorème 6.2

Théorème 6.2 *Etant donné un module de NICE Réel $N = q^2p$, l'ensemble*

$$\mathcal{A}' = \{\hat{g}_k = \textbf{\textit{RéductionRéelleGL2}}(g_k),\ k \in \{1, \dots \boldsymbol{s}_q - 1\}\}$$

des formes réduites de la q-ceinture a au moins $K.\boldsymbol{s}_q$ éléments pour une certaine constante $K > 0$.

Démonstration. Soit U_p l'automorphisme fondamental de la forme principale de $\mathfrak{F}_p$.

Le corollaire 3.4 entraîne que pour tout indice j on a $|||U_p^j||| \leqslant 2(\epsilon_p^j + \epsilon_p^{-j})$.

Par définition de la q-ceinture et d'après la proposition 5.9 pour tout i, j, la matrice

$$Q_{k_{i+j}}^{-1} U_p^j Q_{k_i}$$

transforme g_{i+j} en g_i.

La norme de cette matrice est majorée par $4q\epsilon_p^j$:

$$\begin{aligned}
|||Q_{k_{i+j}}^{-1} U_p^j Q_{k_i}||| &\leqslant |||Q_{k_{i+j}}^{-1}|||\,|||U_p^j|||\,|||Q_{k_i}||| \\
&\leqslant \|Q_{k_{i+j}}^{-1}\|_2 \|Q_{k_i}\|_2 |||U_p^j||| \\
&\leqslant \sqrt{1/q^2 + (q-1)^2/q^2 + q^2/q^2} \cdot \sqrt{q^2 + (q-1)^2 + 1} \cdot |||U_p^j||| \\
&\leqslant 2(q - 1 + 1/q) \cdot |||U_p^j||| \\
&\leqslant 2q \cdot 2(\epsilon_p^j + \epsilon_p^{-j})
\end{aligned}$$

En utilisant les paramètres de NICE réel ($N = q^2p$, les entiers p et p sont de même taille donc $p \approx N^{1/3}$ et $q \approx N^{1/3}$, p est un premier de Schinzel donc du théorème 6.1 et de la proposition 2.3 on déduit $\epsilon_p \approx p$) on sait que $4q\epsilon_p^j \leqslant \epsilon_p^{\mathbf{s}_q/2} = \sqrt{\epsilon_N}$ dès que $1 \leqslant j \leqslant (\mathbf{s}_q/2) - 3$:

$$\begin{aligned}
4q\epsilon_p^j &\leqslant 4q\epsilon_p^{\mathbf{s}_q/2-3} \\
&\leqslant \sqrt{\epsilon_N}\frac{4q}{\epsilon_p^3} && (4q/\epsilon_p^3 \approx 4/N^{2/3})
\end{aligned}$$

donc le théorème 4.4 s'applique :

$$\mathbf{dist}(g_i, g_{i+j}) = \log(|||Q_{k_{i+j}}^{-1} U_p^j Q_{k_i}|||).$$

On peut minorer la distance $\mathbf{dist}(g_i, g_{i+j})$ pour tout indice $j \in [1, (s_q/2) - 3]$:

$$\begin{aligned}\mathbf{dist}(g_i, g_{i+j}) = \log(\|\!| Q_{k_{i+j}}^{-1} U_p^j Q_{k_i} |\!\|) &\geqslant \log(\|\!| U_p^j |\!\| / (\|\!| Q_{k_{i+j}} |\!\| \, \|\!| Q_{k_i}^{-1} |\!\|)) \\ &\geqslant j \log(\epsilon_p) - \log(2q),\end{aligned}$$

car la norme triple se minore par la plus grande des valeurs propres, et

$$\|\!| U_p^j |\!\| < \|\!| Q_{k_{i+j}} |\!\| \, \|\!| Q_{k_{i+j}}^{-1} U_p^j Q_{k_i} |\!\| \, \|\!| Q_{k_i}^{-1} |\!\|.$$

Puisque la norme Euclidienne induite des matrices de réduction de g_i à $\hat{g}_i$ est majorée par 2 fois leur norme infinie, par le théorème 4.2 il vient que la norme Euclidienne induite des matrices de réduction de g_i (qui est normalisée primaire) à $\hat{g}_i$ est majorée par :

$$2 \cdot \frac{21q^2}{\sqrt{N}} = \frac{42q}{\sqrt{p}},$$

(dans l'énoncé du théorème 4.2 la norme maximale de la matrice de réduction est $|\delta| \leqslant |\gamma\delta|$) puis en utilisant la définition de la distance 4.5 et sa troisième propriété il vient

$$\mathbf{dist}(g_i, g_{i+j}) \leqslant \mathbf{dist}(g_i, \hat{g}_i) + \mathbf{dist}(\hat{g}_i, \hat{g}_{i+j}) + \mathbf{dist}(\hat{g}_{i+j}, g_{i+j})$$

donc

$$\begin{aligned}\mathbf{dist}(\hat{g}_i, \hat{g}_{i+j}) &\geqslant \mathbf{dist}(g_i, g_{i+j}) - 2\log(42q/\sqrt{p}) \\ &\geqslant j \log(\epsilon_p) - \log(2q) - 2\log(42q/\sqrt{p}) \\ &\geqslant j \log(\epsilon_p) - \log\left(\frac{3528 \cdot q^3}{p}\right).\end{aligned}$$

Pour cette raison, si $j > \log\left(\frac{3528 \cdot q^3}{p}\right) / \log(\epsilon_p)$, alors

$$\mathbf{dist}(\hat{g}_i, \hat{g}_{i+j}) > 0 \text{ et } \hat{g}_i \neq \hat{g}_{i+j}.$$

En utilisant les paramètre de *NICE* Réel, dès que $N^{1/3} > 3528$ on a

$$\log\left(\frac{3528 \cdot q^3}{p}\right) / \log(\epsilon_p) \approx \frac{\log(3528 \cdot N^{2/3})}{\log(N^{1/3})} < 3,$$

ainsi les formes $\hat{g}_1, \hat{g}_4, \hat{g}_7, \ldots, \hat{g}_{3n-1}$ sont distinctes (avec $n \leqslant (s_q/6)$). □

6.2.3 Amélioration rationnelle de l'algorithme de Boneh-Durfee-HowgraveGraham

Dans cette section, nous décrivons notre algorithme ***Rationel Boneh-Durfee-HowgraveGraham*** qui est une variante de l'algorithme de Boneh Durfee Howgrave-Graham [5].

