Analog Codes for Gross Error Correction in Signal Transmission

Fangning Hu[1, a], Werner Henkel[2,b] and Mingjie Zhao[3,c]

[1]Hangzhou Dianzi University, Hangzhou, China

[2]Jacobs University Bremen, Germany

[3]University of Manchester, England

[a]fangning.hu@hdu.edu.cn, [b]w.henkel@jacobs-university.de, [c]zhaom@cs.man.ac.uk

Keywords: Analog Codes, DFT Codes, Reed Solomon Codes, Syndrome Repairing

Abstract. We proposed a novel decoding algorithm for Analog Codes (Reed-Solomon Codes over complex numbers), the *syndrome repairing* (SR) algorithm, for gross error correction in signal transmission. Simulations show that, if the number of gross errors is not too large and the amplitude of background noise is small enough (compared to the amplitude of gross errors), the SR algorithm recovers the original signal with nearly the same accuracy as if no gross errors occur upon transmission. In particular, if the transmission is background-noise-free, then the recovery is exact.

Introduction

Analog Codes, or DFT Codes [1] is a class of Reed-Solomon Codes over the complex field $\mathbb{C}$ which allows for the detection and correction of multiple errors. The applications of Analog Codes for multiple errors correction in the transmission of continuous-amplitude signals has been discussed by many authors [2,3]; and can be summarized as follows. Assume we wish to send a signal $u \in \mathbb{C}^m$ to a remote receiver via a channel composed of background noise (or simply, noise) plus gross errors (or simply, errors) reliably. --- Here we distinguish *noise* and *errors* by their amplitudes. More concretely, by *noise* we mean a random vector w with each of its entries having Gaussian distribution $N(0,\sigma^2)$; and by *error* we mean a vector b whose elements are either 0 (no gross errors occur for this time step) or some large value e ($e \gg \sigma_w$). --- One way to address this problem is to transmit the codeword $x = Au$, where A is an $n \times m$ *encoding matrix* with $n > m$; and then recover the original signal u from the received data $y = x + b + w$. In Analog Codes, the encoding matrix A is formed by extracting m consecutive columns from the inverse discrete Fourier transform (IDFT) matrix (of order n)

$$F_n^{-1} = \frac{1}{n}\left[\phi_n^{-ik}\right]_{i,k\in\underline{n}} \quad \text{with} \quad \phi_n := e^{-j\frac{2\pi}{n}}, \tag{1}$$

where $\underline{n}$ denotes the set $\{0,1,\cdots,n-1\}$. The reason why the IDFT matrix F_n^{-1} is utilized here as the encoding matrix is that it possesses some good properties which enable us to recover the transmitted signal x from the corrupted data $y = x + b + w$.

As is well known, if the noise $w = 0$, almost all decoding algorithms for Reed-Solomon Codes over finite fields [4] can be (directly) utilized for Analog Codes to detect and correct up to $\left\lfloor \frac{n-m}{2} \right\rfloor$ errors. For the case in which $w \neq 0$, Wolf proposed a “voting” algorithm [1] and treated a simple example for correcting a single error in noise to illustrate the robustness of his algorithm. Wolf's method, however, has exponential complexity $O(n^t)$ (t is the number of errors) and hence is impractical for large values of t. All in all, an effective and robust decoding algorithm to detect and correct multiple errors in noise without using the statistics of the error and the signal is still missing.

In this paper we proposed a novel decoding algorithm for Analog Codes, the *syndrome repairing* (SR) algorithm, which is of polynomial complexity (better than Wolf's voting algorithm) and is capable of correcting multiple errors in noise. The core of the SR algorithm is the fact that, if there is no background noise, i.e., if $w=0$, the "biggest" *syndrome matrix* S (will be defined in the next Section) has rank t (the number of errors). So if the noise w is small enough, the $t+1$-th singular value (SV) σ_{t+1} of S should be near to zero. Thus, we can determine the number of errors t by checking the SVs of S. Moreover, the perturbation in S caused by the small noise w is also small, which makes it possible to approximate S by a new matrix R with rank t. We call R the *repaired syndrome matrix* --- this is why our algorithm is named "syndrome repairing". We then apply the classical Peterson-Gorenstein-Zierler (PGZ) decoder (over $\mathbb{C}$) [5] to recover the original signal u.
In the following sections, we first introduce the Analog Codes and the PGZ decoder using matrix notation; then introduce the SR algorithm in detail, whose performance were illustrated by some numerical experiments; and finally conclude the paper.

