

# Quantizing Sparse Regression Codes

Chengpin Luo<sup>1</sup>   Lei Liu<sup>2</sup>   Brian M. Kurkoski<sup>1</sup>

<sup>1</sup>Japan Advanced Institute of Science and Technology (JAIST)

<sup>2</sup>Zhejiang University

Information Theory and Its Applications (ITA) Workshop

San Diego, California, USA

February 9, 2026

# Background and Motivation

Sparse regression (SR) codes<sup>1</sup> achieve the Shannon capacity of the AWGN channel.

Approximate message passing (AMP) is a practical decoder for SR codes.

How do SR codes perform in the finite-length domain?

- ▶ Unmodified SR codes have poor finite-length performance
- ▶ Applying **clipping** to the codeword can improve error-rate performance
- ▶ SR codes with **irregular clipping** can further improve finite-length performance <sup>2</sup>
  - ▶ Performance can be optimized by irregular clipping, typically requiring 4–10 clippers

---

<sup>1</sup>A. Joseph and A. R. Barron, Least squares superposition codes of moderate dictionary size are reliable at rates up to capacity, IEEE transactions on Information Theory, 2012

<sup>2</sup>Li et al. Irregularly Clipped Sparse Regression Codes, IEEE ITW, 2021

# Motivation

Can SR codes be made hardware-friendly?

- ▶ Implementations use discrete, fixed-point values.
- ▶ We take a first step of quantizing the SR codeword before transmission.

Performance improvement by non-linear operations?

- ▶ Clipping is a nonlinear operation
- ▶ Quantization is also nonlinear — *we show improvements over clipping at high rates*
- ▶ Possible generalizations to other nonlinear operations

## Standard Sparse Regression (SR) Code

SR codeword  $\mathbf{c}$  is encoded as:

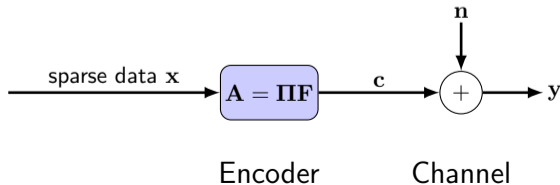
$$\mathbf{c} = \mathbf{A}\mathbf{x}$$

where:

- ▶  $\mathbf{A}$  is a randomly sampled discrete cosine transform (DCT) matrix.  $\mathbf{A} = [\mathbf{\Pi}\mathbf{F}]_{1:M}$ , where  $\mathbf{\Pi} \in \mathbb{R}^{M \times N}$  is a random permutation matrix,  $\mathbf{F} \in \mathbb{R}^{N \times N}$  is full DCT.
- ▶  $\mathbf{x}$  is information as a sparse vector:

$$\mathbf{x} = [ 0 \dots \sqrt{B} \dots 0 \mid \underbrace{0 \dots \sqrt{B} \dots 0}_{\text{encode } \log B \text{ bits}} \mid \dots \mid 0 \dots \sqrt{B} \dots 0 ]$$

with  $L$  sections and  $B$  entries per section.

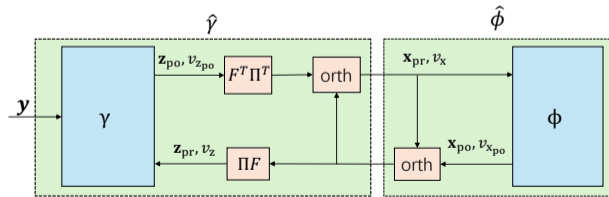


# Orthogonal AMP (OAMP) Decoder

- ▶ Consider the AWGN channel model:  $\mathbf{y} = \mathbf{c} + \mathbf{n}$ , where  $\mathbf{n} \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_M)$
- ▶ Orthogonal approximate message passing (OAMP) is used for decoding
  - ▶  $\gamma$ : MMSE estimator for  $\mathbf{y} = \mathbf{z} + \mathbf{n}$  where  $\mathbf{z} \sim \mathcal{N}(\mathbf{z}_{\text{pr}}, v_z \mathbf{I}_M)$
  - ▶  $\phi$ : MMSE estimator for  $\mathbf{x}_{\text{po}} = \mathbf{x} + \mathcal{N}(0, v_{x_{\text{po}}} \mathbf{I}_N)$
  - ▶ Orthogonalizer ensures the estimation errors between  $\hat{\gamma}$  and  $\hat{\phi}$  are independent:

$$v_{\text{or}} = \mathcal{O}_{\text{SE}}(v_{\text{po}}, v_{\text{pr}}) = (v_{\text{po}}^{-1} - v_{\text{pr}}^{-1})^{-1},$$

$$\mathbf{x}_{\text{or}} = \mathcal{O}\left(\mathbf{x}_{\text{po}}, v_{\text{po}}, \mathbf{x}_{\text{pr}}, v_{\text{pr}}, v_{\text{or}}\right) = v_{\text{or}} \left( \frac{\mathbf{x}_{\text{po}}}{v_{\text{po}}} - \frac{\mathbf{x}_{\text{pr}}}{v_{\text{pr}}} \right),$$



OAMP decoder of standard SR codes

Messages:

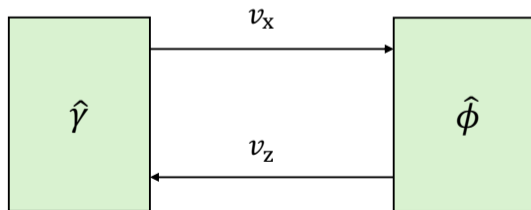
- means are vector  $\mathbf{z}$
- common variance  $v$

## State Evolution (SE)

- ▶ Compression ratio  $\delta = M/N \in (0, 1]$ ,  $M, N \rightarrow \infty$  (recall  $\mathbf{A} \in \mathbb{R}^{M \times N}$ )
- ▶ Without running the algorithm, the performance of the OAMP decoder can be tracked by its state evolution (SE):

$$\hat{\phi} : v_z = \phi_{\text{or}}^{\text{SE}}(v_x) \dots \text{details omitted}$$

$$\hat{\gamma} : v_x = v_z \left[ \frac{1}{\delta(1 - \gamma(v_z)/v_z)} - 1 \right].$$



State evolution tracks the variances.

## Nonlinear and Irregular SR Code

- ▶ Nonlinear SR codes are standard SR codes followed by nonlinear operations, such as clipping or quantization.
- ▶ A **regular nonlinear** SR code is

$$\mathbf{c} = \alpha Q(\mathbf{z}),$$

where  $Q(\cdot)$  is an entrywise nonlinear function,  $\mathbf{z}$  is the standard SR codeword and  $\alpha$  is a normalization factor to keep constant average power.

- ▶ An **irregular nonlinear** SR code is

$$\mathbf{c} = \begin{bmatrix} \mathbf{c}_1 \\ \vdots \\ \mathbf{c}_K \end{bmatrix} = \begin{bmatrix} \alpha_1 Q_1(\mathbf{z}_1) \\ \vdots \\ \alpha_K Q_K(\mathbf{z}_K) \end{bmatrix}$$

Nonlinear functions  $Q_1, \dots, Q_K$  are distinct.

The length of  $\mathbf{c}_i$  is  $\lambda_i M$ , i.e.  $\lambda_i$  is a degree distribution,  $\sum_i \lambda_i = 1$

## Clipping/Quantization as Nonlinear Functions

Clipping:

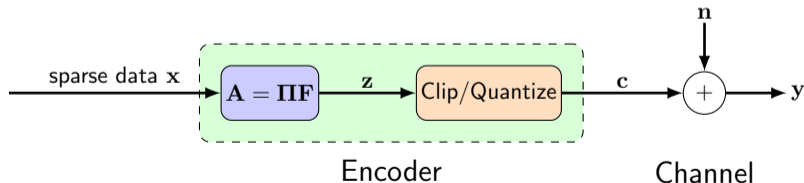
$$Q(z) = \begin{cases} -\epsilon, & \text{if } z \leq -\epsilon \\ z, & \text{if } -\epsilon < z < \epsilon \\ \epsilon & \text{if } z \geq \epsilon \end{cases}$$

where  $\epsilon$  is the clipping threshold. The clipping ratio (CR) is  $CR = 10 \log_{10} \frac{\epsilon^2}{E\{z^2\}}$ .

