Algorithmes pour les couplages en cryptographie : construction de courbes adéquates, évaluation efficace des couplages et inversion du couplage

Introduction et aperçu du sujet de recherche
pour une cotutelle de thèse
sous la direction de

Pierrick GAUDRY
LORIA (Nancy)
Centre National de Recherche Scientifique

Tanja LANGE
Dept. of Mathematics and Computer Science
Technische Universiteit Eindhoven

Résumé

L'usage constructif des couplages en cryptographie date du début du millénaire [13]. Jusque là, ils n'étaient étudiés que pour les attaques qu'ils permettent sur certains cryptosystèmes qui sont basés sur des courbes elliptiques et en particulier des courbes supersingulières [12]. En 2000, de grands problèmes jusqu'alors ouverts ont été résolus en utilisant des couplages : l'échange de clef tripartie en une étape, par JOUX [11], et le schéma pratique basé sur l'identité, par BONEH et FRANKLIN [4]. Cela a notamment été l'occasion de reconsidérer le bénéfice apporté par les courbes dans le cadre de la cryptographie asymétrique.

Depuis lors, des centaines d'articles concernant l'utilisation des couplages en cryptographie ont été publiés. La majeure partie élabore de nouveaux protocoles cryptographiques à base de couplages Les autres articles concernent les aspects algorithmiques liés à l'existence, l'évaluation efficace et la sécurité de ces couplages.

Cette thèse vise à contribuer à un ou plusieurs de ces aspects algorithmiques liés aux couplages sur des courbes hyperelliptiques dans le cadre de la cryptographie. Avant de les détailler plus précisément, nous rappelons le cadre théorique et historique concernant les couplages sur des courbes algébriques — ce sont, actuellement, les seuls couplages intéressants d'un point de vue cryptographique.

1 Courbes et couplages

1.1 Jacobiennes

Plaçons nous dans un corps fini, noté $\mathbb{F}_q$. Considérons $\mathcal{C}$, une courbe (sous-entendu : algébrique, projective et non-singulière) sur ce corps.

Notons $\mathrm{Div}_{\overline{\mathbb{F}_q}}(\mathcal{C})$ le groupe libre engendré par les points de la courbe ; on peut y étendre naturellement l'action du groupe de Galois absolu de $\mathbb{F}_q$ et ainsi considérer le sous-groupe formé des éléments stables pour cette action. Nous le noterons $\mathrm{Div}_{\mathbb{F}_q}(\mathcal{C})$; ses éléments sont les diviseurs définis sur $\mathbb{F}_q$.

En tout point P de la courbe, l'anneau local est un anneau de valuation discrète ; cette valuation s'étend naturellement au corps des fonctions de la courbe en un ordre que l'on note $\mathrm{ord}_P : \mathbb{F}_q(\mathcal{C}) \twoheadrightarrow \mathbb{Z}$. On a donc comme sous-groupe de $\mathrm{Div}_{\mathbb{F}_q}(\mathcal{C})$ l'image de l'application suivante (un peu de théorie de Galois montre en effet qu'un tel diviseur est défini sur les corps où la fonction dont il dérive est définie).

$$\mathrm{div} : f \in \mathbb{F}_q(\mathcal{C}) \mapsto \sum_P \mathrm{ord}_P(f)\, P$$

Tous les diviseurs de ce sous-groupe sont de degré nul : les coefficients de tout diviseur $\sum_P \alpha_P P$ qui dérive, par l'application div ci-dessus, d'une fonction définie sur la courbe vérifient $0 = \sum_P \alpha_P$ — notons bien que, la courbe étant lisse, cette somme a un nombre fini de termes non-nuls.

