Progressive Algebraic Soft Decoding of Reed-Solomon Codes with a Gröbner Basis Approach

Yi Lv, Li Chen

School of Information Science and Technology, Sun Yat-sen University, Guangzhou, China.

***Abstract*—In algebraic soft decoding (ASD) of Reed-Solomon (RS) codes, interpolation dominates the computational complexity. This paper proposes an efficient ASD variant which performs interpolation in a progressive manner using a Gröbner basis approach. Our simulation results show that the proposed algorithm offers a significant complexity reduction.**

I. Introduction

Reed-Solomon (RS) codes are widely employed in digital communications and storage systems. The algebraic soft decoding (ASD) algorithm outperforms the algebraic hard decoding (AHD) and the conventional unique decoding algorithms. But its decoding complexity remains high due to the computationally expensive interpolation process. This paper proposes a progressive ASD (PASD) [1] algorithm that performs interpolation with a modified Gröbner basis, facilitating the advanced decoding of RS codes.

Let $\mathbb{F}_q$ denote the finite field of q elements. Given the message polynomial $u(x) = \sum_{i=0}^{k-1} u_i x^i$, where $u_i \in \mathbb{F}_q$, the codeword $\underline{c} \in \mathbb{F}_q^n$ of an (n, k) RS code can be generated by:

$$\underline{c} = (c_0, c_1, \cdots, c_{n-1}) = (u(x_0), u(x_1), \cdots, u(x_{n-1})), \quad (1)$$

where $x_0, x_1, \cdots, x_{n-1}$ are n distinct nonzero elements of $\mathbb{F}_q$. For the reliability matrix $\Pi \in \mathbb{R}_{q \times n}$ with entry $\pi_{i,j} = \Pr[c_j = \rho_i]$ and $\rho_i \in \mathbb{F}_q$, we can identify the maximal entry of each column as $\pi_j^* = \max\{\pi_{i,j} \mid i = 0, 1, \cdots, q-1\}$. Then we order the maximal entries, yielding a refreshed column indices sequence $\theta_0, \theta_1, \cdots, \theta_{k-1}, \cdots, \theta_{n-1}$, which implies $\pi_{\theta_0}^* > \pi_{\theta_1}^* > \cdots > \pi_{\theta_{k-1}}^* > \cdots > \pi_{\theta_{n-1}}^*$. We use $i(\theta) = \{i | \pi_{i,\theta} = \pi_\theta^*\}$ to denote the row index of π_θ^*, and we can constitute the set R of points to perform re-encoding, that is $R = \{(x_{\theta_0}, \rho_{i(\theta_0)}), (x_{\theta_1}, \rho_{i(\theta_1)}), \cdots, (x_{\theta_{k-1}}, \rho_{i(\theta_{k-1})})\}$ and $|R| = k$. The proposed algorithm performs interpolation with a series of designed output list size (OLS) $l_1, l_2, \cdots, l_{v-1}, l_v, \cdots, l_T$, where l_T is the maximal OLS that is set according to the the system's decoding complexity budget. The progressive decoding can be described in three stages. Stage 1, Lagrange interpolation of the points in R will be performed to obtain a re-encoding polynomial $r(x)$, which satisfies $\rho_{i(\theta_t)} = r(x_{\theta_t})$ and $t = 0, 1, \cdots, k-1$. Stage 2, the multiplicity matrix M will be obtained by transforming Π with a designed OLS. For (x_j, ρ_i) with a nonzero multiplicity, perform the coordinate transform to obtain $(x_j, \rho_i + r(x_j))$. Stage 3, let $\Theta = \{\theta_0, \theta_1, \cdots, \theta_{k-1}\}$. For $w \in \mathbb{Z}$, we define $[w]^+ = \max\{w, 0\}$. A polynomial group $\mathbf{G} = \{g_0, g_1, \cdots, g_{l_1}\}$ will be initialized and the polynomials of $\mathbf{G}$ are defined as [2]:

$$g_j = y^j \prod_{\delta \in \Theta} (X - x_i)^{[m_{i(\delta),\delta} - j]^+}, \quad (2)$$

where $j = 0, 1, \cdots, l_1$ and $m_{i(\delta),\delta} \in$ M. They satisfy the constraints w.r.t. points of R. Polynomial group $\mathbf{G}$ defines the Gröbner basis on which the following interpolation will be performed with $l_1 = 1$. Specifically, interpolation will be performed w.r.t. the remaining points that are implied by M, yielding a polynomial $Q(x, y)$. Factorization will be performed to obtain its y-roots $u'(x)$. Perform shifting of $u'(x) + r(x)$ to find the intended message polynomial $u(x)$, with which the codeword $\tilde{\underline{c}}$ is generated. If $\tilde{\underline{c}}$ is identified as the most likely (ML) codeword, the decoding will be terminated and output $u(x)$. Otherwise, the OLS will be enlarged as $l_v = l_{v-1} + 1$ and $v = 2, 3, \cdots, T$. The progressive interpolation [1] will be performed either the ML codeword is found or l_T is reached.

II. Numerical results

Numerical results on the average computational complexity for decoding the (15, 13) RS code is shown by Fig.1. It can be seen that the average complexity of the proposed algorithm is channel dependent. Moreover, with using a Gröbner basis approach, the proposed algorithm has a lower complexity than the PASD algorithm and the ASD algorithm.

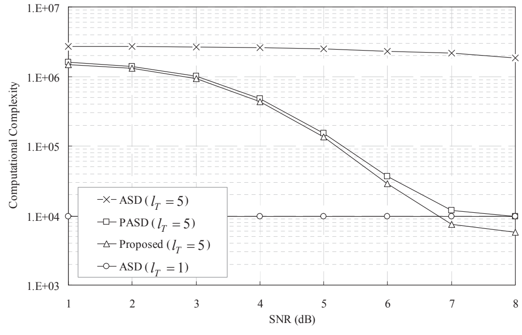

Fig. 1. Average computational complexity of the proposed algorithm in decoding the (15, 13) RS code.

References

[1] L. Chen and S. Tang and X. Ma, "Progressive Algebraic Soft-Decision Decoding of Reed-Solomon Codes," *IEEE Trans. Commun.*, vol. 61 (2), pp. 433-442, Feb. 2013.

[2] R. Koetter and J. Ma and A. Vardy, "The re-encoding transformation in algebriac list-decoding of Reed-Solomon codes," *IEEE Trans. Inform. Theory*, vol. 57 (2), pp. 633-647, Feb. 2011.