Analog Codes and the PGZ Decoder

Without loss of generality, let us take the last m columns of the IDFT matrix F_n^{-1} to form the encoding matrix A, yielding an (n,m) Analog Code

$$C := \{x \in \mathbb{C}^n \mid \hat{x}_k = 0, k \in \underline{d}\}, \tag{2}$$

where $d := n-m$ is the number of redundancies of the code C, $\hat{x}$ is the frequency domain of x.

In this section we introduce the PGZ decoder for the code C in a new manner that only employes matrix notation and concepts from linear algebra.
Applying DFT on both sides of $y = x+b+w$ we get $\hat{y} = \hat{x}+\hat{b}+\hat{w}$. Since x is a codeword in C, $\hat{x}_k$ and so $\hat{y}_k = \hat{b}_k + \hat{w}_k$ for all $k \in \underline{d}$. In the sequel, we will call these $\hat{b}_k$'s *syndromes* and $\hat{y}_k$'s *noised syndromes*. In particular, if the noise $w=0$, the syndrome $\hat{b}_k$ is equal to the noised syndrome $\hat{y}_k$. To derive the PGZ decoding algorithm, we need two propositions concerning the DFT.
Theorem 1 For any $b \in \mathbb{C}^n$ with weight $|b|_0 = t < \lfloor d/2 \rfloor$, where $|b|_0$ denotes the number of nonzero elements in the vector b, there is a unique vector $z \in \mathbb{C}^n$ satisfying (1) $z_i = 0$ iff $b_i \neq 0$ for all $i \in \underline{n}$; (2) $|\hat{z}_0| = \hat{z}_t = 1$ and $\hat{z}_k = 0$ for all $k > t$.
For any $\hat{b} \in \mathbb{C}^n$ and nonnegative integers α, β, γ such that $\beta - 1 \le \gamma \le n-\alpha$, we define $S^{\gamma}_{\alpha,\beta}(\hat{b})$ to be the $\alpha \times \beta$ matrix with the (i,k)-th entry $\hat{b}_{\gamma+i-k} (i \in \underline{\alpha}, k \in \beta)$

$$S^{\gamma}_{\alpha,\beta}(\hat{b}) := \begin{bmatrix} \hat{b}_{\gamma} & \hat{b}_{\gamma-1} & \cdots & \hat{b}_{\gamma-\beta+1} \\ \hat{b}_{\gamma+1} & \hat{b}_{\gamma} & \cdots & \hat{b}_{\gamma-\beta+2} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{b}_{\gamma-1+\alpha} & \hat{b}_{\gamma-2+\alpha} & \cdots & \hat{b}_{\gamma-\beta+\alpha} \end{bmatrix}. \tag{3}$$

Note that $S^{\gamma}_{\alpha,\beta}(\hat{b})$ is Toeplitz matrix.
Theorem 2 For any $b \in \mathbb{C}^n$ with $\|b\|_0 = t \le \lfloor d/2 \rfloor$, its DFT $\hat{b}$ has the property $rank\left\{S^{\gamma}_{\alpha,\beta}(\hat{b})\right\} = t$ whenever $\alpha, \beta \ge t$.
Now we are ready to introduce the PGZ algorithm. Assume that $\|b\|_0 = t \le \delta := d/2$ and that $w=0$, then $y = x+b$. For this error vector b, let $z \in \mathbb{C}^n$ be the unique vector stated by Theorem 1, called the *error locator* (in the time-domain). It has been already known in [5] that

$$S_{\alpha,\beta}^{\gamma}(\hat{b})\hat{z}_{(\beta)} = 0 \qquad (\beta \ge t+1, \gamma \ge t) \tag{5}$$

where $\hat{z}_{(\beta)} := \left[\hat{z}_0, \hat{z}_1, \cdots, \hat{z}_{\beta-1}\right]$. For $k < d$, $\hat{b}_k = \hat{y}_k$, since $2t \le d$, we have $\hat{b}_{(t)} = \hat{y}_{(t)}$. With initial values $\hat{b}_k = \hat{y}_k (k \in \underline{d})$, the PGZ algorithm can be summarized as follows [5]:

Algorithm 1 *The PGZ algorithm for the code C (Eq. 3).*

1) determine the number of errors t by

$$t = rank\left\{S_{d-\delta,\delta+1}^{\delta}(\hat{b})\right\} = rank\left\{S_{d-\delta,\delta+1}^{\delta}(\hat{y})\right\}; \tag{6}$$