Il résout les équations linéaires rationnelles

$$u/v - C = 0 \mod q$$

en les variables (u, v, q) quand un multiple de $N = pq^r$ est connu.

La description de ***Rationel Boneh-Durfee-HowgraveGraham*** est résumée dans l'algorithme 9.

Entre autres, il peut être utilisé pour résoudre toutes les équations

$$\hat{g}_k(u, v) = au^2 + buv + cv^2 = 0 \mod q^2$$

avec la forme $\hat{g}_k$ (de discriminant pq^2) de la section précédente, car $\hat{g}_k(u, v) = av^2((u/v + b/2a)^2 - \Delta/4a^2)$ donc ces équations sont équivalentes à

$$u/v + b/2a = 0 \mod q.$$

Comme la solution que l'on cherche satisfait $|uv| = O(N^{1/6})$, le théorème 6.3 qui suit prouve que ***Rationel Boneh-Durfee-HowgraveGraham*** trouve toutes les solutions $|uv| = O(N^{2/9})$, et conclut la preuve de notre nouvelle attaque contre *NICE* Réel.

Plus généralement, étant donné un polynôme P, la technique de Boneh Durfee Howgrave-Graham transforme l'équation $P(u/v) = 0 \mod q$, en un réseau L de dimension m de déterminant borné, et dont les vecteurs courts sont orthogonaux au vecteur entier $S = (u^m, u^{m-1}v, ..., uv^{m-1}, v^m)$.

Les solutions u et v peuvent être extraites à partir de n'importe lequel de ces vecteurs courts.

Ce réseau est décrit par une base B, dont les lignes contiennent les coefficients de polynôme de degré $(m - 1)$ qui ont u/v comme racine modulo une puissance de q.

Lorsque u et v ont environ la même taille (comme dans HomogeneousCoppersmith de [10]), le célèbre algorithme de réduction de réseau LLL sur B retourne directement le vecteur désiré orthogonal à S.

Sinon, si u et v sont déséquilibrés, disons par exemple que u est 1000 fois plus grand que v, il faut d'abord rétablir l'équilibre du réseau en multipliant chaque i-ième colonne par C^i, où C est proche de 1000, et ensuite seulement on réduit la base.

L'algorithme original de Boneh-Durfee-HowgraveGraham, qui s'intéresse aux solutions entières (un entier arbitraire u avec $v = 1$), suit la règle ci-dessous :

la base du réseau qui est en fait LLL-réduite est la base de HomogeneousCoppersmith où chaque i-ème colonne a été multipliée par X^i, où X est une puissance de 2 seulement plus grande que la solution u.

Plus généralement, si nous ne savons pas l'équilibre relatif entre u et v, mais seulement que la taille du produit uv est sur n-bits, alors nous pouvons tester les n puissances possibles des deux suites, ce qui coûte un facteur linéaire.

Théorème 6.3 *Étant donné un entier $N = pq^r$ (avec p et q inconnus), la taille de q^r, un entier $r \geqslant 2$ et une borne $\beta < \frac{1}{4} \cdot q^{\log(q^r)/\log(N)}$, L'algorithme 9 termine en temps polynomial , et trouve une solution (si elle existe) de l'équation $\frac{u}{v} = c \mod q$ avec (u, v) qui sont des entiers inconnus qui vérifient $|uv| < \beta$ et un entier c.*

Algorithme 9 : ***Rationel Boneh-Durfee-HowgraveGraham***

Entrée : Un entier $N \in \mathbb{N}$ de la forme pq^r (p et q sont inconnus), un entier $c \in [0, N-1]$ et une borne $\beta < \frac{1}{4} \cdot q^{\log(q^r)/\log(N)}$

Sorties : $(u, v) \in \mathbb{N}^2$ tels que $\frac{u}{v} = c \mod q$ et $|\mathrm{u}| \cdot |\mathrm{v}| < \beta$ si ils existent

1: Choisir le plus petit m tel que $\left(\sqrt{m+1} \cdot N^{\frac{1}{8r}+\frac{17}{8}}\right)^{\frac{1}{m+1}} < 1,8.$

2: $t \leftarrow \left\lfloor \frac{(m+1)\cdot\log(q^r)}{\log(N)} \right\rceil$.

3: Calculer la famille $P_k(X,Y) = N^{\max(0,\lceil\frac{t-k}{r}\rceil)} \cdot (X - cY)^k \cdot Y^{m-k}$ pour $k = [0..m]$

4: **pour** $l = 0$ à $\lfloor \log_2(\beta) \rfloor$ **faire**

5: $U \leftarrow 2^l$; $V \leftarrow \lfloor \beta/2^l \rfloor$

6: Calculer (ou mettre à jour) la famille $(P_k)_{k\in 0..m}$ sur la base monomiale $\left(\frac{X^k Y^{m-k}}{U^k V^{m-k}}\right)_{k=0..m}$, et former la matrice $B \in \mathcal{M}_{m+1}(\mathbb{Z})$

7: Réduire avec LLL B, et appeler $(\alpha_0, \ldots, \alpha_m)$ le premier vecteur

8: **pour** chaque racine rationnelle $\frac{u}{v}$ de $R(X) = \sum_{k=0}^{m} \frac{\alpha_k}{U^k V^{m-k}} X^k = 0$ **faire**

9: si $|uv| \leqslant \beta$ et $\mathrm{pgcd}(u - cv, N)$ est non trivial retourner (u, v)

10: **fin pour**

11: **fin pour**

Démonstration. Soit $(U, V) \in \mathbb{R}^2$ tel que $|u| \leqslant U$ et $|v| \leqslant V$.