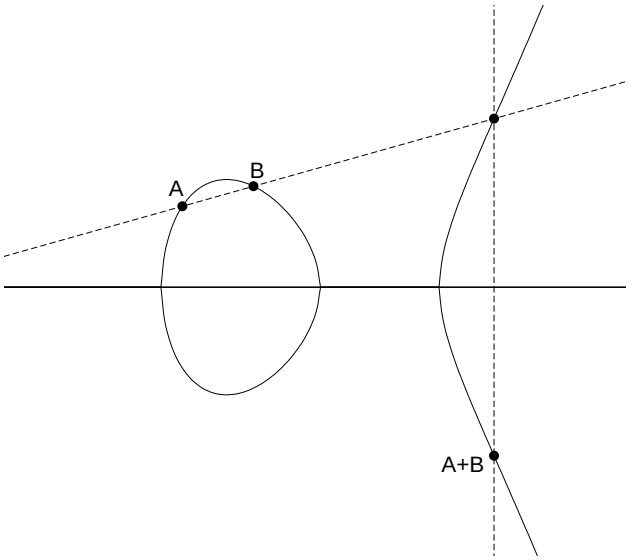
Quotientant le sous-groupe $\mathrm{Div}^0_{\mathbb{F}_q}(\mathcal{C})$ des diviseurs de degré nul définis sur $\mathbb{F}_q$ par celui des diviseurs principaux (i.e. qui dérivent d'une fonction), on obtient un groupe abélien fini appelé Jacobienne de $\mathcal{C}$ que l'on note $\mathrm{Jac}_{\mathbb{F}_q}(\mathcal{C})$.

On distingue plusieurs cas selon le genre g de la courbe.

En genre 1, les courbes sont dites elliptiques ; elles ont une équation affine de la forme suivante.

$$\mathcal{E} : Y^2 + a_1XY + a_3Y = X^3 + a_2X^2 + a_4X + a_6$$

Dans ce cas (pour plus de détails, voir [14]), la Jacobienne est tout simplement isomorphe à la courbe dont elle dérive. Par une construction géométrique bien connue (qui est rappelée à la page suivante), on peut transcrire la loi de groupe de la Jacobienne en termes de coordonnées de points rationnels. Cette géométrie admet des équations simples qui permettent ainsi de calculer très efficacement cette loi de groupe.



En genre supérieur (i.e. 2 et plus), les choses ne sont plus aussi simples : la Jacobienne est un objet bien distinct de la courbe dont elle provient.

Plus exactement, la Jacobienne d'une courbe $\mathcal{C}$ est une variété abélienne de dimension g ; on peut donc considérer ses points définis sur le corps de base $\mathbb{F}_q$ (ce qui est le cas d'intérêt en cryptographie classique car, bien évidemment, les objets considérés sont finis) ou sur toute extension de ce corps. Dans le second cas, on obtient un groupe plus gros dont les points définis sur $\mathbb{F}_q$ forment un sous-groupe.

Toutefois, on sait tout aussi bien calculer dans les Jacobiennes de courbes de genre 2 ou plus que dans celles des courbes elliptiques en utilisant la représentation de Mumford pour ses points : l'algorithme de Cantor s'inspire pour cela de l'algorithme de Gauss (qui concerne la réduction des formes quadratiques). Bref, on dispose, indépendamment du genre de la courbe considérée, de moyens algorithmiques raisonnablement efficaces pour calculer dans sa Jacobienne (relativement, bien sûr, au cardinal de cette dernière).

1.2 Torsion

Étant donné un entier $\ell \geqslant 2$, notons $[\ell]$ l'application qui consiste à additionner un diviseur n fois à lui-même. On dispose ainsi de son noyau que l'on note $\mathrm{Jac}_{\mathbb{F}_q}(\mathcal{C})[\ell]$; c'est le sous-groupe formé des points de la Jacobienne définis sur $\mathbb{F}_q$ dont l'ordre divise ℓ. Ce groupe est évidemment fini.

