Sparse Regression Codes and AMP Decoding: Impact on Modern Coding Paradigms

Brian M. Kurkoski

Japan Advanced Institute of Science and Technology

Presentation to Bangalore Chapter of IEEE Information Theory Society

2026 March 6

Collaborators

Lei Liu
Zhejiang University

Shunqi Huang
JAIST

Chengpin Luo
JAIST

Further Reading

- W. Li, L. Liu, and B. M. Kurkoski, "Irregularly clipped sparse regression codes," *Proceedings of the IEEE Information Theory Workshop*, Kanazawa, Japan, October 2021.
- L. Liu, S. Huang, and B. M. Kurkoski, "Memory AMP," *IEEE Transactions on Information Theory*, vol. 68, no. 12, pp. 8015–8039, 2022.
- S. Huang, L. Liu, and B. M. Kurkoski, "Overflow-avoiding memory AMP," *Proceedings of the IEEE International Symposium on Information Theory*, June 2024.
- L. Liu, Y. Chi, S. Huang, and Z. Zhang, "Random Multiplexing," *IEEE Transactions on Information Theory*, doi: 10.1109/TIT.2026.3653055, 2026.

Motivation

- Modern communication systems are seen as large linear estimation problems
  - High-dimensional: long blocklengths, many antennas
  - Mixing across the entire block via precoding or random transforms
  - Strong statistical dependence between symbols
- Approximate Message Passing (AMP)
  - Low-complexity alternative to optimal decoding
  - Predictable behavior via state evolution
  - Effective "decoupling" into scalar channels
- Goal of this talk:
  - Investigate communication problems that are suitable for AMP
  - Design the system so that AMP operates near its ideal regime
- Two design directions we explore:
  - Random linear transforms (Random Multiplexing)
  - Nonlinear shaping of codewords (Quantized SR codes)

Outline

Outline

Problem Formulation

Consider observing $\mathbf{y}$, a noisy version of $\mathbf{x}$:

$$\mathbf{y} = \boldsymbol{A}\mathbf{x} + \mathbf{n}$$

where $\boldsymbol{A} \in \mathbb{C}^{M \times N}$, $\mathbf{n} \sim \mathcal{CN}(0, \sigma^2 \boldsymbol{I}_M)$, and $\boldsymbol{A}, \boldsymbol{y}, \sigma^2, P_X(\cdot)$ are known.

- Goal: Given $\{\boldsymbol{y}, \boldsymbol{A}, \Gamma, \Phi\}$, find an MMSE estimate of $\boldsymbol{x}$:

$$\widehat{\boldsymbol{x}} = \arg\min_{\tilde{\boldsymbol{x}}} \mathbb{E}\left[\|\boldsymbol{x} - \tilde{\boldsymbol{x}}\|^2\right]$$

Classical LMMSE — When is LMMSE Optimal?

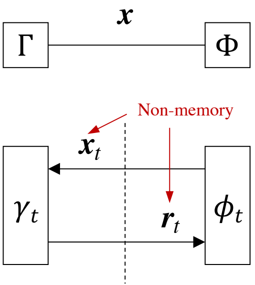
- Consider estimating $\boldsymbol{x}$ from $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{n}$ under MSE.
- **Fact 1:** The LMMSE estimator
  - Is optimal among all *linear* estimators
  - Has a closed-form solution (covariance-based formula)
- **Fact 2:** LMMSE equals the true MMSE estimator if
  - $\boldsymbol{x}$ and $\boldsymbol{n}$ are jointly Gaussian
  - The posterior is Gaussian $\Rightarrow$ conditional mean is linear
- Otherwise (discrete or structured priors):
  - The MMSE estimator is nonlinear
  - Exact computation requires high-dimensional integration
  - Generally computationally intractable
- This gap motivates low-complexity nonlinear estimators $\rightarrow$ AMP

Outline

Approximate Message Passing (AMP)

- AMP-type algorithms:

$$\text{linear estimator (LE)}: \quad \boldsymbol{r}_t = \gamma_t(\boldsymbol{x}_t),$$
$$\text{non}-\text{linear esitmator (NLE)}: \quad \boldsymbol{x}_{t+1} = \phi_t(\boldsymbol{r}_t).$$