Nous utilisons les même paramètres $m \in \mathbb{N}\backslash\{0\}$ et $t = \left\lfloor \frac{(m+1)\cdot\log(q^r)}{\log(N)} \right\rceil$ de l'algorithme 9.

On appelle $\mathbb{R}_m[X, Y]$ l'ensemble des polynômes homogènes de degré m, et on définit l'isomorphisme $\varphi : \mathbb{R}_m[X, Y] \to \mathbb{R}^{m+1}$ qui calcule les coordonnées d'un polynôme dans la base $\left(\frac{X^k Y^{m-k}}{U^k V^{m-k}}\right)_{k=\{0,\cdots,m\}}$.

Par exemple, $\varphi(X^k Y^{m-k}) = U^k V^{m-k}\vec{e}_k$ avec $\vec{e}_k$ qui est le k-ième vecteur de la base canonique.

Soit $(P_k)_{k\in\{0,\cdots,m\}}$ la famille

$$P_k(X, Y) = N^{\max(0,\lceil\frac{t-k}{r}\rceil)} \cdot (X - cY)^k \cdot Y^{m-k} \in \mathbb{R}_m[X, Y].$$

Par construction, toute combinaison entière linéaire

$$R \in \sum_{k=0}^{m} \mathbb{Z} \cdot P_k$$

vérifie

$$R(u, v) = 0 \mod q^t \text{ et } |R(u, v)| \leqslant \sqrt{m+1} \cdot ||\varphi(R)||_2$$

– Le premier point vient de $N^{\lceil\frac{t-k}{r}\rceil} = 0 \pmod{q^{t-k}}$ lorsque $t > k$,et de $(u - cv)^k = 0 \pmod{q^k}$ pour $k \in \{1, \cdots, m\}$:
pour $t > k$, $N^{\lceil\frac{t-k}{r}\rceil} = q^{r\cdot\lceil\frac{t-k}{r}\rceil} p^{\lceil\frac{t-k}{r}\rceil} \geqslant q^{t-k} p^{\lceil\frac{t-k}{r}\rceil}$ entraîne $N^{\lceil\frac{t-k}{r}\rceil} = 0 \pmod{q^{t-k}}$
et par hypothèse on a $u/v - c = 0 \pmod q$ ce qui implique $u - cv = 0 \pmod q$.
Il vient donc $\forall k \in \{0, \cdots, m\}$ $P_k(u, v) = 0 \pmod{q^t}$.

– Le deuxième point se montre en utilisant l'inégalité de Cauchy-Schwarz [1] :
les polynômes P_k s'écrivent

$$P_k(X, Y) = \sum_{i=0}^{k} \left((C_k^i \cdot N^{\max(0,\lceil\frac{t-k}{r}\rceil)} \cdot (-c)^{k-i}) \cdot X^i Y^{m-i} \right)$$

ce qui implique que tout polynôme R s'écrit

$$R(X, Y) = \sum_{k=0}^{m} a_k \cdot X^k Y^{m-k}$$

où a_k est un entier
et puisque $|u| \leqslant U$ et $|v| \leqslant V$ en utilisant inégalité de Cauchy-Schwarz il vient
$|R(u, v)| = |\sum_{k=0}^{m} a_k \cdot U^k V^{m-k} \cdot \frac{u^k v^{m-k}}{U^k V^{m-k}}| \leqslant \sum_{k=0}^{m} |a_k \cdot U^k V^{m-k}| \leqslant \sqrt{m+1} \cdot ||\varphi(R)||_2$.

On suppose maintenant que $\varphi(R)$ est un vecteur court du réseau généré par la base (triangulaire) $B = (\varphi(P_k))_{k\in[1,m]}$.

Par cela, nous entendons $||\varphi(R)||_2 \leqslant (1.08)^{m+1} \det(B)^{1/(m+1)}$.

Un tel vecteur peut être trouvé en faisant tourner l'algorithme LLL sur la base du réseau B (comme le montre [27]).

Dans le reste de la preuve on vérifie que quand m augmente, $\det(B)$ est assez petit pour garantir que $|R(u, v)| < q^t$, et donc que $R(u, v) = 0$ (dans $\mathbb{Z}$) :

[1] $\forall (x_1, ..., x_n) \in \mathbb{R}^n$, $\forall (y_1, ..., y_n) \in \mathbb{R}^n$: $\sum_{i=1}^{n} |x_i y_i| \leqslant (\sum_{i=1}^{n} x_i^2)^{1/2} \cdot (\sum_{i=1}^{n} y_i^2)^{1/2}$

– Nous avons la majoration

$$\frac{|R(u,v)|}{q^t} \leqslant q^{-t} \cdot \sqrt{m+1} \cdot (1.08)^{m+1} \cdot \det(B)^{1/(m+1)}$$
$$\leqslant q^{-t} \cdot \sqrt{m+1} \cdot (1.08)^{m+1} \cdot \left(N^{\sum_{k=0}^{m} \max(0,\lceil \frac{t-k}{r} \rceil)} \cdot (UV)^{m(m+1)/2}\right)^{1/(m+1)}$$