En fait, même lorsqu'on considère les points définis sur une clôture algébrique du corps de base plutôt que sur $\mathbb{F}_q$ lui-même, il est toujours fini. Ces groupes sont d'ailleurs classifiés comme il suit dans le cas où ℓ est premier avec la caractéristique du corps de base.

$$\mathrm{Jac}_{\overline{\mathbb{F}_q}}(\mathcal{C})[\ell] \simeq (\mathbb{Z}/\mathbb{Z})^{2g}$$

Notons aussi le résultat donné par la suite exacte ci-dessous qui est l'égalité entre les cardinaux des groupes $\mathrm{Jac}_{\mathbb{F}_q}(\mathcal{C})[\ell]$ et $\mathrm{Jac}_{\mathbb{F}_q}(\mathcal{C})/\ell\mathrm{Jac}_{\mathbb{F}_q}(\mathcal{C})$. Toutefois, il faut bien garder en tête que ces groupes sont deux objets bien distincts et, en général, non isomorphes (par exemple, il suffit pour cela que la Jacobienne admet un point de ℓ^2-torsion, ce qui arrive dans certains cas particuliers).

$$0 \longrightarrow \mathrm{Jac}_{\mathbb{F}_q}(\mathcal{C})[\ell] \to \mathrm{Jac}_{\mathbb{F}_q}(\mathcal{C}) \to \ell\,\mathrm{Jac}_{\mathbb{F}_q}(\mathcal{C}) \to 0$$

1.3 Couplages

Sur ces points de ℓ-torsion agit une application bilinéaire, non-dégénérée et alternée qui se définit de façon naturelle : le couplage de Weil, noté e_ℓ. Son domaine est celui des couples de points de ℓ-torsion et il est à valeur dans la clôture algébrique du corps de base.

$$e_\ell : \left\{ \begin{array}{rcl} \mathrm{Jac}_{\overline{\mathbb{F}_q}}(\mathcal{C})[\ell] \times \mathrm{Jac}_{\overline{\mathbb{F}_q}}(\mathcal{C})[\ell] & \to & \overline{\mathbb{F}_q} \\ (P,Q) & \mapsto & \frac{f_P(Q)}{f_Q(P)} \end{array} \right.$$

Dans ce qui précède, P étant un point de ℓ-torsion, $\ell(P-O)$ est un diviseur principal qui dérive donc d'une fonction que l'on a notée f_P. Cette fonction peut s'appliquer à Q (quitte à prendre un diviseur équivalent à support disjoint avec f_P) très naturellement par l'extension qui suit.

$$f\left(\sum_P \alpha_P P\right) := \prod_P f(P)^{\alpha_P}$$

Par linéarité, il est facile de voir que l'image du couplage est exclusivement formée de racines ℓ-ème de l'unité. Ainsi, ce couplage peut être corestreint à la plus petite extension de $\mathbb{F}_q$ contenant les racines ℓ-ème de l'unité ; cette extension est bien entendu finie : il s'agit de $\mathbb{F}_{q^k}$ où k est le plus petit entier positif non nul vérifiant la propriété $q^k \equiv 1 \bmod \ell$. S'agissant de cryptographie, lorsqu'une courbe ainsi que la valeur de l'entier ℓ sont fixés, k est communément appelé « le degré d'immersion ».

Remarquons que, lorsque ℓ est premier, il est vain d'évaluer un couplage en des points de ℓ-torsion de la Jacobienne définis dans un corps strictement plus petit que $\mathbb{F}_{q^k}$. En effet, ce corps ne peut avoir de racines ℓ-ème de l'unité non triviale. Le couplage doit donc être évalué en se basant sur $\mathbb{F}_{q^k}$ et, incidemment, lorsque k est grand, l'arithmétique effective du corps $\mathbb{F}_{q^k}$ ne sera pas assez rapide pour permettre une évaluation pratique du couplage.

Gardons donc à l'esprit qu'un couplage ne peut s'évaluer en temps raisonnable que lorsque le degré d'immersion n'est pas trop grand.

Du point de vue algorithmique, le couplage de Weil est très intéressant car on sait l'évaluer en temps polynomial : étant donné deux points de ℓ-torsion, P et Q dans $\mathrm{Jac}(C)$ définis sur $\mathbb{F}_{q^k}$, l'algorithme de Miller calcule $e_\ell(P,Q)$ en temps polynomial en k, $\log(q)$, g et $\log(\ell)$. Aussi, aucun algorithme efficace n'est actuellement connu pour l'inversion du couplage de Weil (voir plus loin).