2) compute the vectors $\hat{z}, \hat{b}$ by setting $\alpha, \beta, \gamma = t$ in Eq. 5:

$$\hat{z}_{(t)} = -\left[S_{t,t}^{t}(\hat{y})\right]^{-1} \hat{y}_{(t)}, \qquad \hat{y}_k = \hat{b}_k, \qquad (k \in \underline{d}) \tag{7}$$

$$\hat{b}_k = -\hat{z}_0^{-1}\sum_{l=1}^{t}\hat{z}_l\hat{b}_{k-l} \qquad (d \le k < n); \tag{8}$$

3) recover the input u which is the vector consisting of the last m elements of $\hat{x} = \hat{y} - \hat{b}$.

The Syndrome Repairing Decoder

The PGZ decoder can detect and correct at most $\lfloor d/2 \rfloor$ when $w = 0$. In this section, we consider the case where the noise $w \ne 0$ has Gaussian distribution; and the number of errors t is less than $\delta = \lfloor d/2 \rfloor$.

Firstly, we determine the number of errors t. By Theorem 2, all syndrome matrices $S_{\alpha,\beta}^{\gamma}(\hat{b})$ with $\alpha, \beta \ge t$ have rank t and thus have the singular values (SVs):

$$\sigma_1^* \ge \sigma_2^* \ge \cdots \ge \sigma_t^* > 0 = \sigma_{t+1}^* = \cdots \quad . \tag{9}$$

If the noise w is small enough, then, by matrix perturbation theory [6], the noised syndrome matrix $S_{\alpha,\beta}^{\gamma}(\hat{y})$, as a perturbed version of $S_{\alpha,\beta}^{\gamma}(\hat{b})$, should have SVs very near to that of $S_{\alpha,\beta}^{\gamma}(\hat{b})$. Numerical experiment show that there is a big drop between the t-th SV and the $t+1$-th SV of $S_{\alpha,\beta}^{\gamma}(\hat{y})$ (the "biggest" noised syndrome matrix we can form), providing us with an empirical formula for determining the number of errors t:

$$t = \max\left\{i \mid \sigma_i > 2\sigma_{i+1} - \sigma_{i+2} + 6\sigma_{last}\right\}, \tag{10}$$

where σ_i is the i-th SV of the matrix $S_{d-\delta,\delta+1}^{\delta}(\hat{y})$ and σ_{last} its last SV.

Next, we would determine the positions of errors. For the case where $w = 0$, the PGZ algorithm provides a method for computing the error locator z from the syndromes $\hat{b}_k = \hat{y}_k$ (cf. Eq. 7). When $w \ne 0$, only the noised syndromes $\hat{y}_k = \hat{b}_k + \hat{w}_k$ ($k \in \underline{d}$) are available and one can only obtain inaccurate error locator $\hat{z}$ by Eq. 7. But Eq. 7 only utilized the first $2t$ noised syndromes; and one may get more accurate estimations of the error locator by using the whole family of noised syndromes, by putting $\alpha = d-t, \beta = d+1$ and $\gamma = t$ in Eq. 5. Now Eq. 5 becomes an overdetermined system $S_{d-t,t+1}^{t}(\hat{b})\hat{z}_{(t+1)} = 0$, i.e., $S_{d-t,t}^{t}(\hat{b})\hat{z}_{(t)} = -\hat{b}_{d-t}$. When the noise w is small enough, the syndromes $\hat{b}_k$ can be approximated by the corresponding noised syndromes $\hat{y}_k$ ($k \in \underline{d}$), thus we may estimate the error locator by

$$\hat{z}_{(t)} \approx -\left[S_{d-t,t}^{t}(\hat{y})\right]^{\dagger} \hat{y}_{(d-t)}. \tag{11}$$

Comparing to Eq. 7, the above formula uses more noised syndromes and may procures more accurate estimations of the error locator $\hat{z}$. Moreover, based on Eq. 6, we can estimate the error locator even more accurately. Here is the basic idea: first modify the noised syndromes $\hat{y}_k$ ($k \in \underline{d}$) a little such that the modified syndromes, denoted by $\hat{r}_k$, have the property stated by Eq. 6, i.e., $rank\left\{S^{\delta}_{d-\delta,\delta+1}(\hat{r})\right\}=t$; and then estimate the error locator by Eq. 11 but with all $\hat{y}_k$'s replaced by the modified syndromes $\hat{r}_k$. The procedure of modifying the noised syndromes $\hat{y}_k$ to $\hat{r}_k$ is called *syndrome repairing* (SR), and the whole decoding procedure is hence called the syndrome repairing algorithm. The intuition underlying SR is quite clear: since $\hat{y}_k$ is a noised version of $\hat{b}_k$ and since the syndromes $\hat{b}_k$ satisfy Eq. 6, one may obtain a better approximation $\hat{r}_k$ of $\hat{b}_k$ than $\hat{y}_k$ by requiring that $\hat{r}_k \approx \hat{y}_k$ and $rank\left\{S^{\delta}_{d-\delta,\delta+1}(\hat{r})\right\}=t$. The SR procedure now amounts to the problem of low rank approximation of Toeplitz matrices:

$$\begin{aligned} \min \quad & J=\left\|S^{\delta}_{d-\delta,\delta+1}(\hat{y}-\hat{r})\right\|_F^2 \\ s.t. \quad & rank\left\{S^{\delta}_{d-\delta,\delta+1}(\hat{r})\right\}=t \end{aligned}, \qquad (12)$$

where $\|\bullet\|_F$ denotes the Frobenius norm of matrices. This problem can be efficiently solved by the lift-and-project algorithm [7], as follows.

Algorithm 2

The lift-and-project algorithm for Eq. (12).

Input: The noised syndromes $\hat{y}_k$ ($k \in \underline{d}$) and the rank t of the target matrix.

Procedure: Let $\upsilon=0$, $R^{(\upsilon)}=S^{\delta}_{d-\delta,\delta+1}(\hat{y})$ and repeat the following steps until convergence.

1) ***Lift*****:** (a) Compute the SVD $R^{(\upsilon)}=U^{(\upsilon)}\Sigma^{(\upsilon)}V^{(\upsilon)}$;

(b) replace $\Sigma^{(\upsilon)}$ by $S^{(\upsilon)}=diag\left\{\sigma_1^{\upsilon},\sigma_2^{\upsilon},\cdots,\sigma_t^{\upsilon},0,\cdots,0\right\}$ and define $X^{(\upsilon)}=U^{(\upsilon)}S^{(\upsilon)}V^{(\upsilon)}$.

2) ***Project***: Let $R^{(\upsilon+1)}$ be the matrix formed from $X^{(\upsilon)}$ by replacing the diagonals of $X^{(\upsilon)}$ by the averages of their entries, forcing it to be Toeplitz again.

3) Put $\upsilon \leftarrow \upsilon+1$ and return Step 1).

Output: Extract the repaired syndromes $\hat{r}_k$ from the Toeplitz matrix $R^{(\upsilon)}$ in an obvious way.

The syndrome repairing algorithm can now be stated as follows.

1) Compute the SVD of $S^{\delta}_{d-\delta,\delta+1}(\hat{y})$, where $\delta=\left\lfloor \frac{d}{2} \right\rfloor$, and determine the number of errors t by Eq. 10.

2) Compute the repaired syndromes $\hat{r}_k$ by Algorithm2.

3) Estimate the error locator $\hat{z}$ by $\hat{z}_{(t)}=-\left[S^{t}_{d-t,t}(\hat{r})\right]^{\dagger}\hat{r}_{d-t}$.

Numerical Experiments

In this section, the performance of the SR algorithm is verified by some numerical experiments. In the first experiment, a (40,20) Analog Code is considered (so $d=20$ and $\delta=10$). We set the variance of noise $\sigma_w=10^{-3}\sim 10$, the number of errors $t\in\{1,2,\cdots,5\}$ and the amplitude of errors $e=10$. 5000 input vectors u are randomly produced; for each of which we detect the error locations by the PGZ decoder, Eq. 11 and the SR algorithm. Fig. 1 shows the percentage of correctly detected error locations versus the amplitude of noise. We see that even for very small noise ($\sigma_w=10^{-3}$), PGZ fails to find all the error locations; whereas Eq. 11 works perfectly well when $\sigma_w \le 10^{-2}$ and SR is even better: it finds all error locations for $\sigma_w \le 2\times 10^{-1}$.

In the second experiment, we use a (10, 3) Analog Codes to transmit the vector $u=[1,2,3]^T$. To the codeword x we add Gaussian noise with deviation $\sigma_w=0.1$ and $t=2$ errors of amplitude $e=10$ (on randomly chosen positions). The received data y are decoded by our SR algorithm and the least squares method (with or without the knowledge of gross errors). The experiment were repeated 1000 times, yielding 1000 corrupted data y. In Fig. 2 we plotted the decoding result of the first 100 data, where the three elements of the (estimated) input are denoted respectively by plus (+), dot ($\cdot$) and cross ($\times$); and the real input $u=[1,2,3]^T$ is represented by circled red points. We see from the figure that the SR algorithm (the middle subplot) recovers the input with nearly the same accuracy as the LS method for which the gross errors is assumed to be known (the top subplot); and that the LS method simply failed to recover the input if the errors are unknown (the bottom subplot).