Car la matrice B est une matrice triangulaire inférieure de coefficients diagonaux égaux à $C_k^k \cdot N^{\max(0,\lceil \frac{t-k}{r} \rceil)} \cdot (-c)^{k-k} \cdot U^k V^{m-k} = N^{\max(0,\lceil \frac{t-k}{r} \rceil)} U^k V^{m-k}$, ce qui entraîne que son déterminant est égal au produit de ses coefficients diagonaux qui est $\prod_{k=0}^{m} N^{\max(0,\lceil \frac{t-k}{r} \rceil)} U^k V^{m-k} = N^{\sum_{k=0}^{m} \max(0,\lceil \frac{t-k}{r} \rceil)} \cdot (UV)^{\sum_{k=0}^{m} k} = N^{\sum_{k=0}^{m} \max(0,\lceil \frac{t-k}{r} \rceil)} \cdot (UV)^{(m+1)m/2}$.
On peut encore majorer le déterminant par :

$$\det(B) \leqslant N^{\left(\frac{t^2}{2r}+\frac{3t}{4}\right)} \cdot (UV)^{m(m+1)/2} =$$
$$q^{\left(\frac{t^2}{2r}+\frac{3t}{4}\right)\cdot\frac{\log(N)}{\log(q)}} \cdot (UV)^{m(m+1)/2} =$$
$$q^{t^2\cdot\frac{\log(N)}{2\log(q^r)}+t\cdot\frac{3\log(N)}{4\log(q)}} \cdot (UV)^{m(m+1)/2}$$

La première inégalité vient du fait que $\sum_{k=0}^{m} \max(0, \lceil \frac{t-k}{r} \rceil) \leqslant \sum_{k=0}^{t-1} \lceil \frac{t-k}{r} \rceil \leqslant \left(\frac{t^2}{2r} + \frac{3t}{4}\right)$ car $2 \cdot \sum_{k=0}^{t-1} \lceil \frac{t-k}{r} \rceil \leqslant t(\lceil \frac{t}{r} \rceil + \lceil \frac{1}{r} \rceil) \leqslant t(\frac{t}{r} + 1/2 + 1)$.
Il vient alors :

$$\frac{|R(u,v)|}{q^t} \leqslant \sqrt{m+1} \cdot (1.08)^{m+1} \cdot q^{t^2\cdot\frac{\log(N)}{2(m+1)\log(q^r)}+t\cdot\frac{3\log(N)}{4(m+1)\log(q)}-t} \cdot (UV)^{m/2}.$$

La puissance de q est un polynôme en t de racine 0 et $\frac{2(m+1)\log(q^r)}{\log(N)} - \frac{3r}{2}$, donc cette puissance est proche de 0 si t est proche de la deuxième racine.

En prenant t qui est l'arrondi de $\dfrac{(m+1)\cdot\log(q^r)}{\log(N)}$ on a :

$$
\begin{aligned}
& t^2\cdot\frac{\log(N)}{2(m+1)\log(q^r)}+t\cdot\frac{3\log(N)}{4(m+1)\log(q)}-t \\
\leqslant & \left(\frac{(m+1)\cdot\log(q^r)}{\log(N)}+\frac{1}{2}\right)^2\cdot\frac{\log(N)}{2(m+1)\log(q^r)}+\left((m+1)+\frac{1}{2}\right)\cdot\frac{3\log(N)}{4(m+1)\log(q)} \\
& -\left(\frac{(m+1)\cdot\log(q^r)}{\log(N)}-\frac{1}{2}\right)= \\
& \frac{(m+1)\log(q^r)}{2\log(N)}+\frac{\log(N)}{8(m+1)\log(q^r)}+1+\frac{3\log(N)}{4\log(q)}+\frac{3\log(N)}{8(m+1)\log(q)}-\frac{(m+1)\cdot\log(q^r)}{\log(N)}= \\
\leqslant & -\frac{(m+1)\log(q^r)}{2\log(N)}+\frac{\log(N)}{8(m+1)\log(q^r)}+1+\frac{3\log(N)}{4\log(q)}+\frac{3\log(N)}{8(m+1)\log(q)}.
\end{aligned}
$$

– Ce qui implique :

$$
\begin{aligned}
\left(\frac{|R(u,v)|}{q^t}\right)^{\frac{2}{m+1}} & \leqslant (1+\varepsilon_m)^2\cdot(1.08)^2\cdot q^{-\frac{\log(q^r)}{\log(N)}}\cdot(UV) \\
& \leqslant (1+\varepsilon_m)^2\cdot(1.08)^2\cdot q^{-\frac{\log(q^r)}{\log(N)}}\cdot\beta \\
& \leqslant (1+\varepsilon_m)^2\cdot(1.08)^2\cdot q^{-\frac{\log(q^r)}{\log(N)}}\cdot\frac{1}{4}\cdot q^{\frac{\log(q^r)}{\log(N)}} \\
& \leqslant (1+\varepsilon_m)^2\cdot(1.08/2)^2
\end{aligned}
$$

car $(1+\varepsilon_m)$ vérifie

$$
\begin{aligned}
|1+\varepsilon_m| & \leqslant \left(\sqrt{m+1}\cdot q^{\frac{\log(N)}{8(m+1)\log(q^r)}+1+\frac{3\log(N)}{4\log(q)}+\frac{3\log(N)}{8(m+1)\log(q)}}\right)^{\frac{1}{(m+1)}} \\
& = \left(\sqrt{m+1}\cdot N^{\frac{1}{8r(m+1)}+\frac{\log(q)}{\log(N)}+\frac{3}{4}+\frac{3}{8(m+1)}}\right)^{\frac{1}{(m+1)}} \\
& \leqslant \left(\sqrt{m+1}\cdot N^{\frac{1}{8r}+1+\frac{3}{4}+\frac{3}{8}}\right)^{\frac{1}{(m+1)}}
\end{aligned}
$$

et converge rapidement vers 1.

– Donc en choisissant m le plus petit entier tel que $\left(\sqrt{m+1}\cdot N^{\frac{1}{8r}+\frac{17}{8}}\right)^{\frac{1}{m+1}} < 1,8$ on garantit que $|R(u,v)| < q^t$.

Puisque R est homogène et que $R(u,v) = 0$ (dans $\mathbb{Z}$), cela permet de retrouver u et v (voir "Factoring polynomials with rational coefficients" [27]). □

6.3 Cryptosystème et cryptanalyse de *NICE* imaginaire

Nous présentons l'application de la cryptanalyse de *NICE* réel sur le cryptosystème *NICE* imaginaire.