- AMP:

$$\text{LE}: \quad \boldsymbol{r}_t = \boldsymbol{x}_t + \boldsymbol{A}^{\mathrm{H}}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}_t) + \boldsymbol{r}_t^{\text{Onsager}},$$
$$\text{NLE}: \quad \boldsymbol{x}_{t+1} = \phi(\boldsymbol{r}_t) = \mathrm{E}\{\boldsymbol{x}|\boldsymbol{r}_t\},$$

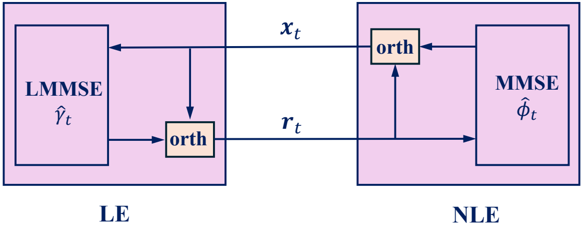
✓ Bayes optimal

✓ Low-complexity

× IID $\boldsymbol{A}$ is required

□ D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," in *Proc. Nat. Acad. Sci.*, 2009.

Orthogonal/Vector AMP (OAMP/VAMP)



✓ Bayes optimal (replica)

✓ Unitarily-invariant $\boldsymbol{A}$

× High-complexity

OAMP/VAMP:

- Standard AMP works well when the mixing matrix $\boldsymbol{A}$ behaves like an i.i.d. Gaussian random matrix; important practical systems do not satisfy this condition.
- **Key idea of OAMP:** design the algorithm so that estimation errors remain *orthogonal* across iterations.

□ J. Ma and L. Ping, "Orthogonal AMP," *IEEE Access*, 2017.

□ S. Rangan, P. Schniter, and A. Fletcher, "Vector approximate message passing," *IEEE Trans. Inf. Theory*, 2019.

Convergence Conditions and Various AMP Algorithms

Large-scale linear system:

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{n}, \quad s_i \sim P_s, \ \forall i.$$

For general $\boldsymbol{A}$, optimal detection of $\boldsymbol{s}$ from $\boldsymbol{y}$. is NP-hard.

Theorem (Optimality): For $\boldsymbol{A} \in \mathscr{U}$, AMP algorithms (e.g., OAMP/VAMP, MAMP...) can achieve replica optimality [Dudeja, 2024].

- AMP (2009)
  - ✓ low complexity
  - × IID Gaussian $\boldsymbol{A}$
- OAMP/VAMP (2017, 2019)
  - ✓ $\boldsymbol{J}$ is ROI $\rightarrow \boldsymbol{J} \in \mathscr{U}$
  - × high complexity
- MAMP (2022)
  - ✓ $\boldsymbol{J} \in \mathscr{U}$
  - ✓ low complexity

☐ D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," 2009.
☐ J. Ma and L. Ping, "Orthogonal AMP," 2017.
☐ S. Rangan, P. Schniter, and A. Fletcher, "Vector approximate message passing," 2019.
☐ L. Liu, S. Huang, and B. M. Kurkoski, "Memory AMP," 2022.

Outline

Challenges in Modulation — Orthogonal Frequency Division Multiplexing (OFDM)

- **Time-invariant multipath channel**:

  Circulant $\boldsymbol{A}$:

$$\boldsymbol{A} = \boldsymbol{F}^{\mathrm{H}} \boldsymbol{\Lambda} \boldsymbol{F}$$

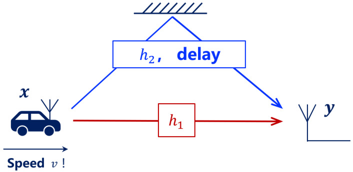
  with DFT matrix $\boldsymbol{F}$ and diagonal $\boldsymbol{\Lambda}$.

- **OFDM**:

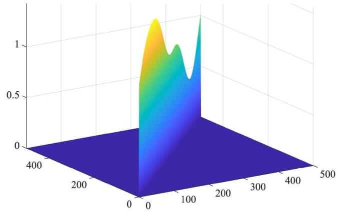
  Modulation matrix: $\boldsymbol{U} = \boldsymbol{F}^{\mathrm{H}}$

$$\begin{aligned}\widetilde{\boldsymbol{y}} &= \boldsymbol{U}^{\mathrm{H}} \boldsymbol{A} \boldsymbol{U} \boldsymbol{s} + \widetilde{\boldsymbol{n}} \\ &= \boldsymbol{F} \boldsymbol{F}^{\mathrm{H}} \boldsymbol{\Lambda} \boldsymbol{F} \boldsymbol{F}^{\mathrm{H}} \boldsymbol{s} + \widetilde{\boldsymbol{n}} \\ &= \boldsymbol{\Lambda} \boldsymbol{s} + \widetilde{\boldsymbol{n}}\end{aligned}$$

