Décodage en liste avec la librairie DECODING

Guillaume Quintin
Équipe Grace,
INRIA SACLAY–Île-de-France,
Laboratoire d'informatique de l'X (LIX),
École polytechnique

10 octobre 2012

- Ceci représente une partie de mon travail de thèse encadrée par Daniel Augot et Grégoire Lecerf.

Plan :

1. Codes quasi cycliques et codes sur les anneaux.
2. Précisions sur les codes de Reed-Solomon.
3. La librairie DECODING pour le décodage en liste.
4. Démonstration logicielle.

Les codes quasi cycliques

E un ensemble et $\mathcal{C} \subseteq E^n$, et $n = m\ell$.
On dit que $\mathcal{C}$ est ℓ-quasi cyclique si

$$(x_1, \ldots, x_\ell, \ldots, x_{\ell+1}, \ldots, x_{2\ell}, x_{n-\ell+1}, \ldots, x_n) \in \mathcal{C}$$
$$\Rightarrow (x_{n-\ell+1}, \ldots, x_n, x_1, \ldots, x_\ell, \ldots, x_{\ell+1}, \ldots, x_{2\ell}) \in \mathcal{C}.$$

- Structure :
  - Lally et Fitzpatrick [LF01],
  - Ling et Solé [LS01] et
  - Cayrel, Chabot et Necer [CCN10].
- Application McEliece :
  - Berger, Cayrel, Gaborit et Otmani [BCGO09].

Résultats (Barbier, Chabot, Quintin [BCQ12b])

Les codes ℓ-quasi cycliques sont en correspondance bijective avec les idéaux à gauche de $M_{\ell\times\ell}(\mathbb{F}_q)[X]/(X^m - 1)$.

Les codes quasi BCH

Définition (Barbier, Chabot, Quintin [BCQ12b])

Soit $A \in M_{\ell\times\ell}(\mathbb{F}_q^s)$ une racine primitive m-ième de l'unité.
On définit un code quasi BCH de distance construite δ par

$$BCH := \left\{ (c_1, \ldots, c_m) \in \left(\mathbb{F}_q^\ell\right)^m : \sum_{j=0}^{m-1} (A^i)^j c_{j+1} = 0 \text{ pour } i = 1, \ldots, \delta - 1 \right\}$$

Codes BCH	"subfield subcode"	Codes RS.
Codes quasi-BCH	"subfield submodule"	Codes RS sur $M_{\ell\times\ell}(\mathbb{F}_q)$.

Espoir de trouver une algorithme de décodage.

Codes de Reed-Solomon sur un anneau A

- On fixe un anneau (fini) A.
- $A^\times$ désigne le groupe des unités de l'anneau A.
- On choisit des entiers $0 < k < n \leq q$.
- On choisit un support : un vecteur de A^n.

$$(x_1, \ldots, x_n) \in A^n \text{ tel que } (x_i - x_j) \in A^\times \text{ et } x_i x_j = x_j x_i.$$

- On définit un code de Reed-Solomon comme le sous module à gauche des

$$(f(x_1), \ldots, f(x_n)) \text{ pour } f \in A[X] \text{ et } \deg f < k.$$

Résultats (Barbier, Chabot, Quintin 2012 [BCQ12a])

Ses paramètres sont $[n, k, n-k+1]_A$. Les algorithmes de Welch-Berlekamp et Guruswami-Sudan restent valides.

Rappel sur l'algorithme de Guruswami-Sudan

- Les mots de code d'un code de Reed-Solomon sont en bijection avec les polynômes de $A[X]$ de degré au plus $k-1$.
- Rechercher un mot de code, c'est donc rechercher un polynôme.

Entrée : un mot reçu $y = (y_1, \ldots, y_n) \in A^n$.
Sortie : les mots de code à distance (de Hamming) au plus $n - \sqrt{n(k-1)}$ de y.