6.3.1 Génération des clés

Dans le cas imaginaire la clé publique du cryptosystème *NICE* est composée de deux paramètres :
– l'entier $N = -q^2p$ avec p et q deux nombres premiers distincts, qui vérifient $p \equiv 3 \pmod 4$ (pour garantir que N est un discriminant) et $\sqrt{p/3} < q$ (pour garantir que toutes les formes réduites de $\mathfrak{F}_{-p}$ ont un premier coefficient premier à q puisque ce premier coefficient est forcément $< \sqrt{p/3}$ d'après la proposition 3.1)
– mais aussi une forme réduite $\bar{g} \in \mathfrak{F}_N$ différente de la forme principale de $\mathfrak{F}_N$ et vérifiant que son image par la fonction ***Lift*** est une forme équivalente à la forme principale de $\mathfrak{F}_{-p}$.

La clé secrète de *NICE* imaginaire est (p,q) (comme dans le cas réel).

Puisque $\sqrt{p/3} < q$ on déduit aussi que toutes les formes réduites de $\mathfrak{F}_N$ ont un premier coefficient premier à q (comme pour les formes réduites de $\mathfrak{F}_{-p}$) : la fonction ***Lift*** peut donc s'appliquer sur toutes les formes réduites de $\mathfrak{F}_N$.

Pour le montrer on suppose que q divise le premier coefficient d'une forme réduite de $\mathfrak{F}_N$, alors avec la formule de son troisième coefficient en fonction de N cela entraîne que q divise forcément le deuxième coefficient de la forme. Avec ces hypothèses en écrivant la formule du discriminant et puisque le premier coefficient est majoré par $q\sqrt{p/3}$ cela entraîne que q divise le troisième coefficient de la forme, ce qui contredit le fait que la forme est dans $\mathfrak{F}_N$.

De façon générale, avec les paramètres de *NICE* imaginaire, le théorème 5.4 implique que pour toute forme réduite donnée de $\mathfrak{F}_{-p}$ (forcément de premier coefficient premier à q) on a exactement $\mathbf{L}_q$ formes réduites dans $\mathfrak{F}_N$ qui vérifient que leur image par la fonction ***Lift*** est équivalente à la forme donnée ($-p$ est un discriminant fondamental).

En particulier, et puisque le théorème 5.3 indique que le ***Lift*** est un morphisme de $(\mathcal{O}_{\mathfrak{F}_N}, \star)$ dans $(\mathcal{O}_{\mathfrak{F}_{-p}}, \star)$ en posant

$$\ker(\varphi) = \{h \in \mathfrak{F}_N : \varphi(h) \sim (1, 1, (1+p)/4)\},$$

il vient $\bar{g} \in \ker(\varphi) \backslash \{(1, 1, (1+q^2p)/4)\}$, et le théorème 5.4 indique en particulier que le cardinal de $\ker(\varphi)$ est $\mathbf{L}_q = q - \left(-p/\Delta_q\right)$.

La forme $\bar{g}$ va servir à "masquer" le message.

Elle peut être générée à partir de la réduction de la forme $(1, 1, (1+p)/4).Q_i$ pour un indice $i \in \mathfrak{I} \backslash \{\infty\}$, l'ensemble $\mathfrak{I}$ correspond à la définition 5.4. On enlève l'indice ∞ car $\bar{g}$ ne doit pas être la forme principale de $\mathfrak{F}_N$.

Choisir la forme $\bar{g}$ implique de connaître la factorisation de N, ce paramètre doit donc forcément être public, et on ne peut pas le supprimer puisqu'il est nécessaire au chiffrement du message.

Or c'est justement grâce à la donnée de cette forme $\bar{g}$ dans la clé publique que notre algorithme HomogeneousCoppersmith retrouve la clé secrète de *NICE* imaginaire.

6.3.2 Chiffrement

Comme pour *NICE* réel un message m est intégré dans un nombre premier $a < \sqrt{p}/2$ tel que N est un carré modulo $4a$, mais cette fois on n'impose aucune condition sur le motif que doit satisfaire a.

Cet entier est développé en une forme quadratique normalisée $f_s = (a, b', c')$ qui est réduite d'après la proposition 3.2 (en raison de sa petite taille). L'entier b' peut se calculer à l'aide de l'algorithme RESSOL de Shanks [39].

Le chiffré, noté f_c, est la réduite de $f_s \star (\bar{g})^r$ pour un entier $r \leqslant \mathbf{L}_q$ choisit aléatoirement (la puissance de r s'interprète par la loi de gauss $\star$ vu dans la section 5.1) .

Le nombre de chiffré possible est $\mathbf{L}_q$.

6.3.3 Déchiffrement

Pour déchiffrer le message on utilise la clé secrète q pour appliquer la fonction ***Lift*** sur le chiffré f_c (de premier coefficient premier à q).

On a $\varphi(f_c) \sim \varphi(f_s)$ car $\bar{g} \in \ker(\varphi)$ et la fonction φ est un morphisme :

$$\varphi(f_c) \sim \varphi(f_s \star (\bar{g})^r) \sim \varphi(f_s) \star \varphi((\bar{g})^r) \sim \varphi(f_s) \star (1, 1, (1+p)/4) \sim \varphi(f_s).$$

Comme pour *NICE* réel, on vérifie que $\varphi(f_s)$ est de premier coefficient a (car le ***Lift*** préserve le premier coefficient), et que sa normalisation $(a, *, *)$ est réduite en raison de la petite taille de a (proposition 3.2).

Puisque dans le cas imaginaire chaque classe a une unique représentante, on en déduit que le premier coefficient de la réduite $\varphi(f_c)$ est a, dont on extrait le message m.