  ✓ Simple optimal MMSE detector



$v = 0$ km/h



OFDM: frequency domain

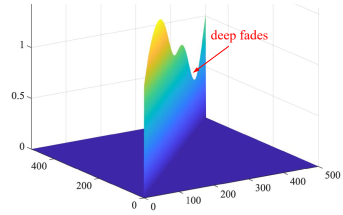
Challenges in OFDM

- **Frequency selective fading**:

  Deep fades:

  $$\mathbf{\Lambda} = \mathrm{diag}\{\lambda_1, \cdots, \lambda_N\}$$
  $$|\lambda_k| \ll |\lambda_i|, \; i \neq k$$

  × Poor performance



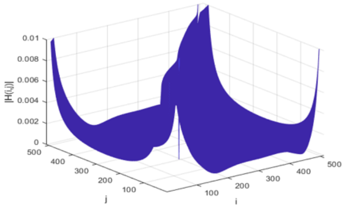
$v = 0$ km/h

- **High-mobility multipath channel**:

  Doppler effect: $\boldsymbol{A} \neq \boldsymbol{F}^{\mathrm{H}} \mathbf{\Lambda} \boldsymbol{F}$

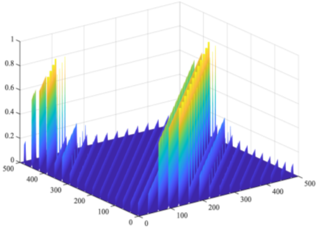
  $$\widetilde{\boldsymbol{y}} = \underbrace{\boldsymbol{F}\boldsymbol{A}\boldsymbol{F}^{\mathrm{H}}}_{\text{non-diagonal}} \boldsymbol{s} + \widetilde{\boldsymbol{n}}$$

  × Lack of efficient detectors



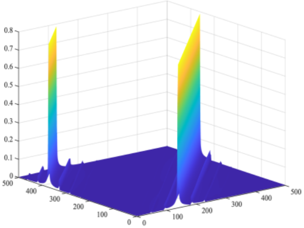
$v = 500$ km/h

High Mobility Using OTFS and AFDM, and Its Challenges



Orthogonal Time Frequency Space (OTFS)

$\boldsymbol{U} = \boldsymbol{F}_L^{\mathrm{H}} \otimes \boldsymbol{I}_K$, DD domain



Affine Frequency Division Multiplexing (AFDM)

$\boldsymbol{U} = \boldsymbol{\Lambda}_{c_1}^{\mathrm{H}} \boldsymbol{F}_N^{\mathrm{H}} \boldsymbol{\Lambda}_{c_2}^{\mathrm{H}}$, DAFT domain

Sparsification-oriented (instead of orthogonalization):

× ISI remains — Lack of efficient detectors

× Modulation matrix $\boldsymbol{U}$ depends on channel (coupling with channel matrix)

□ R. Hadani, S. Rakib, M. Tsatsanis, A. Monk, A. Goldsmith, A. Molisch, and R. Calderbank, "Orthogonal time frequency space modulation," IEEE WCNC, 2017.

□ A. Bemani, G. Cuozzo, N. Ksairi, and M. Kountouris, "Affine frequency division multiplexing for next-generation wireless networks," ISWCS, 2021.

Outline

Small-Dimensional Random Multiplexing: 2D with No Rotation

- BPSK signal $\boldsymbol{s}$, Gaussian noise $\boldsymbol{n}$
- **Orthogonal modulation:**

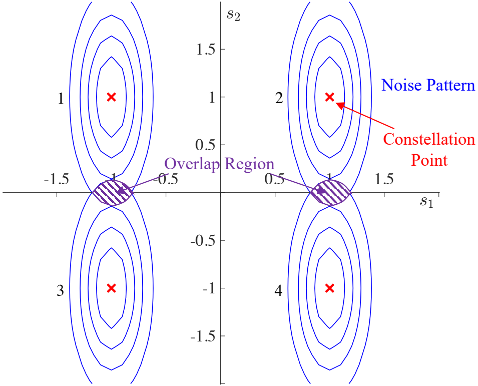
$$\boldsymbol{y} = \begin{bmatrix} 1.5 & 0 \\ 0 & 0.5 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2 \end{bmatrix}$$

$$\boldsymbol{y}' = \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + \begin{bmatrix} \frac{2}{3} n_1 \\ 2 n_2 \end{bmatrix}$$

