Introduction

Notations

$f \in L(\mathbb{F}^k, \mathbb{F}^n)$	avec $k < n$
$G = Mat(f)$	G matrice génératrice $n \times k$
$Ker(H) = Im(f)$	H matrice de parité $k \times n$
$S_y(X) = 0 \Leftrightarrow y \in Im(f)$	$S_y(X)$ syndrome
$d : y \in \mathbb{F}^n \rightarrow Card(i/y_i \neq 0)$	d poids de Hamming

Définition

- g polynôme de $\mathbb{F}_{2^m}[X]$ irréductible de degré t.
- $L = (\alpha_1, .., \alpha_n) \in \mathbb{F}_{2^m}^n$ le support

On définit un code de Goppa par son syndrome

$$\mathbf{S}_y(x) = \sum_{i=1}^{n} \frac{y_i}{x - \alpha_i} \ mod \ g(x)$$

- En pratique on prendra $n = 2^m$.
- Capacité de correction $\geq \frac{t}{2}$

Décodage

- $\sigma_y(X)$ polynôme localisateur d'erreurs
- $z = y + \epsilon \in \mathbb{F}^n$ avec y mot de code
- $L = (\alpha_1, .., \alpha n)$ le support

$$\sigma_z(X) = \prod (X - \alpha_i)^{\epsilon_i}$$

Soit l'équation clef suivante :

$$\omega_z(X) = S_z(X)\sigma_z(X) \ mod \ g$$

- On montre que $deg(\omega_z(X) < \frac{t}{2})$
- Unicité du couple solution avec les contraintes de degrés
- Existence grâce à l'algorithme d'Euclide étendu

Clefs

Clef privée

- G matrice génératrice, ainsi que L et g
- P matrice de permutation $n \times n$
- Q $\in GL_k(\mathbb{F})$

Clef publique

- G' = PGQ
- capacité de correction

Principe

Soit $x \in \mathbb{F}^k$ le message à envoyer de Alice vers Bob

- Bob partage $G' = PGQ$ et une capacité de correction
- Alice envoie $y = PGQx + \epsilon$
- Bob calcule $P^{-1}y = GQx + P^{-1}\epsilon$
- Bob corrige l'erreur et obtient Qx
- Bob obtient x en connaissant Q inversible

Sécurité par 2 problèmes de théorie des codes :

- Indistinguabilité d'un code de Goppa
- Décodage par syndrome NP-complet

Programmation Orientée objet

Rédaction de classes :

- Corps de Galois $\mathbb{F}_{p^m}$
- Optimisation en binaire pour $p = 2$
- Matrices
- Polynômes

Intérêts : Pédagogique, flexibilité de Python.
Inconvénients : Trop lent, peu optimisé par rapport aux bibliothèques standards.

Algorithme de Berlekamp-Hensel

Teste l'irréductibilité de $g \in \mathbb{F}_{2^m}[X]$

$$\exists P / 1 < deg(P) < deg(g) \text{ et } P^{2^m} - P = 0 \ mod \ g \Leftrightarrow g \text{ reductible}$$

Par le morphisme de Frobenius, on regarde le noyau de l'application linéaire :

$$P \leftarrow P^{2^m} - P$$

Bilan

Programme fonctionnel mais lent :

- Paramètres conseillés $(n, k, t) = (1024, 524, 50)$
- 300s pour générer les clefs de paramètres $(256, 184, 9)$

Possibilités d'optimisation :

- Usage de Sage pour le calcul matriciel : 100s pour $(1024, 524, 20)$