1.4 Algorithme de Miller

La difficulté relative à l'évaluation du couplage de Weil réside uniquement dans le calcul de la fonction f_P. Rappelons que c'est une fonction dont dérive le diviseur principal ℓP.

L'idée de MILLER est de construire cette fonction petit à petit, en s'aidant d'une suite de fonctions, (f_i), vérifiant la propriété suivante (O est un point à l'infini de la courbe).

$$\operatorname{div} f_i = iP - [i]P - (i-1)O$$

Bien évidemment, comme P est un point de ℓ-torsion, f_ℓ est la fonction recherchée. Aussi, f_1 est aisément calculable : on peut prendre tout simplement 1.

Ensuite, on peut construire récursivement les fonctions de cette suite de la façon suivante.

$$f_{i+j} := f_i f_j \frac{u}{v} \quad \text{où} \begin{cases} u \text{ est la droite } ([i]P, [j]P) \\ v \text{ est la droite } ([i+j]P, O) \end{cases}$$

On calcule donc aisément la fonction f_P et ainsi le couplage en utilisant cette construction dans un algorithme à la structure semblable à celle de l'exponentiation rapide.

Notons aussi que, dans cet algorithme, on peut, afin de réduire la quantité mémoire requise, évaluer la fonction construite en Q à chaque itération ; ainsi, l'algorithme donnera directement la valeur de $f_P(Q)$ — bien évidemment, il en va de même pour $f_Q(P)$. Bien sûr, il est nécessaire pour cela de traiter les bits de ℓ en partant du bit de poids fort et en allant au bit de poids faible.

2 Aspects cryptographiques

Pour les applications cryptographiques de la théorie des courbes elliptiques, on pourra se référer à [2, 3].

Le cas hyperelliptique est traîté dans [5].

2.1 Le problème du logarithme discret

La cryptographie asymétrique se base sur des problèmes dont la difficulté algorithmique n'est avérée que dans un sens. Le problème du logarithme discret est en quelque sorte *le* problème de référence en la matière, à la fois parce que son utilisation est très flexible mais aussi parce qu'il a été longuement étudié.

Il s'agit, étant donné un groupe fini G engendré par l'un de ses éléments, g, d'inverser la fonction exponentielle comme il suit ; la fonction obtenue, que le problème est donc de calculer, est notée $\log_g$.

$$\begin{array}{ccccccc} \mathbb{Z}/(\#G)\mathbb{Z} & \xrightarrow{\sim} & G & & G & \xrightarrow{\sim} & \mathbb{Z}/(\#G)\mathbb{Z} \\ m & \longmapsto & mg & = & h & \longmapsto & \log_g h \ (:= m) \end{array}$$

La première observation est que le problème du logarithme discret d'un groupe G est aussi difficile qu'il l'est dans ses sous-groupes d'ordre premier.

En effet, tout d'abord, en utilisant les restes chinois, on peut ramener ce problème dans des groupes dont les ordres sont des puissances de nombres premiers.

Ensuite, si l'on cherche à calculer $\log_g(h)$ dans un groupe d'ordre p^α, on commence par calculer $\log_{g^{p^{\alpha-1}}}\left(h^{p^{\alpha-1}}\right)$, ce pour quoi on peut se placer dans un sous-groupe d'ordre p, et cela donne $\log_g(h) \bmod p$. On peut alors itérer ce processus de façon à calculer $\log_g(h)$ modulo chacune des puissances de p jusqu'à α.

Lorsque le groupe G est générique, les meilleurs algorithmes pour calculer le logarithme discret sont des méthodes de recherche exhaustive — celle de Shanks, par collision, et la méthode ρ de Pollard qui améliore la quantité mémoire nécessaire. Ces méthodes sont de complexité exponentielle.