- Covariance matrix:

$$\boldsymbol{V}_{n'} = \begin{bmatrix} 0.0444 & 0 \\ 0 & 0.4 \end{bmatrix}$$

- Degradation in one of the dimensions.



2D Example — Rotation

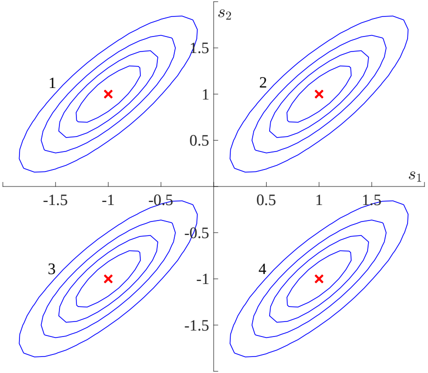
- **$45°$ rotated modulation:**

$$\boldsymbol{y} = \begin{bmatrix} 1.5 & 0 \\ 0 & 0.5 \end{bmatrix} \underbrace{\begin{bmatrix} \cos\frac{\pi}{4} & -\sin\frac{\pi}{4} \\ \sin\frac{\pi}{4} & \cos\frac{\pi}{4} \end{bmatrix}}_{\boldsymbol{U}} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2 \end{bmatrix}$$

$$\boldsymbol{y}' = \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + \boldsymbol{U}^H \begin{bmatrix} \frac{2}{3}n_1 \\ 2n_2 \end{bmatrix}$$

  - Covariance matrix:

$$\boldsymbol{V}_{n'} = \begin{bmatrix} 0.222 & 0.177 \\ 0.177 & 0.222 \end{bmatrix}$$



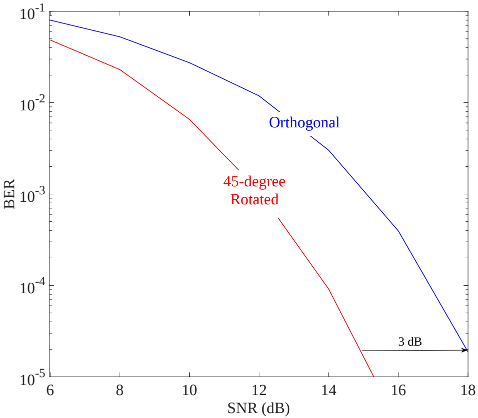
2D Example — Bit Error Rate (BER)

- 2D real system:

  45° rotated modulation is optimal.

- High dimensional cases:

  How to find a good $\boldsymbol{U}$?

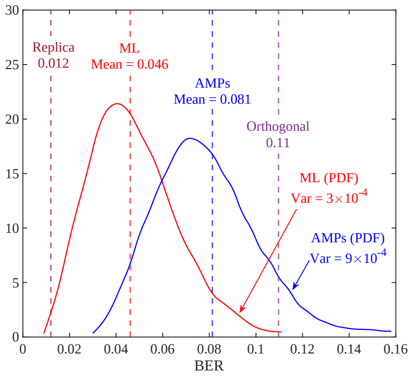


Rotation gives 3 dB gain with maximum likelihood (ML) detection

High-Dimensional Experiment — Random Multiplexing $N = 8$

1. Generate many $\boldsymbol{A}$ with eigenvalues $\boldsymbol{\Sigma}_A$
2. For each $\boldsymbol{A}$, generate $\boldsymbol{U} \sim \mathrm{Unif}\big(\mathbb{O}(N)\big)$

- Orthogonal: $\boldsymbol{y} = \boldsymbol{\Sigma}_A \boldsymbol{s} + \boldsymbol{n}$
- Random unitary: $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{U}\boldsymbol{s} + \boldsymbol{n}$
  - ML detection
  - AMP detectors
- Replica method: asymptotic BER