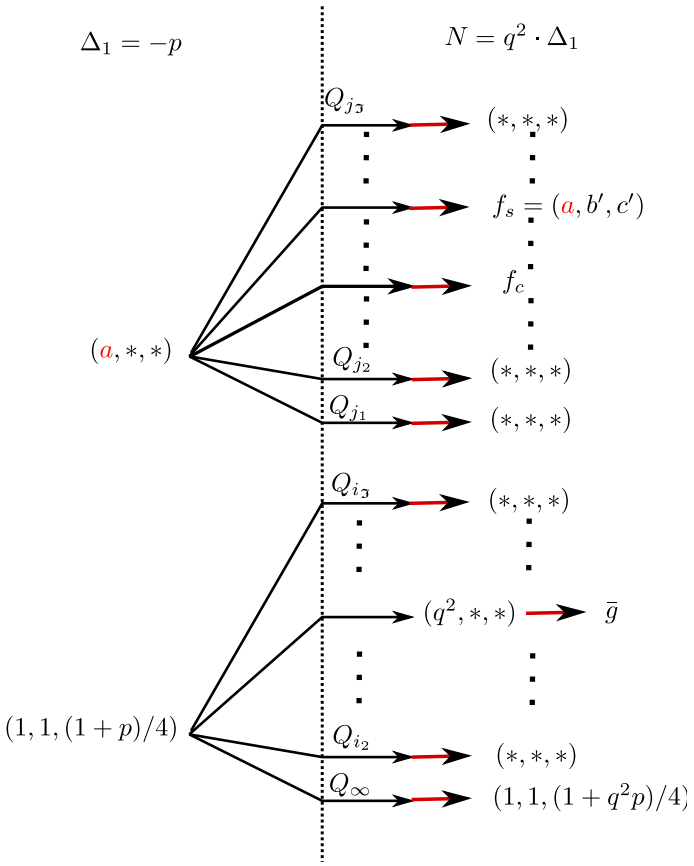


FIG. 6.4 – *Schéma du cryptosystème NICE imaginaire et sa cryptanalyse*

Cette figure illustre un cryptosystème NICE imaginaire de clé publique $(N, \bar{g})$ *et de clé secrète* (p, q) *vérifiant* $\sqrt{p/3} < q$.
Chaque représentante de $\mathfrak{F}_{-p}$ *a un premier coefficient premier à* q. *A partir de chacune de ces formes on obtient exactement* $\boldsymbol{L}_q$ *formes de* $\mathfrak{F}_N$ *qui ont pour réduite une forme de premier coefficient premier à* q, *et toutes ces formes réduites sont non équivalentes entre elles.*
La clé publique $\bar{g} \in \ker(\varphi)$ *s'exprime à l'aide d'une matrice de réduction et d'une forme* $(q^2, *, *)$ *dont la normalisation est une forme* $g = (q^2, qk, *)$.

6.3.4 Cryptanalyse

Puisque $\bar{g} \in \ker(\varphi)$ on en déduit que $\bar{g}$ s'écrit comme l'action d'une matrice de réduction $(\alpha, \beta; \gamma, \delta)$ sur une forme normalisée $g = (q^2, qk, *)$ pour un entier k, ce qu'on peut écrire :

$$g = \bar{g}.\begin{pmatrix} \delta & -\beta \\ -\gamma & \alpha \end{pmatrix}.$$

Il vient $q^2 = \bar{g}(\delta, -\gamma)$ et puisque notre théorème 4.1 donne pour les valeurs de *NICE* imaginaire $|\delta\gamma| = O(\frac{|N|^{1/3}}{|N|^{1/4}}) = O(N^{1/12})$ (en supposant que g n'est pas réduite sinon le secret est dans la clé publique) on peut conclure grâce au théorème 6.3 que HomogeneousCoppersmith ayant en entrée la clé publique de *NICE* imaginaire, récupère $(\delta, -\gamma)$ (et donc la clé secrète de *NICE* imaginaire) en temps polynomial.

Conclusion du chapitre

Comme dans le chapitre 4, nous constatons que des problèmes qui se résolvent assez simplement pour le cas imaginaire, sont de véritables défis à résoudre dans le cas réel.

Conclusion et perspectives

Nous avons vu que les algorithmes de réduction sont plus simples à étudier dans GL_2 principalement parce que nous manipulons des matrices à coefficients positifs, ce qui est plus facile à majorer.

Cela nous a permis de présenter une variante de la réduction de Gauss des formes quadratiques réelles conçue pour minimiser la norme de la réduction de la matrice, tout en gardant une complexité quadratique.

La matrice calculée par notre algorithme sur l'entrée f est de norme

$$O\left(\|f\|^{1/2}/|\Delta_f|^{1/4}\right),$$

qui est la racine carrée de la meilleure majoration connue avant ma thèse, en utilisant des algorithmes classiques.

Un futur travail serait d'étendre ces résultats à la réduction des formes en dimension supérieure.

La précision de notre analyse, dans le pire des cas et aussi dans le cas moyen, nous permet de démontrer pleinement l'attaque basée sur les réseaux des cryptosystèmes *NICE* [22, 24], ce qui est inhabituel dans l'histoire de la cryptographie à base réseau.

Cette attaque consiste à factoriser une sous-classe particulière d'entiers de la forme $\pm pq^r$.

Pour faire cette démonstration, nous avons aussi créé une variante homogène de l'algorithme Boneh-Durfee-HowgraveGraham qui trouve de petites racines rationnelles d'un polynôme modulo un diviseur inconnu.

Ce nouvel algorithme ***Rationel Boneh-Durfee-HowgraveGraham*** peut aussi être utilisé pour accélérer la factorisation de pq^r pour de grands r.

Enfin il serait intéressant de comparer plus en détails notre notion de distance entre deux formes, avec la distance de Shanks [38]. Plus particulièrement on pourrait chercher s'il existe une relation entre la distance (définie dans ce manuscrit) de deux formes et la distance de leur composée par la loi de Gauss (notée $\star$).

