Calcul de racines de polynômes dans un corps de nombres

Andrea Lesavourey

Contexte Dans la recherche actuelle de primitives pouvant résister à l'utilisation d'un ordinateur quantique, une des pistes majeure se base sur les réseaux euclidiens, et en particulier sur le problème Learning With Errors (LWE). En effet, il existe une réduction pire cas – moyen cas vers le problème classique de réseaux qu'est le Shortest Vector Problem (SVP). Pour des raisons d'efficacité, les schémas envisagés se basent sur des versions structurées de LWE, comme Ring ou Module-LWE. Il existe par ailleurs des réductions pire cas – moyen cas de ces problèmes vers le SVP restreint respectivement aux réseaux idéaux (Ideal-SVP) et modules (Module-SVP). C'est pourquoi l'analyse de Ideal-SVP a reçu une attention soutenue ces dernières années [4, 2, 7, 3]. Dans cette optique, il est important de pouvoir faire des calculs *en pratique* pour ne pas se contenter d'arguments asymptotiques, notamment sur dans des corps de grande dimension. Les algorithmes connus pour résoudre Ideal-SVP nécessitent de calculer des groupes de S-unités du corps de nombres considéré. Une étape courante pour calculer ces objets est une étape de *saturation*, qui permet d'agrandir un sous-groupe de rang plein en y adjoignant des racines p-èmes d'éléments, pour p un nombre premier. Cette étape est notamment évoquée dans des travaux récents s'attachant à étudier en pratique la possibilité de retrouver des éléments courts de réseaux idéaux [1, 6, 3]. Cette étape nécessite donc le calcul des racines de ces éléments, qui est un cas particulier du calcul de racines polynomiales. Par conséquent, améliorer les performances des algorithmes permettant de retrouver les racines de polynômes à coefficients dans un corps de nombres peut avoir un impact important sur notre capacité à étudier Ideal-SVP en pratique.

Contribution Ma contribution est de proposer un algorithme efficace permettant de calculer les racines de polynômes dans des corps de nombres, en se ramenant notamment à des décodages dans un sous-corps (s'il en existe un).

Il est connu depuis le papier fondateur introduisant l'algorithme LLL [5] qu'il est possible de retrouver les coefficients d'un élément x connaissant l'approximation d'un de ses plongements complexes $\sigma(x)$. En effet, étant donné L un corps de nombres, soient n son degré, σ un de ses plongements complexes et $B = (b_1, \dots, b_n)$ une base de $L/\mathbb{Q}$, fixons $\mathcal{B}_l = (v_i)_i$ la base de $\mathbb{R}^{n+1}$ telle que $v_i = -\sigma(b_i)e_1 + Ce_{i+1}$, où $(e_i)_i$ est la base canonique de $\mathbb{Z}^{n+1}$, $C > 0$ est une constante et les plongments sont calculés avec précision $l \in \mathbb{N}$. Alors pour tout $g \in \mathbb{Z}[B]$, résoudre un BDD sur la cible $(\sigma(g), 0, \dots, 0)$ par rapport à la base LLL-réduite correspondant au réseau engendré par $\mathcal{B}_l$ renvoie un vecteur $(g_1, \dots, g_n)$ tel $g = \sum_{i=1}^{n} g_i b_i$ (si l est suffisamment grand). Ainsi, étant donné un polynôme $f(X) \in L[X]$, il est possible de calculer ses racines en le plongeant dans $\mathbb{C}$.

Je propose une extension de cette approche dans le cas où L admet un sous-corps K. Etant donné un plongement complexe τ de K, il est possible d'utiliser les plongements relatifs de L/K qui étendent τ afin de de se ramener au décodage d'éléments de K au lieu de L. Ceci est au prix d'une recherche dans un ensemble de taille $\deg f(X)^{[L:\mathbb{Q}]}$, mais nécessite de réduire une matrice de taille $[K : \mathbb{Q}] \times ([K : \mathbb{Q}] + 1)$ au lieu de $[L : \mathbb{Q}] \times ([L : \mathbb{Q}] + 1)$, ce qui permet d'accélérer les calculs dans un certain nombres de cas. C'est notamment vrai lorsque $[L : K]$ est petit, par exemple dans les corps cyclotomiques qui contiennent toujours un sous-corps K de degré $[L : \mathbb{Q}]/2$. Il est également possible d'utiliser des observations heuristiques afin d'accélérer les temps de calculs, en mettant notamment en place une méthode de rejet rapide dans la phase de recherche.

Références

[1] J. Bauch et al. "Short Generators Without Quantum Computers : The Case of Multiquadratics". In : *Advances in Cryptology – EUROCRYPT 2017*. Sous la dir. de J.-S. Coron et al. Cham : Springer International Publishing, 2017, p. 27-59. isbn : 978-3-319-56620-7.

[2] O. Bernard et al. *Twisted-PHS : Using the Product Formula to Solve Approx-SVP in Ideal Lattices*. Cryptology ePrint Archive, Report 2020/1081. `https://eprint.iacr.org/2020/1081`. 2020.

[3] O. Bernard et al. *Log-S-unit lattices using Explicit Stickelberger Generators to solve Approx Ideal-SVP*. Cryptology ePrint Archive, Report 2021/1384. `https://ia.cr/2021/1384`. 2021.

[4] R. Cramer et al. "Short Stickelberger Class Relations and Application to Ideal-SVP". In : *EUROCRYPT*. 2017.

[5] A. Lenstra et al. "Factoring Polynomials with Rational Coefficients". In : *Mathematische Annalen* 261 (déc. 1982).

[6] A. Lesavourey et al. "Short Principal Ideal Problem in multicubic fields". In : *Journal of Mathematical Cryptology* 14.1 (1Jan. 2020), p. 359-392.

[7] A. Pellet-Mary et al. "Approx-SVP in Ideal Lattices with Pre-processing". In : *Advances in Cryptology – EUROCRYPT 2019*. Sous la dir. d'Y. Ishai et al. Cham : Springer International Publishing, 2019, p. 685-716. isbn : 978-3-030-17656-3.