The decoding Library for List Decoding

Guillaume Quintin

The DECODING Library for List Decoding

Guillaume Quintin

May 22, 2012

1 Introduction and motivation

The DECODING library is a C library whose main goal is to implement as efficiently as possible the Guruswami-Sudan algorithm [GS98]. It is written in C89 and is stand-alone.

Reed-Solomon codes (denoted by RS codes) form an important and well-studied family of codes. They were first proposed in 1960 by Irvin Stoy Reed and Gustave Solomon in their original paper [RS60]. They are widely used in practice such as in compact disc players, disk drives, satellite communications, and high-speed modems such as ADSL. See [WB99] for details about applications of RS codes.

RS codes can be efficiently unique decoded. See for example [Gao02] and [Jus76]. A breakthrough has been made by Madhu Sudan in 1997 about the list decoding of RS codes in his paper [Sud97], further improved by Venkatesan Guruswami and Madhu Sudan in [GS98].

2 The implementation

To the knowledge of the author no implementation of the Guruswami-Sudan algorithm has been proposed since the apparition of the Sudan algorithm in 1997. The only available implementation is constituted by a set of C++ functions, not directly available, inside the PERCY++ library [Gol07] whose purpose is not error correction algorithms and which does not use fast algorithms for dense bivariate polynomials.

2.1 The algorithms provided by decoding

The DECODING library is devoted to algorithms concerning the Guruswami-Sudan decoding scheme. The Guruswami-Sudan scheme has two steps. The first step is called the "interpolation" step which consists in finding a curve of equation $Q(X, Y) = 0$ in $\mathbb{A}^2_{\mathbb{F}_q}$ which passes through given points with given multiplicities. The second step finds the roots of $Q(X, Y)$ seen in $(\mathbb{F}_q[X])[Y]$.

The implemented algorithm for interpolation is a variant of the Koetter algorithm [McE03] available in the `algos/koetter-in.c` file. It uses polynomial

arithmetic with a fast bivariate shifting (computation of $Q(X + x_0, Y + y_0)$ where $(x_0, y_0) \in \mathbb{F}_q^2$) algorithm available in `algos/dbpol_shift_fast-in.c` and a fast univariate shifting (computation of $f(X + x_0)$ where $f \in \mathbb{F}_q[X]$ and $x_0 \in \mathbb{F}_q$) algorithm available in `algos/upol_shift_fast.c`. Special variants of these algorithms for rings of characteristic 2 are also proposed in the same files.

The second (root-finding) step is a variant of the Roth and Ruckenstein algorithm [RR98] and the naive algorithm of [BLQ11] available in `algos/dbpol_Xroots-in.c`. It also uses fast bivariate polynomials arithmetic provided by `algos/dbpol_replace_X_by_XY-in.c` and `algos/dbpol_shift_in_X-in.c`.

2.2 The design of decoding

Although DECODING proposes certain fast bivariate polynomials algorithms, it is not its goal to propose fast algorithms for univariate polynomials and bivariate polynomials multiplication. In fact, DECODING is designed to be used in conjonction with other efficient libraries like GMP, NTL or FLINT for exemple. For the sake of completeness DECODING provides these algorithms in their "schoolbook" form but it is recommended, for efficiency, to use external libraries.

The DECODING library is designed to be easy to use in a C or C++ program. One of its particularities is to use the C preprocessor to generate algorithms for a ring (generally a finite field) which must be provided by the end-user. Hence, efficient libraries like MPFQ can be used. Again, for the sake of completeness, some finite fields are provided by default. This flexibility is needed when studying codes over Galois rings for exemple where the end-user needs to manipulate bivariate and univariate polynomials over a given Galois ring and its residue field.

3 Presentation

The DECODING library is the first library which proposes a flexible and efficient way to implement algorithms related to the Guruswami-Sudan decoding scheme. It can be used with efficient external libraries to obtain more efficient decoding algorithms for the implementation part of the Guruswami-Sudan related algorithms.

I will first present the very short history of Guruswami-Sudan algorithms to show that it needs dense bivariate polynomials only available, not necessarily directly, in computer algebra systems such as MAGMA or MATHEMAGIX. As error correcting codes are often used over binary fields, dedicated fast algorithms must used, which are not yet available. The bivariate polynomials used by the list decoding algorithms can have large degrees even for RS codes with small parameters. Hence fast algorithms are needed.

Then I will present the flexibility of DECODING, needed to obtain efficient algorithms over several finite rings and fields:

- it is easy to replace a key algorithm, such as univariate polynomials multiplication or univariate polynomials root-finding, by a very efficient one provided by an external library such as FLINT or NTL.
- it is easy to choose a finite ring or a finite field, or even to use different rings at the same time in order to implement algorithms related to RS codes over Galois rings. The DECODING library is not restricted to mathemtical object whose representation holds in a single machine word. It requires no supplementary efforts to use, for exemple, multiprecision integers from GMP or large binary fields from MPFQ.

Finally, I will present the provided algorithms concerning dense bivariate polynomials and their applications to list decoding.

References

[BLQ11] J. Berthomieu, G. Lecerf, and G. Quintin. Polynomial root finding over local rings and application to error correcting codes. 2011.

[Gao02] S. Gao. A new algorithm for decoding Reed-Solomon codes. In *Communications, Information and Network Security, V. Bhargava, H.V. Poor, V. Tarokh, and S. Yoon*, pages 55–68. Kluwer, 2002.

[Gol07] I. Goldberg. Percy++. Software available from `http://percy.sourceforge.net/`, 2007.

[GS98] V. Guruswami and M. Sudan. Improved decoding of Reed-Solomon and algebraic-geometric codes. *IEEE Trans. Inform. Theory*, 45:1757–1767, 1998.

[Jus76] J. Justesen. On the complexity of decoding Reed-Solomon codes (corresp.). *IEEE Trans. Inform. Theory*, 22(2):237–238, March 1976.

[McE03] R. J. McEliece. The guruswami-sudan decoding algorithm for reed-solomon codes, 2003.

[RR98] R. M. Roth and G. Ruckenstein. Efficient decoding of Reed-Solomon codes beyond half the minimum distance. In *IEEE Trans. Inform. Theory*, page 56, 1998.

[RS60] I. S. Reed and G. Solomon. Polynomial codes over certain finite fields. *Journal of the Society for Industrial and Applied Mathematics*, 8(2):300–304, 1960.

[Sud97] M. Sudan. Decoding Reed-Solomon codes beyond the error-correction diameter. In *the 35th Annual Allerton Conference on Communication, Control and Computing*, pages 215–224, 1997.

[WB99] S.B. Wicker and V.K. Bhargava. *Reed-Solomon Codes and Their Applications*. John Wiley & Sons, 1999.