Bien évidemment, dans des cas tels que $G = \mathbb{Z}/n\mathbb{Z}$, il existe de biens meilleurs algorithmes qui exploitent la structure arithmétique que donne la description du groupe. Mais, dans le cas de $G = \mathbb{F}_q^\times$, l'absence d'algorithme trivial permettant de faire de même a fait de cette famille de groupes une solution de choix en cryptographie.

Pourtant, dans les années 1990, la mise au point de l'algorithme dit « de calcul d'index » a permis d'attaquer le problème du logarithme discret dans les groupes multiplicatifs des corps finis, et ce avec une complexité sous-exponentielle. Ainsi, à difficulté égale du calcul du logarithme discret, il faut un groupe de taille bien supérieure à celle qu'aurait un groupe générique.

Toutefois, pour les sous-groupes cycliques (d'ordre premier) des Jacobiennes de courbes hyperelliptiques de petit genre, les meilleurs algorithmes connus restent les algorithmes génériques mentionnés plus haut. C'est d'ailleurs le premier avantage quant à l'utilisation de courbes en cryptographie : le même niveau de sécurité est atteint avec des paramètres de taille nettement inférieure.

2.2 Les couplages : vision pessimiste

La première apparition des couplages en cryptographie est l'attaque MOV. Il s'agit d'exploiter la bilinéarité d'un couplage, et en particulier la relation suivante qui en découle, pour ramener le problème du logarithme discret de la Jacobienne d'une courbe dans le groupe multiplicatif d'un corps fini.

$$e_\ell(mP, Q) = e_\ell(P, Q)^m$$

De ce point de vue, le corps dans lequel ce problème est réduit, à savoir $\mathbb{F}_{q^k}^\times$, doit lui aussi être résistant aux attaques ; il faut donc qu'il soit suffisamment grand, ce qui se mesure par rapport à la complexité des meilleurs algorithmes de calcul du logarithme discret connus, dans ce cas sous-exponentielle.

Ainsi, pendant longtemps, on s'est contenté, pour obtenir une valeur de q^k suffisamment grande, de faire en sorte que les degrés d'immersion considérés soient très grands. En fait, il y a peu à faire pour cela ; en effet, les couplages avec de petits degrés d'immersion sont rares.

2.3 Les couplages : vision optimiste

Très récemment, on a eu l'idée d'intégrer les couplages à une nouvelle famille de protocoles cryptographiques. Les riches propriétés des couplages permettent en effet nombre d'applications très prometteuses en cryptographie asymétrique comme par exemple (pour n'en citer qu'une, la première historiquement) l'échange de clef tripartie en une étape mis au point par JOUX [11].

Incidemment, cela donne un second attrait à la cryptographie à base de courbes : les couplages permettent de faire des choses qu'on ne sait pas faire en cryptographie asymétrique classique (i.e. à base de corps finis).

Toutefois, pour que cela puisse être mis en pratique, il faut résoudre bon nombre de questions algorithmiques liées à l'existence, l'évaluation et l'inversion de couplages. Ce sont là les trois thèmes qui gouverneront la recherche menée dans cette thèse.

3 Construction de courbes adéquates

3.1 Enjeux et objectifs

La mise en pratique des systèmes cryptographiques à base de couplages requiert des courbes C ayant de bonnes propriétés : tout d'abord, le problème du logarithme discret doit être difficile dans la Jacobienne. Pour cela, le genre de C doit être petit et le cardinal de sa Jacobienne doit avoir un grand facteur premier (en l'état actuel des connaissances et des puissances de calcul, plus grand que 2^{160}).

Il faut ensuite que le degré d'immersion, k, soit adapté. En premier lieu, parce que, comme vu plus haut, un couplage permet à un attaquant de réduire le problème du logarithme discret de $\mathrm{Jac}(C)$ dans le groupe multiplicatif de $\mathbb{F}_{q^k}$, il faut que q^k soit suffisamment grand pour que ce problème soit algorithmiquement insoluble dans $\mathbb{F}_{q^k}$ — où nous disposons du calcul d'index — (en pratique, $q^k > 2^{1024}$).