Quantization:

$$Q(z) = \begin{cases} \hat{z}_1, & \text{if } z \leq g_1 \\ \hat{z}_i, & \text{if } g_i < z \leq g_{i+1}, 1 \leq i \leq L \\ \hat{z}_L & \text{if } z \geq g_{L+1} \end{cases}$$

$g_1, \dots, g_{L+1}$  are quantization boundaries and  $\hat{z}_1, \dots, \hat{z}_L$  are quantization levels.

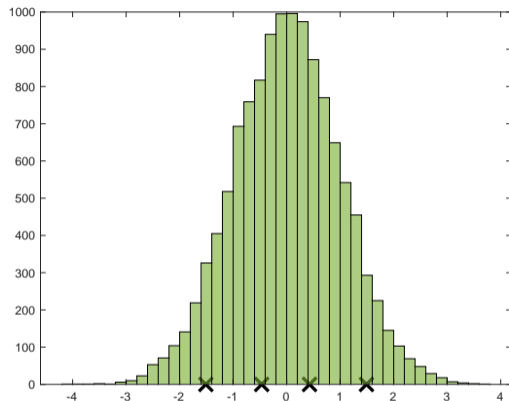


## Quantizer Implementaton

- ▶ The distribution of standard SR codes is approximately Gaussian, it appears

$$\lim_{M \rightarrow \infty} \mathbf{z} \sim \mathcal{N}(0, \mathbf{I}_M).$$

- ▶ Lloyd-Max/K-means algorithm on codeword samples is used for obtaining  $L$  quantization levels.
- ▶ For irregular quantization, subblocks  $\mathbf{z}_1, \dots, \mathbf{z}_K$  are quantized with distinct values,  $L_1, \dots, L_K$ .



Histogram of standard SR code symbols (green bars) and quantization levels (cross) obtained by Lloyd-Max algorithm.

## OAMP for Nonlinear SR Codes

- ▶  $\gamma$ : scalar MMSE estimator for  $y = \alpha Q(z) + n$ , where  $z \sim \mathcal{N}(z_{\text{pr}}, v_{\text{pr}})$ 
  - ▶ Declipper for clipping and dequantizer for quantization
  - ▶ For irregular case, there are  $K$  different estimators  $\gamma_1, \dots, \gamma_K$  for  $Q_1, \dots, Q_K$ .
- ▶  $\phi$ : same as standard SR codes

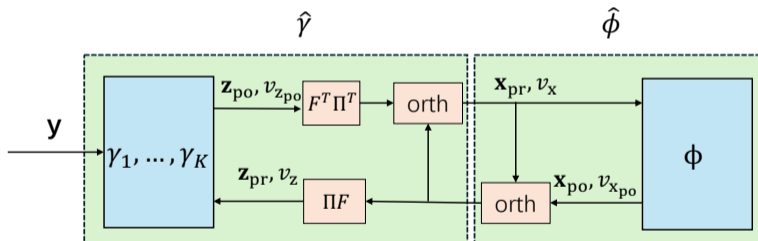


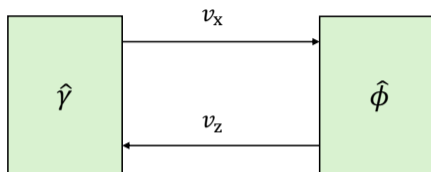
Figure 1: OAMP decoder of irregular SR codes