1. Chercher une courbe $Q(X, Y)$ passant par tous les points (x_i, y_i) avec, au moins, une certaine multiplicité et de degré bornée.
2. Trouver les A-paramétrisations de la forme $Y = f(X)$ de $Q(X, Y)$, où f est une de $A[[X]]$.
3. Retourner les $f(X)$ tels que $Q(X, f(X)) = 0$.

Une autre motivation pour utiliser des anneaux (1)

Deux autres motivations pour les codes de Reed-Solomon sur des anneaux (finis) quelconques.

- Sur un quotient d'un anneau de valuation discrète.
- Algorithmes de décodage en liste peu étudiés.
- Sur un Anneau fini commutatif, comme $\mathrm{GR}(p^s, r)$ et $\mathbb{F}_q[[t]]/(t^r)$:
  - Armand [Arm04, Arm05b, Arm05a, AdT05].

Résultats (Berthomieu, Lecerf, Quintin 2011 [BLQ11])

Il existe un algorithme en temps polynomial pour l'étape de recherche de racines dans Guruswami-Sudan.

Tous les algorithmes de [BLQ11] ont été implantés dans Mathemagix [H$^+$02].

Une autre motivation pour utiliser des anneaux (2)

Codes entrelacées étudié notamment par

- Bleichenbacher, Kiayias et Yung [BKY03].
- Coppersmith et Sudan [CS03].
- Gopalan, Guruswami et Raghavendra [GGR11].

Résultats (Quintin 2012 [Qui12])

Soit un code $\mathcal{C}'$ linéaire sur $\mathbb{F}_q$ et un code entrelacé $\mathcal{C}$ par rapport à $\mathcal{C}'$ de degré r. Alors il existe un code $\mathcal{C}''$ sur $\mathbb{F}_q[[t]]/(t^r)$ tel que corriger les erreurs pour $\mathcal{C}$ revient exactement à corriger les erreurs pour $\mathcal{C}''$ pour un certain modèle d'erreurs.

La librairie DECODING

`http://www.lix.polytechnique.fr/~quintin/decoding/`

Pourquoi ?

- Besoin de généricité car besoin de plusieurs types d'anneaux.
  - Généricité non proposée par les librarires existantes en C.
  - Plus compliqué en C++ (template non adaptés).
- Aucune implantation existante sauf
  - GUAVA [CRB+13] un paquet de GAP [GAP12], seulement l'algorithme de Sudan et
  - Percy++, "Private Information Retrieval".

- Écrite en C, sous licence GPL.
- Utilise les performances de GMP [Gra91] et MPFQ [GT06].
- Permet d'écrire avec une certaine généricité les algorithmes.

Les fonctions disponibles

Les corps finis suivants :

- $\mathbb{F}_{2^s}$ pour $2 \leq s \leq 255$ (MPFQ).
- $\mathbb{F}_p$ pour p rentrant sur un mot machine (GMP).

Les algorithmes de décodage suivants :