Enfin, il ne faut pas que k soit trop grand non plus si l'on veut que le couplage puisse être calculé efficacement (la complexité de l'algorithme de Miller est linéaire en k). De telles courbes sont rares [1] et il faut donc mettre en oeuvre des techniques spéciales pour les construire. On qualifiera ces courbes de « bien couplées ».

3.2 Techniques à l'œuvre

Historiquement, les premiers exemples de courbes bien couplées sont des courbes supersingulières. Malheureusement, elles n'offrent qu'un nombre fini de choix pour le degré d'immersion. De plus, ayant une structure très particulière, ces courbes sont considérées comme potentiellement sujettes à d'éventuelles attaques exploitant cette particularité ; il est donc de bon ton de les éviter pour les applications cryptographiques.

Plusieurs procédés ont donc été mises au point pour construire des courbes ordinaires bien couplées.

La plupart de ces procédés concerne uniquement les courbes elliptiques [7] mais, récemment, d'autres ont été proposés pour les courbes de genre 2 [10, 6].

3.2.1 Jeux de paramètres

Tout d'abord, il faut déterminer les paramètres dont on a besoin de fixer les propriétés. Cela inclut les paramètres relatifs au sous-groupe d'ordre premier, à son couplage ainsi que des paramètres assurant qu'une courbe pourra être construite. En général, ces derniers paramètres proviennent de la théorie de la multiplication complexe (qui est utilisée — voir ci-dessous — pour générer l'équation explicite de la courbe) dans un corps quadratique imaginaire de petit nombre de classes.

Il faut dans un premier temps calculer un jeu de valeurs pour ces paramètres consistant — en le sens où une courbe réalisant ces valeurs existe. Ceci se fait en écrivant un système d'équations garantissant cette consistance et en calculant les paramètres de façon à satisfaire ce système.

3.2.2 Multiplication complexe

Dans un second temps, on utilise ces derniers paramètres pour construire l'équation de Weierstrass de la courbe — qui permet de calculer explicitement dans la Jacobienne de celle-ci.

Pour les courbes de genre 1, le discriminant suffit : à partir de ce dernier, on peut trouver la valeur de l'invariant j de la courbe elliptique et ainsi son équation explicite.

Pour les courbes de genre supérieure, il faut introduire d'autres invariants ; en effet, ce n'est plus de corps quadratiques qu'il s'agit mais d'extensions de degré 4 du corps des rationnels. Par exemple, il faut, en genre 2, trois invariants en lieu et place de l'unique invariant j du cas elliptique et la théorie qui entre en jeu est plus complexe.

C'est cette théorie qui fixe une partie des équations qui sont à considérer plus haut.

3.3 Cas elliptique

Par exemple, dans le cas des courbes elliptiques, on fixe comme paramètres :
- q, le cardinal du corps de base ;
- t, la trace de l'endomorphisme de Frobenius ;
- r, l'ordre du sous-groupe ;
- k, son degré d'immersion ;
- $-D$, le discriminant de l'ordre de multiplication complexe.

Dans ce jeu de valeurs entières, q et r se doivent d'être des premiers, t peut-être quelconque mais, par contre, D et k doivent être positifs.

La consistance de ces valeurs est assurée par les conditions suivantes :
- $r \mid q+1-t$ (il existe bien un sous-groupe d'ordre r) ;
- $r \mid \Phi_k(t-1)$ (k est bien de degré d'immersion) ;
- $(\exists y \in \mathbb{Z},)\ t^2 + Dy^2 = 4q$ (D est bien le discriminant de l'ordre CM).

Un théorème de WATERHOUSE [15] nous assure que, dès que $t \neq 0$ et $|t| \leqslant 2\sqrt{q}$ (ici, cela découle de la dernière équation), une courbe elliptique de trace t existe sur $\mathbb{F}_q$. Les équations ci-dessus s'assurent que cette courbe pourra bien avoir les autres paramètres prescrits.

Il existe diverses méthodes pour trouver des jeux de valeurs consistantes en ce sens.