Index des notations

$\mathbb{C}$	Ensemble des nombres complexes
$\mathbb{R}$	Ensemble des nombres réels
$\mathbb{Q}$	Ensemble des rationnels
$\mathbb{Z}$	Ensemble des entiers relatifs
$\mathbb{N}$	Ensemble des entiers naturels
$\mathcal{M}_2(\mathbb{Z})$	Ensemble des matrices 2×2 à coefficients entiers
$\mathcal{I}_2$	Matrice identité de $\mathcal{M}_2(\mathbb{Z})$
GL_2	Groupe linéaire général des matrices de $\mathcal{M}_2(\mathbb{Z})$ de déterminant ± 1
SL_2	Groupe linéaire spécial des matrices de $\mathcal{M}_2(\mathbb{Z})$ de déterminant $+1$
f	Forme quadratique binaire
$\mathrm{pgcd}(f)$	Plus grand dénominateur commun aux coefficients de f
Δ_f	Discriminant de f
$p_f(x)$, $\mathcal{M}_f$	Représentation affine, polaire de f
ζ_f, ζ_f^-, ζ_f^+	Racine, la plus petite, la plus grande de f
$\sim$	Relation d'équivalence entre deux formes
$\mathcal{O}_f$	Orbite de l'action de SL_2 sur f
$Aut(f)$	Groupe des automorphismes de f
U_f	Automorphisme fondamental de f
Δ	Discriminant
$\mathfrak{F}_\Delta$	Ensemble des formes primitives de discriminant Δ où Δ n'est pas un carré parfait
$\mathcal{O}_{\mathfrak{F}_\Delta}$	Ensemble des orbites de l'action de SL_2 sur $\mathfrak{F}_\Delta$
B_Δ	Cardinal de $\mathcal{O}_{\mathfrak{F}_\Delta}$
ϵ_Δ	Unité fondamentale de Δ
$\mathbf{dist}(.,.)$	Distance entre deux formes
$\star$	Loi du groupe de classes de $\mathcal{O}_{\mathfrak{F}_\Delta}$
$Q_{\ell \in \{0,\cdots,q-1,\infty\}}$	Matrices $Q_k = \begin{pmatrix} q & k \\ 0 & 1 \end{pmatrix}, k \in \{0,\ldots,q-1\}$ et $Q_\infty = \begin{pmatrix} 1 & 0 \\ 0 & q \end{pmatrix}$
φ	Fonction ***Lift***

Bibliographie

[1] B. F. Agnès. Generators of sl(2,z). *McGill Mathematics Magazine*.

[2] A. Bernard and N. Gama. Smallest reduction matrix of binary quadratic forms. *Proceedings of ANTS IX, LNCS*, 2010.

[3] I. Biehl and J. Buchmann. Algorithms for quadratic orders. In *Proceedings of Symposium on Mathematics of Computation 48*, 1993.

[4] I. Biehl and J. Buchmann. An analysis of the reduction algorithms for binary quadratic forms. *Voronoi's Impact on Modern Science*, pages 71–98, 1999.

[5] D. Boneh, G. Durfee, and N. A. Howgrave-Graham. Factoring $n = p^r q$ for large r. In *Proc. of Crypto '99*, 1999.

[6] J. Buchmann, T. Takagi, and U. Vollmer. Number field cryptography, high primes and misdemeanours : Lectures in honour of the 60th birthday of h.c. williams. In V. der Poorten and Stein, editors, *Fields Institure Communications 41, AMS*, pages 111–125, 2004.

[7] J. Buchmann, C. Thiel, and H. Williams. Short representation of quadratic integers. In *Proc of CANT'92, Math. Appl. 325*, pages 159–185, 1995.

[8] J. Buchmann and U. Vollmer. *Binary Quadratic Forms An Algorithmic Approach*. 2007.

[9] D. A. Buell. *Binary Quadratic Forms Classical Theory and Modern Computations*. Springer-Verlag, 1989.

[10] G. Castagnos, A. Joux, F. Laguillaumie, and P. Q. Nguyen. Factoring pq^2 with quadratic forms : Nice cryptanalyses. In *Proc. of Asiacrypt '09*, pages 469–486, 2009.

[11] G. Castagnos and F. Laguillaumie. On the security of cryptosystems with quadratic decryption : The nicest cryptanalysis. In *Proc. of Eurocrypt'09*, volume 5479, pages 260–277, 2009.

[12] K. H. F. Cheng and H. C. Williams. Some results concerning certain periodic continued fractions. *Acta Arith. 117*, pages 247–264, 2005.

[13] H. Cohen. *A Course in Computational Algebraic Number Theory*. Springer-Verlag, 1995.

[14] D. Coppersmith. Small solutions to polynomial equations, and low exponent rsa vulnerabilities. *Journal of Cryptology 10(4)*, pages 233–260, 1997.

[15] D. A. Cox. *Primes of the Form x2 + ny2 : Fermat, Class Field Theory, and Complex Multiplication.* John Wiley and Sons, 1997.

[16] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, pages 644–654, 1976.

[17] J. P. G. L. Dirichlet. Uber die composition der binaren quadratische formen. In *Vorlesungen uber Zahlentheori*, Vorlesungen über Zahlentheorie, 1871.

[18] P. L. Dirichlet and R. Dedekind. Vorlesungen uber zahlentheorie. *Braunschweig*, sect. 82, 1893.

[19] J. Durand. Eléments de calcul matriciel et d'analyse factorielle de données. *Université Montpellier II*, 2002.

[20] C. F. Gauss. *Disquisitiones Arithrneticae.* PhD thesis, p210-280 1801.

[21] M. Granger. http://math.univ-angers.fr/~granger/ananum/Chapitre_II.pdf. *Université d'Angers*, 2011.

[22] M. Hartmann, S. Paulus, and T. Takagi. NICE - New Ideal Coset Encryption. In *Proc of CHES'99*, volume 1717, pages 328–339. Springer, 1999.

[23] D. Huhnlein, M. Jacobson, and D. Weber. Towards practical non-interactive public key cryptosystems using non-maximal imaginary quadratic orders. *Des. Codes Cryptography*, 30(3) :281–299, 2003.

[24] M. J. Jacobson, R. Scheidler, and D. Weimer. An adaptation of the NICE cryptosystem to real quadratic orders. In *Proc. of Africacrypt'08*, volume 5023, pages 191–208, 2008.