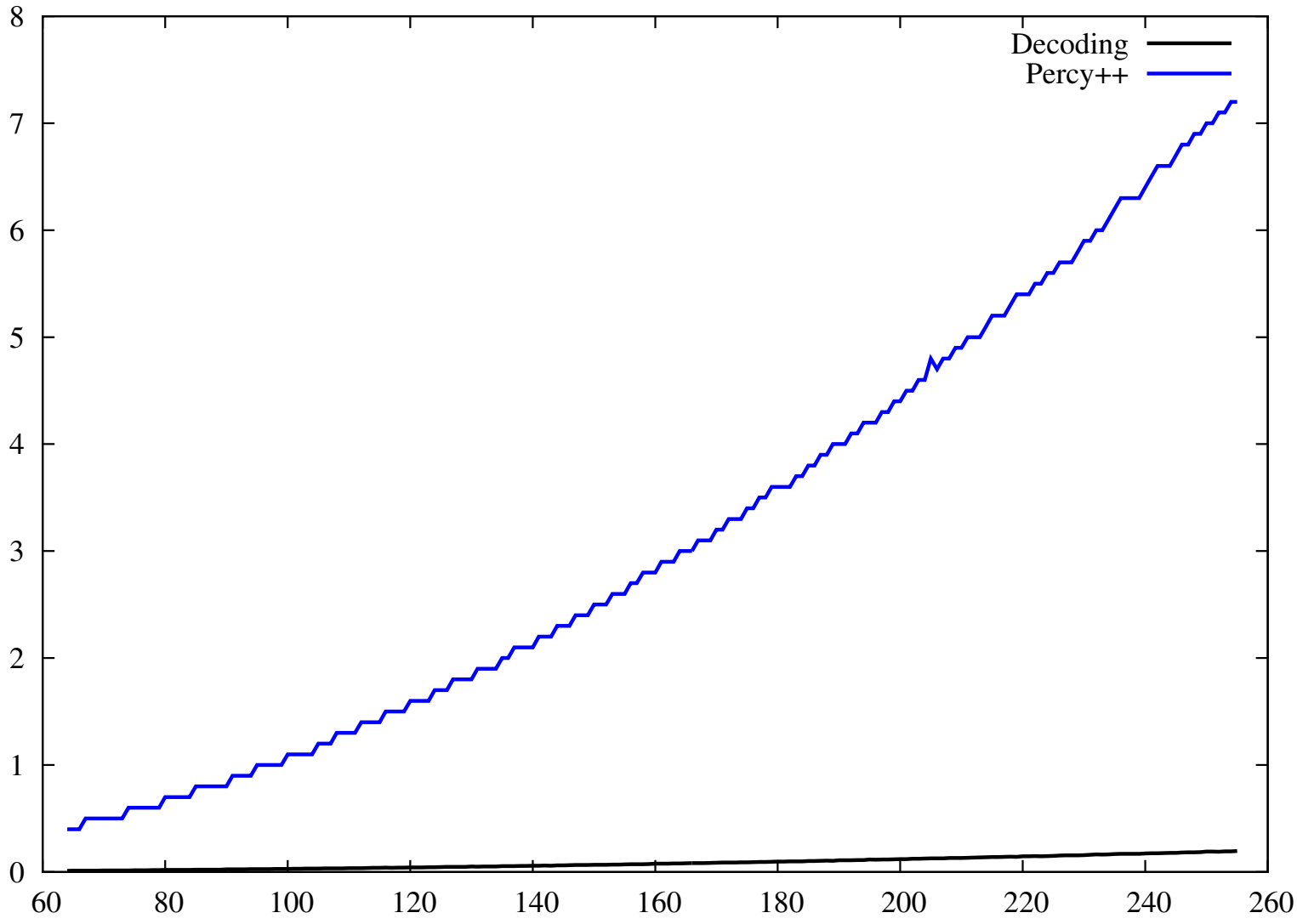
- Sudan (variante spéciale de Guruswami-Sudan).
- Guruswami-Sudan.
- Guruswami-Sudan avec multiplicités différentes (soft decoding).
- Berlekamp-Massey (Morgan Barbier) orphelin pour l'instant.