Toutefois, comme le sous-groupe d'ordre r sera la seule partie « utile » de la courbe du point de vue cryptographique, on aimerait aussi maximiser sa taille relative telle qu'évaluée par la grandeur suivante.

$$\rho = \frac{\log q}{\log r}$$

Disons simplement qu'il n'est pas difficile de générer des courbes avec $\rho \approx 2$ et que l'on cherche à faire tendre ρ vers 1. De ce point de vue là, les méthodes les plus efficaces consistent à considérer que les paramètres en jeu sont non pas des entiers mais des polynômes (dont on instanciera la valeur a posteriori) : elles construisent directement une « famille » de courbe avec une valeur de asymptotique de ρ qui peut être très proche de 1.

Les cas elliptique et hyperelliptique semblent d'approche aussi délicate l'un que l'autre ; en effet, même si les courbes de genre 2 ont été, jusqu'ici, moins étudiées, leur compréhension nécessite des outils plus avancés (notamment, comme nous l'avons évoqué plus haut, pour ce qui concerne la multiplication complexe).

4 Évaluation efficace des couplages

Une fois une courbe adéquate construite, il faut être capable de calculer son couplage de Weil aussi rapidement que possible. Nous avons vu l'algorithme de Miller qui est une brique de base pour ces algorithmes et plusieurs travaux vont déjà dans le sens d'optimisations pour le calcul de ces couplages.

4.1 Couplages spécifiques

Une première façon de faire est de considérer un couplage qui n'est pas exactement celui de Weil mais qui en est dérivé et construit précisément de telle sorte qu'il soit plus aisé à calculer. Quoi qu'il en soit, du point de vue cryptographique, toute application bilinéaire non dégénérée convient. Ainsi, on peut être amené à considérer le couplage de Tate, le couplage Eta ou le couplage Ate [5].

Notons aussi que beaucoup de travaux tournent autour de l'optimisation de l'évaluation de « l'exponentiation finale » qui intervient notamment dans le calcul du couplage de Tate et qui a pour but de lui faire retourner un élément unique de chaque classe modulo les puissances ℓ-èmes.

$$e_\ell : \underbrace{\mathrm{Jac}_{\mathbb{F}_{q^k}}(C)[\ell] \times \mathrm{Jac}_{\mathbb{F}_{q^k}}(C)[\ell] \longrightarrow \overbrace{\mathbb{F}_{q^k} \overset{x \mapsto x^{\frac{q-1}{l}}}{\longrightarrow} \mu_\ell \subset \mathbb{F}_{q^k}}^{\text{exponentiation finale}}}_{\text{application bien définie}}$$

4.2 Optimisations spéciales

Une seconde approche est, étant donné un couplage quelconque, de précalculer des formules explicites pour son évaluation. Cela se fait de façons très similaire à celles qui sont employées pour optimise les formules de la loi d'addition des courbes elliptiques [8].

Enfin, on peut aussi optimiser le calcul d'un couplage particulier dans le cadre d'un protocole particulier. Notamment, pour une certaine classe d'applications, l'un des deux points du couplage en question est fixé pour toute une série d'évaluations et il est donc (dans certains cas) possible de précalculer les équations concernées intervenant dans les algorithmes pour cette valeur précise.

Beaucoup d'articles discutent de l'évaluation efficace des couplages. Toutefois, ce sujet reste très vaste et très actif. Des avancées sont toujours à venir, notamment dans le domaine des implémentations matérielles pour lesquelles les algorithmes doivent être remaniés (car l'efficacité n'est alors plus mesurée par la seuls complexité de l'algorithme) et adaptés aux supports qui sont concernés.

5 L'inversion du couplage

Comme mentionné ci-dessus, la sécurité d'un système cryptographique basé sur un couplage dépend non seulement de la difficulté du problème du logarithme discret dans la Jacobienne ainsi que dans le corps d'immersion mais aussi de la difficulté d'inverser le couplage d'un point de vue algorithmique.