[25] J. C. Lagarias. Worst-case complexity bounds for algorithms in the theory of integral quadratic forms. *Journal of Algorithm 1*, pages 142–186, 1980.

[26] J. L. Lagrange. Recherches d'arithmétique. *Nouveaux Mémoires de l'Académie de Berlin*, 1773.

[27] A. K. Lenstra, H. W. Lenstra, and L. Lovasz. Factoring polynomials with rational coefficients. *Mathematische Ann. 261*, pages 513–534, 1982.

[28] H. W. Lenstra. On the calculation of regulators and class numbers of quadratic fields. In I. J. V. Armitage, editor, *Journees Arithmetiques 1980 LMS Lecture Notes 56*, pages 23–151, 1982.

[29] H. W. Lenstra. Solving the pell equation. *NOTICES OF THE AMS*, 49(2) :182–192, 2002.

[30] P. Malbos. http://math.univ-lyon1.fr/~malbos/Ens/AlgebreIII/notesCours.pdf. *Université Claude Bernard Lyon1*, .

[31] G. B. Mathews. *Theory of Numbers.* Chelsea Pub. Co, 1927.

[32] A. May. Using LLL-reduction for solving RSA and factorization problems : A survey. In P. Nguyen and B. Vallee, editors, *The LLL algorithm, survey and Applications*, Information Security and Cryptography, pages 315–348, 2010.

[33] P. Q. Nguyen and D. Stehlé. Low-dimensional lattice basis reduction revisited (extended abstract). *Proceedings of ANTS VI, LNCS*, 2004.

[34] D. Pastre. http://www.math-info.univ-paris5.fr/~pastre/meth-num/MN/7-normes-condi/cours-normes-conditionnement.pdf. *Université René Descartes*, 2004.

[35] S. Paulus and T. Takagi. A new public key cryptosystem over quadratic orders with quadratic decryption time. *Journal of Cryptology 13*, pages 263–272, 2000.

[36] R. L. Rivest, A.Shamir, and L. M. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2) :123–126, 1978.

[37] A. Schinzel. On some problems of the arithmetical theory of continued fractions. *Acta Arithmetica 6*, pages 393–413, 1961.

[38] D. Shanks. The infrastructure of a real quadratic field and its applications. In *Proc NTC'92*, pages 217–224, 1972.

[39] D. Shanks. Five number theoretic algorithms. *Proceedings of the Second Manitoba Conference on Numerical Mathematics*, pages 51–70, 1973.

[40] J. J. Sylvester. A demonstration of the theorem that every homogeneous quadratic polynomial is reducible by real orthogonal substitutions to the form of a sum of positive and negative squares. *Philosophical Magazine*, 4 :138–142, 1852.

[41] B. Vallee and A. Vera. Lattice reduction in two dimensions : Analyses under realistic probalistic models. In *Proc. of AofA'07*, pages 181–216. DMTCS AH, 2007.

[42] A. J. van der Poorten and H. C. Williams. On certain continued fraction expansions of fixed period length. *Acta Arithmetica 89*, pages 23–35, 1999.

[43] L. Vepstas. The minkowski question mark, gl(2,z) and the modular group. *http ://linas.org/math/chap-minkowski.pdf*, aout 2010.

[44] D. Weimer. *An Adaptation of the NICE Cryptosystem to Real Quadratic Orders, Master's thesis.* PhD thesis, Technische Universitat Darmstadt, 2004.

Titre : Formes quadratiques binaires et applications cryptographiques

Résumé : Dans cette thèse, nous étudions les formes quadratiques binaires et leur utilisation dans des schémas cryptographiques récents. Nous avons obtenu deux résultats principaux : l'un concernant l'algorithme de réduction des formes et l'autre concernant les cryptosystèmes NICE. Tout d'abord, nous présentons une modification de l'algorithme de réduction de Gauss sur les formes réelles qui permet d'obtenir la plus petite matrice de réduction en temps quadratique. Nous fournissons deux majorations quasi optimales de la norme de la plus petite matrice de réduction, l'une pour le cas des formes imaginaires et l'autre pour les formes réelles. Ces majorations sont la racine carrée des meilleures majorations connues en utilisant l'algorithme de Gauss. Deuxièmement, en nous appuyant sur nos résultats sur la réduction des formes, nous présentons une modification de la récente attaque sur les cryptosystèmes NICE. Notre modification consiste à utiliser nos majorations sur la norme de la plus petite matrice de réduction (afin d'avoir un meilleur contrôle sur la taille de la matrice de réduction) et à remplacer une variante de l'algorithme de Coppersmith par notre variante qui trouve de petites racines rationnelles d'un polynôme modulo un diviseur inconnu. Cela permet de modifier la précédente attaque heuristique en une attaque rigoureusement prouvée.

Mots-clés : forme quadratique, matrice de réduction, algorithme de réduction, réduction de Gauss, NICE, factorisation, cryptographie

Title: Binary quadratic forms and cryptographic applications

Abstract: In this thesis, we study binary quadratic forms and their use in recent cryptographic schemes. We obtained two main results : one on the reduction algorithm of forms the other on the NICE cryptosystem. First, we present a modification of the Gauss reduction algorithm of real forms allowing to obtain the smallest reduction matrix in quadratic time. We provide two near-optimal bounds of the norm of the smaller matrix reduction, one for the case of imaginary forms and one for the real forms. These bounds are the square root of the best known bounds using Gauss algorithm. Second, building on our results on the reduction of forms, we present a modification of the recent attack again the NICE cryptosystems. Our modification is to use our bounds on the norm of the smallest reduction matrix (to have better control over the size of the matrix reduction) and to replace a variant of the Coppersmith algorithm with our variant which finds small rational roots of a polynomial modulo a divisor unknown. This allows you to transform the previous heuristic attack into a fully rigorous one.

Keywords: quadratic form, reduction matrix, reduction algorithm, Gauss reduction, NICE, factorization, cryptography