Les temps de calculs

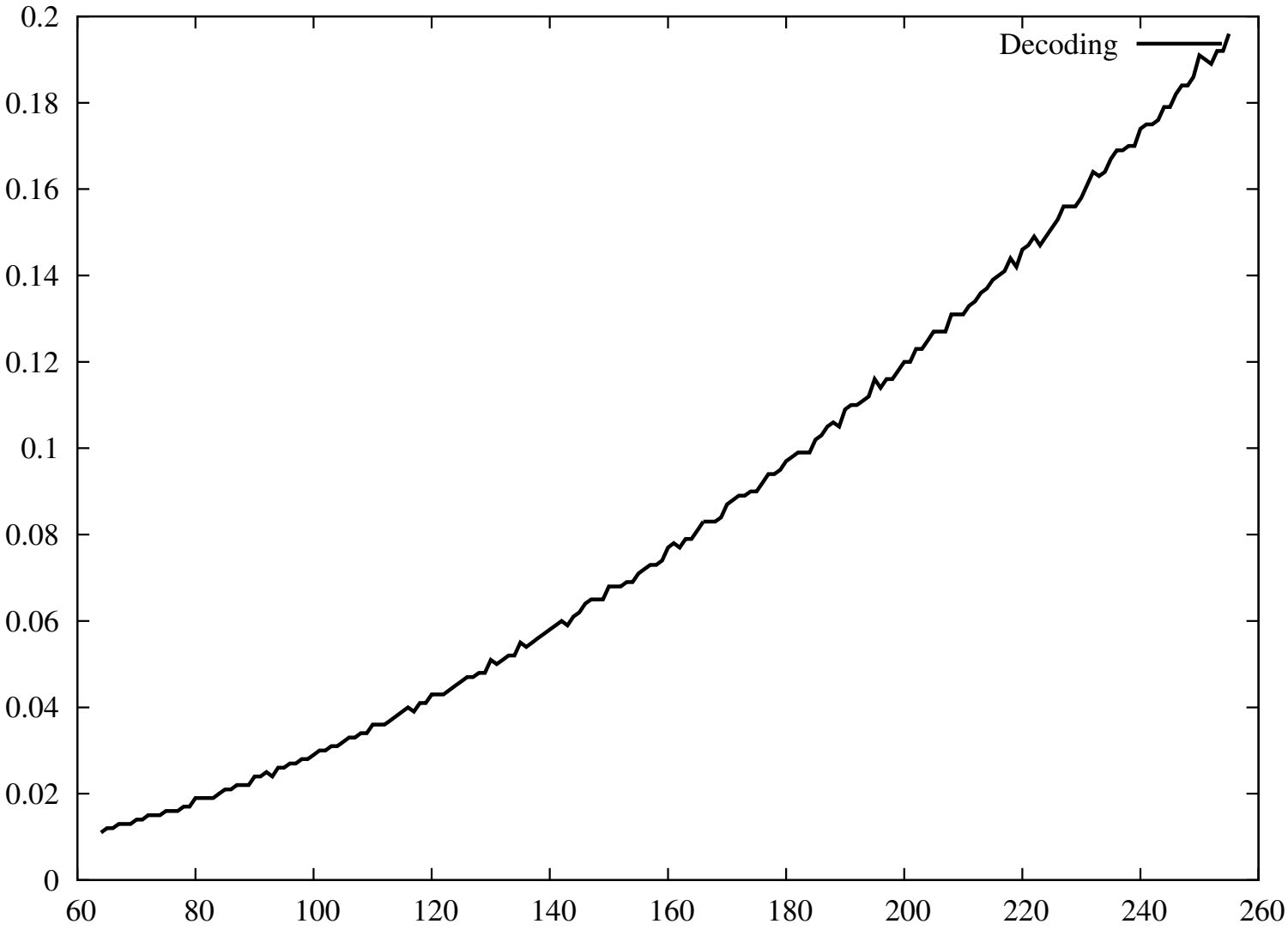
- **Corps de base :** $\mathbb{F}_{256}$.
- **Abcisse :** longueur du code.
- **Ordonnée :** temps en seconde.

- Intel(R) Core(TM)2 CPU 6400 @ 2.13GHz
- 4 Go RAM
- gcc 4.4.6
- GMP 5.0.5
- MPFQ 1.0-rc3
- Percy++ 0.8

Les temps de Decoding et Percy++



Les temps de Decoding



Démonstration.

Merci beaucoup pour votre attention ! ! !