5.1 Définition du problème

Il s'agit, étant donné un point et une racine de l'unité, de trouver un second point dont le couple avec le premier donne comme image par le couplage la racine voulue, soit de calculer l'inverse de la fonction suivante.

$$e_\ell^P : Q \mapsto e_\ell(P, Q)$$

Le problème du logarithme discret a fait l'objet d'intenses recherches mais ce n'est pas encore le cas du problème d'inversion du couplage qui semble plus difficile d'approche [9].

5.2 Liens avec des problèmes plus étudiés

Un certain nombre d'autres problèmes difficiles peuvent être considérés comme, notamment, le problème de Diffie-Hellman (que ce soit dans sa version algorithmique ou décisive). Les relations entre ces différents problèmes sont assez délicates à établir et il reste des choses à faire à ce sujet.

L'existence d'un algorithme efficace pour l'inversion du couplage aurait des retombées au delà de la cryptographie à base de couplages. Notamment, cela impliquerait la résolution du problème de Diffie-Hellman algorithmique dans certains corps finis ce qui serait une avancée majeure. Ainsi, même s'il semble irréaliste de trouver un algorithme polynomial pour résoudre ce problème, son étude est vitale.

Références

[1] R. BALASUBRAMANIAN and Neal KOBLITZ. The improbability that an elliptic curve has subexponential discrete log problem under the Menezes-Okamoto-Vanstone algorithm. *Journal of Cryptology*, 11(2) :141–145, 1998.

[2] Ian F. BLAKE, Gadiel SEROUSSI, and Nigel P. SMART. *Elliptic curves in cryptography*. Cambridge University Press, 1999.

[3] Ian F. BLAKE, Gadiel SEROUSSI, Nigel P. SMART, and John W. S. CASSELS. *Advances in elliptic curve cryptography*. Cambridge University Press, 2005.

[4] Dan BONEH and Matt FRANKLIN. Identity-based encryption from the Weil pairing. *SIAM Journal of Computing*, 32(3) :586–615, 2003.

[5] Henri COHEN and Gerhard FREY. *Handbook of elliptic and hyperelliptic curve cryptography*. CRC Press, 2005.

[6] David FREEMAN. Constructing pairing-friendly genus 2 curves with ordinary jacobians. Cryptology ePrint Archive, Report 2007/057, 2007.

[7] David FREEMAN, Michael SCOTT, and Edlyn TESKE. A taxonomy of pairing-friendly elliptic curves. Cryptology ePrint Archive, Report 2006/372, 2006.

[8] Gerhard FREY and Tanja LANGE. Fast bilinear maps from the Tate-Lichtenbaum pairing on hyperelliptic curves. In *Proceedings of the 7th International Symposium on Algorithmic Number Theory*, volume 4076, pages 466–479. Springer LNCS, 2006.

[9] Steven D. GALBRAITH, Florian HESS, and Frederik VERCAUTEREN. Aspects of pairing inversion. Cryptology ePrint Archive, Report 2007/256, 2007.

[10] Laura HITT. Families of genus 2 curves with small embedding degree. Cryptology ePrint Archive, Report 2007/001, 2007.

[11] Antoine JOUX. A one round protocol for tripartite Diffie-Hellman. In *Proceedings of the 4th International Symposium on Algorithmic Number Theory*, volume 1838, pages 385–394. Springer LNCS, 2000.

[12] Alfred MENEZES, Tatsuaki OKAMOTO, and Scott VANSTONE. Reducing elliptic curve logarithms in a finite field. *IEEE Transactions on Information Theory*, 39(5) :1639–1646, 1993.

[13] Ryuichi SAKAI, Kiyoshi OHGISHI, and Masao KASAHARA. Cryptosystems based on pairing. In *Proceedings of the Symposium on Cryptography and Information Security*, page ref. C20, 2000.

[14] Joseph SILVERMAN. *The arithmetic of elliptic curves*. Springer, 1985.

[15] William C. WATERHOUSE. Abelian varieties over finite fields. *Annales scientifiques de l'École Normale Supérieure*, 2(4) :521–560, 1969.