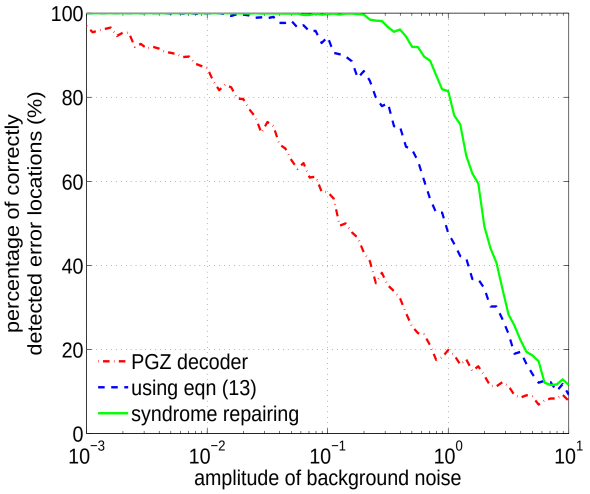

Fig.1 The influence of noise on correctly detected error locations ($t=1,2,\cdots,5$)

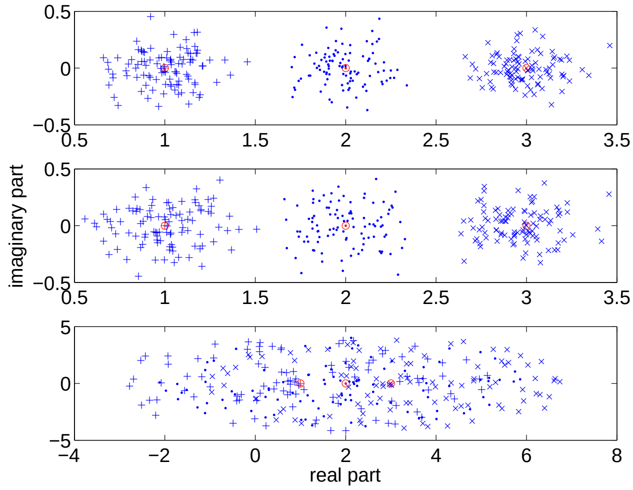

Fig. 2 Decoding results of SR (middle) and LS using (top) or not using (bottom) the knowledge of gross errors.

Conclusions

Decoding Analog codes is essentially a linear problem which requires us to solve a corrupted linear system. Based on matrix perturbation theory, we proposed a novel algorithm, the syndrome repairing (SR) algorithm, for decoding Analog Codes in background noise. Simulations show that the SR algorithm is efficient, accurate, and robust (against the background noise). More concretely, the SR algorithm gives reconstruction errors which are nearly as sharp as if no gross errors had occurred. Another important property of SR is that the performance of SR only depends on the error-noise-ration (not on signal-noise-ratio), as can be seen from the definition of syndrome matrices.

References

[1] Jack Keil Wolf, "Analog codes," in IEEE International Conference on Communication (ICC'83), Boston, MA, USA, vol. 1, pp. 310–312, June 1983.

[2] Werner Henkel and Fangning Hu, "OFDM and analog RS/BCH codes," in 10th International OFDM-Workshp, Hamburg, August 2005.

[3] Thomas G. Marshall, "Coding of real-number sequences for error correction: a digital signal processing problem," IEEE Journal on Selected Areas in Communications, vol. SAC-2, no. 2, pp. 381–392, March 1984.

[4] I. S. Reed and G. Solomon, "Polynomial codes over certain finite fields," Journal of the Society for Industrial and Applied Mathematics, vol. 8, no. 2, pp. 300–304, 1960.

[5] Richard E. Blahut, Algebraic codes for data transmission, Cambridge University Press, 2003.

[6] G. W. Stewart and Ji-Guang Sun, Matrix Perturbation theory, Academic Press, San Diego, 1990.

[7] Moody T. Chu, Robert E. Funderlic, and Robert J. Plemmons, "Structured low rank approximation," Linear Algebra and Its Applications, vol. 366, pp. 157–172, June 2003.