References I


M. A. Armand and O. de Taisne.
Multistage list decoding of generalized Reed-Solomon codes over Galois rings.
Communications Letters, IEEE, 9(7) :625–627, jul 2005.


M. A. Armand.
Improved list decoding of generalized Reed-Solomon and alternant codes over rings.
In *IEEE International Symposium on Information Theory 2004 (ISIT 2004)*, page 384, 2004.


M. A. Armand.
Improved list decoding of generalized Reed-Solomon and alternant codes over Galois rings.
IEEE Trans. Inform. Theory, 51(2) :728–733, feb 2005.

References II


M. A. Armand.
List decoding of generalized Reed-Solomon codes over commutative rings.
IEEE Trans. Inform. Theory, 51(1) :411–419, 2005.


T. Berger, P.-L. Cayrel, P. Gaborit, and A. Otmani.
Reducing Key Length of the McEliece Cryptosystem.
In *Proceedings of the 2nd International Conference on Cryptology in Africa : Progress in Cryptology*, AFRICACRYPT '09, pages 77–97, Berlin, Heidelberg, 2009. Springer-Verlag.


M. Barbier, C. Chabot, and G. Quintin.
On Generalized Reed-Solomon Codes Over Commutative and Noncommutative Rings, 2012.


M. Barbier, C. Chabot, and G. Quintin.
On quasi-cyclic codes as a generalization of cyclic codes.
Finite Fields and Their Applications, 18(5) :904–919, 2012.

References III


D. Bleichenbacher, A. Kiayias, and M. Yung.
Decoding of Interleaved Reed Solomon Codes over Noisy Data.
In Jos Baeten, Jan Lenstra, Joachim Parrow, and Gerhard Woeginger, editors, *Automata, Languages and Programming*, volume 2719 of *Lecture Notes in Computer Science*, pages 188–188. Springer Berlin / Heidelberg, 2003.


J. Berthomieu, G. Lecerf, and G. Quintin.
Polynomial root finding over local rings and application to error correcting codes.
`http://hal.inria.fr/hal-00642075`, 2011.


P.-L. Cayrel, C. Chabot, and A. Necer.
Quasi-cyclic codes as codes over rings of matrices.
Finite Fields and Their Applications, 16(2) :100–115, 2010.

References IV


J. Cramwinckel, E. Roijackers, R. Baart, E. Minkes, L. Ruscio, R. Miller, T. Boothby, J. Fields, C. Tjhai, and D. Joyner.
GUAVA.
`http://opensourcemath.org/guava/`, 2013.


D. Coppersmith and M. Sudan.
Reconstructing curves in three (and higher) dimensional space from noisy data.
In *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*, STOC '03, pages 136–142, New York, NY, USA, 2003. ACM.


The GAP Group.
GAP – Groups, Algorithms, and Programming, Version 4.5.6, 2012.

References V


P. Gopalan, V. Guruswami, and P. Raghavendra.
List Decoding Tensor Products and Interleaved Codes.
SIAM Journal of Computing, 40(5) :1432–1462, 2011.


T. Granlund.
The GNU Multiple Precision Arithmetic Library, 1991.
`http://gmplib.org/`.


P. Gaudry and E. Thomé.
MPFQ : Fast Finite fields, 2006.
`http://mpfq.gforge.inria.fr/`.


J. van der Hoeven et al.
Mathemagix.
Software available from `http://www.mathemagix.org`, 2002.


K. Lally and P. Fitzpatrick.
Algebraic structure of quasicyclic codes.
Discrete Applied Mathematics, 111(1–2) :157–175, 2001.

References VI


S. Ling and P. Solé.
On the algebraic structure of quasi-cyclic codes .I. Finite fields.

IEEE Trans. Inform. Theory, 47(7) :2751–2760, nov 2001.


G. Quintin.
A lifting decoding scheme and its application to interleaved linear codes.
In *Information Theory Proceedings (ISIT), 2012 IEEE International Symposium on*, pages 96–100, july 2012.