## State Evolution for Irregular SR Codes

- ▶ codeword  $\mathbf{c} = [\mathbf{c}_1^T, \dots, \mathbf{c}_K^T]$
- ▶ subblock fractions  
 $\boldsymbol{\lambda} = [\lambda_1, \dots, \lambda_K], \sum_{k=1}^K \lambda_k = 1$
- ▶ length of  $\mathbf{c}_k$ :  $\lambda_k M$
- ▶ Average output variance:  
 $v_{z_{po}} = \sum_{k=1}^K \lambda_k v_k^{po}$
- ▶ SE:

$$v_x = \gamma_{or}^{SE}(v_z, \boldsymbol{\lambda}) = v_z \left[ \frac{1}{\delta(1 - \sum_{k=1}^K \lambda_k v_k^{po}/v_z)} - 1 \right],$$

$$v_z = \phi_{or}^{SE}(v_x) = \dots \text{details omitted}$$



## Optimization of SE for Regular Quantization vs. Irregular Quantization

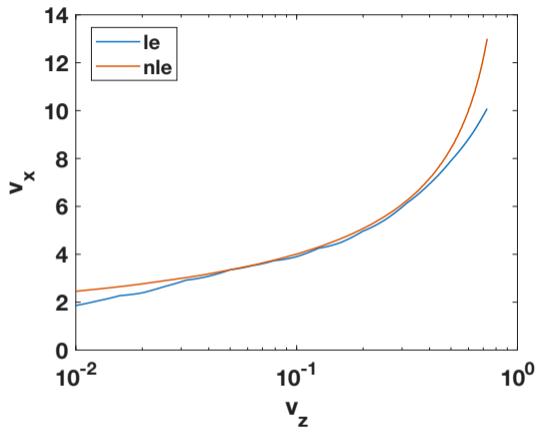


Figure 2: SE of regular quantized SR codes with an 8-level quantizer, SNR=6.1 dB

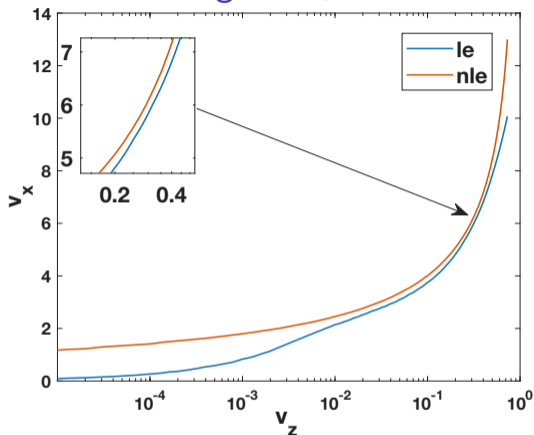


Figure 3: Optimized irregular quantized SR codes with a 4-level quantizer and a 20-level quantizer, SNR=6.1 dB

## SE Optimization

- ▶ Denote  $v_z, v_x$  at time  $t$  as  $v_{z,t}, v_{x,t}$ , the algorithm iterates as

$$v_{z,1} \rightarrow v_{x,1} = \gamma_{\text{or}}^{\text{SE}}(v_{z,1}, \boldsymbol{\lambda}) \rightarrow v_{z,2} = \phi_{\text{or}}^{\text{SE}}(v_{x,2}) \rightarrow \dots$$

- ▶ The gap over  $v_z$  is

$$\mathcal{G}_{\text{irr}}(v_z, \boldsymbol{\lambda}) = \phi_{\text{or}}^{\text{SE}^{-1}}(v_z) - \gamma_{\text{or}}^{\text{SE}}(v_z, \boldsymbol{\lambda}).$$

- ▶ Given SNR and parameters of quantizers/clippers, the goal is to find the minimal gap  $\mathcal{G}_{\text{irr}}(v_z, \boldsymbol{\lambda})$  over  $v_z$  and maximize this gap over  $\boldsymbol{\lambda}$ :

$$\begin{aligned} \max_{\boldsymbol{\lambda}} \min_{v_z \in \mathcal{V}} \mathcal{G}_{\text{irr}}(v_z, \boldsymbol{\lambda}), \\ \text{s.t. } \sum_{k=1}^K \lambda_k = 1, \\ 0 \leq \lambda_k \leq 1. \end{aligned}$$

