Explication du code

March 20, 2024

1 Calcul du réseau de relations

Je l'ai fait en C à l'aide de FLINT. Je n'étais pas conscient du plongement dans le tore. J'utilise l'algorithme de Pohlig-Hellman couplé à un rho-pollard basique. L'implémentation est faite à l'aide du module qfb de FLINT qui permet de calculer dans le groupe de classe de formes quadratiques binaires de discriminant donné.

2 Paramètres

Les candidats sont calculés à l'aide de gen_conductor_choices dans lib/candidate_conductors. Chaque fichier candidate_conductorsN contient les candidats pour les N premiers nombres premiers décomposés dans $\mathbb{Z}[i]$. L'évaluation de chaque candidats est éffectué par eval_candidate présent dans lib/eval_candidate à l'aide de ECM avec abandon et plusieurs processus en parallèle qui sont terminés au bout d'un temps donné si ils n'ont pas finis.

2.1 Des paramètres

Les conducteurs, leurs factorisations et d'autres données sont compilées pour la lecture dans:

lib/eval_candidates/conductor_dataN.

Le dossier txt regroupe les données utilisées par le code.

2.2 Paramètres trouvés

Pour $n_1 + n_2 = 3$ il y'a $\alpha = 109i - 482$ ou $f - 1 = 2 * 2 * 3 * 3 * 3$ (pour un exemple jouet).
Pour $n_1 + n_2 = 14$ j'ai trouvé

$$\alpha = 18359253140637317346421 * i + 51954880756346626090702$$

avec $f \approx 2^{74}$ et $f - (\frac{-1}{f})$ qui est 2^{15}-lisse et $L(f, 1/2) \approx 2 * *20$! J'ai pris

$$p = 44357287260000669680627459$$

$$= cL - 1$$

avec $c = 36$. Pour $n_1 + n_2 = 19$ j'ai trouvé

$$\alpha = 37874631831601553001516281906516 99i$$

$$+ 40291066555757539338137792453288 98$$

avec $f \approx 2^{111}$ et $f - (\frac{-1}{f})$ qui est 2^{24}-lisse. J'ai aussi pris

$$p = 14837983697313767791424137522458 6059$$

$$= cL - 1$$

avec $c = 12$.

La table des logs discrets est txt/dlogs_N_primes.md.

3 Commandes

Pour lancer le calcul du réseau de relation, depuis le dossier Clib:

- make
- ./bin/lattice_relations ../txt/conductor_N_primes.md ../txt/sqrts_N_primes.md ../txt/dlogs_N_primes.md

Actuellement, seulement $N = 3, 14, 19, 20$ sont utilisables (il y'a d'autres bons candidats pour $N = 26, 27$)

Depuis le dossier lib:

- candidate_conductors et eval_candidate contiennent les fonctions permettant de générer et évaluer les conducteurs

Pour les autres, chaque fonction appelée avec $-h$ pour afficher les options. La plupart on $-v$ pour afficher les calculs faits et $-n$ pour faire les calculs sur le conducteur choisi pour n nombres premiers. Parmis ces fonctions:

- gen_curve et endo permettent respectivement de générer la courbe et de calculer un endomorphisme de norme $M > p$.
- gen_prime génère un premier associé à un conducteur