# SER Performance Comparisons

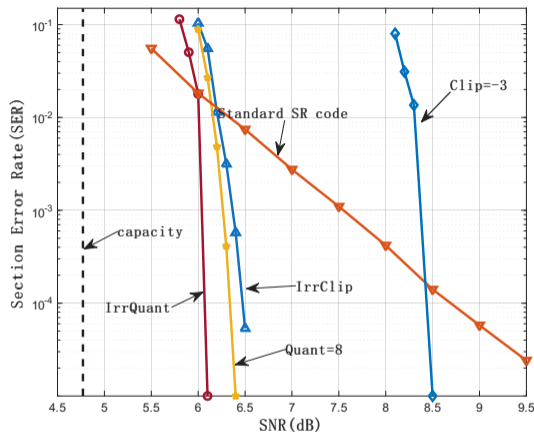


Figure 4: Comparisons between SER performance of regular quantized ( $k = 8$ ) and irregular quantized SR codes ( $k = 4$  and  $k = 20$  combined), standard SR codes, regular clipped SR codes (CR=-3), irregular Clipped SR codes, section length  $B = 64$ , number of sections  $L = 2048$ , code rate  $R = 1$ , codeword length  $M = 12288$ , max iteration number=60.

# Irregular Quantization vs Irregular Clip

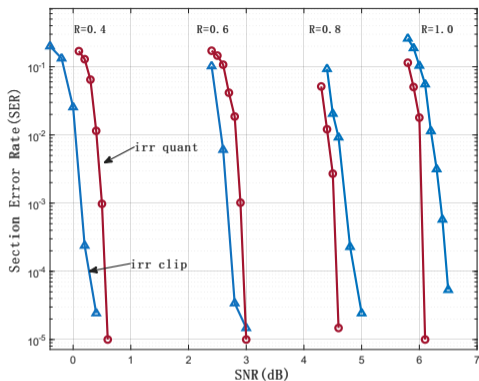
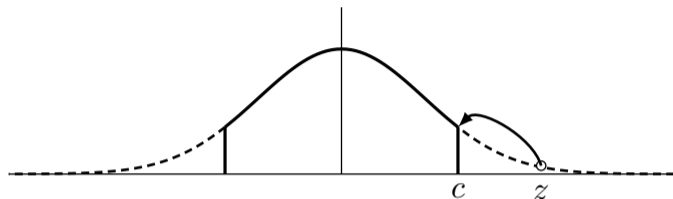


Figure 5: SERs of optimized irregular clipped SR codes (Triangles) vs. irregular quantized SR codes (Circles) with different code rates.

Table 1: Optimized Parameters for irregularly quantized SR codes,  $B=64$ .

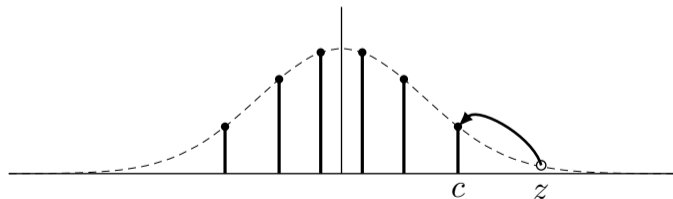
Rate	SNR	Levels	$\lambda$
0.4	0.4	2	0.6492
		28	0.3508
0.6	2.8	3	0.5432
		20	0.4568
0.8	4.8	3	0.3387
		32	0.6613
1.0	6.1	4	0.2648
		20	0.7352

## Why Does Clipping Work?



### Clipping

Due to clipping, can increase  $\alpha$  to maintain constant transmit power. This benefit is greater than the error of  $c = Q(z)$ .



### Quantization

Same benefit as clipping for large  $|z|$ . But why benefit at  $R = 1$ ?

## Conclusion

Motivated by hardware suitability and performance improvement, investigated quantized SR codes.

At higher rates  $R = 1$ :

- ▶ Regular quantized SR codes are 2 dB better than regular clipped
- ▶ Irregular quantized SR codes outperform standard SR codes (unclipped) by 3.5 dB.
- ▶ Irregular quantized SR codes outperform irregular clipped SR codes at higher code rates.

but this benefit disappears at low rates.

Future work:

- ▶ Other nonlinear functions,
- ▶ Multipath channel model,
- ▶ Explain the reason quantization is better than clipping at high rate.