$N = 8.$
Distribution of BERs

High-Dimensional Experiment — Random Multiplexing $N = 16$–512

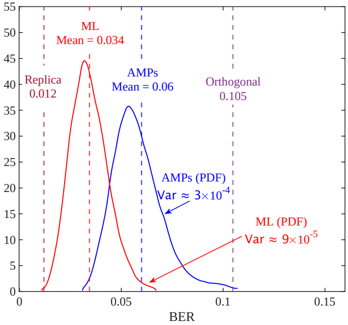


Figure 1: $N = 16$

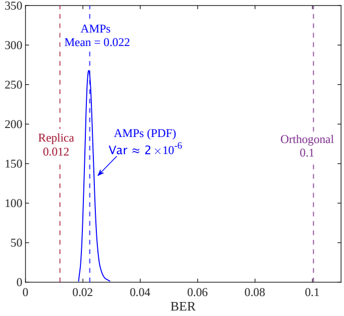


Figure 2: $N = 128$

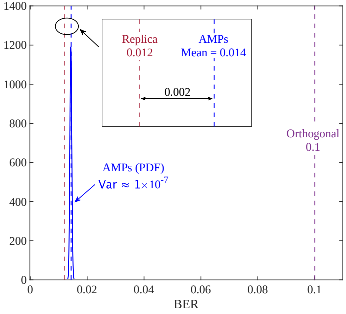


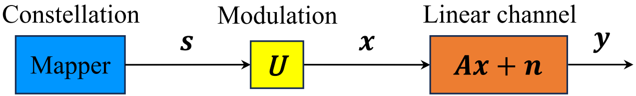
Figure 3: $N = 512$

Numerical experiments suggest:

- Random Multiplexing using random unitary $\boldsymbol{U}$,
- approaches the replica limit for large system size,
- achieved using low-complexity AMP

Outline

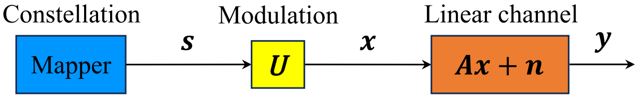
Modulation for Linear Channels



$$\boldsymbol{y} = \boldsymbol{A}\underbrace{\boldsymbol{U}\boldsymbol{s}}_{\boldsymbol{x}} + \boldsymbol{n}$$

- $\boldsymbol{s} \in \mathbb{C}^N$: Signal vector, (QAM, PSK, etc.)
- $\boldsymbol{U} \in \mathbb{U}(N)$: Unitary modulation matrix (OFTM, OTFS or random)
- $\boldsymbol{A} \in \mathbb{C}^{M \times N}$: Time-domain channel matrix
- $\boldsymbol{n} \sim \mathcal{CN}(\mathbf{0}, \sigma^2 \boldsymbol{I}_M)$: Gaussian noise
- $\boldsymbol{y} \in \mathbb{C}^M$: Received vector

Key Idea: Random Multiplexing



Traditional wireless systems choose a specific modulation transform

- OFDM: inverse DFT
- OTFS / AFDM: structured transforms designed for particular channel models

Instead, we propose **Random Multiplexing**:

- Modulate the symbol vector using a random unitary transform $\mathbf{U} \to \Xi$:

$$x = \Xi s$$

- The transmitted system becomes

$$y = A\Xi s + n$$

Why Random?

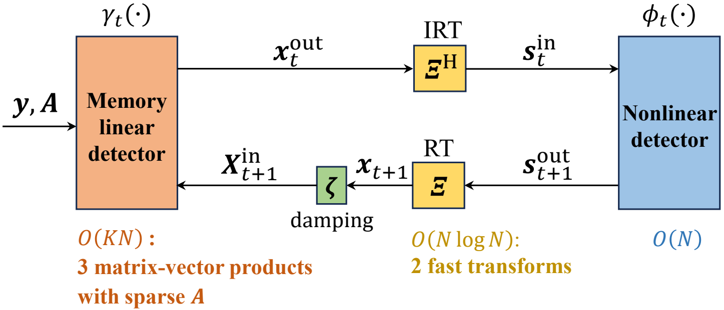
Why random?

- Random transform makes the effective channel matrix $A\Xi$ dense and statistically well behaved
- This places the system in the *universality class* where AMP-type detectors are theoretically optimal

Consequence

- For large systems with norm-bounded, spectrally convergent channels, AMP-type detectors achieve the replica MAP BER when the state evolution has a unique fixed point

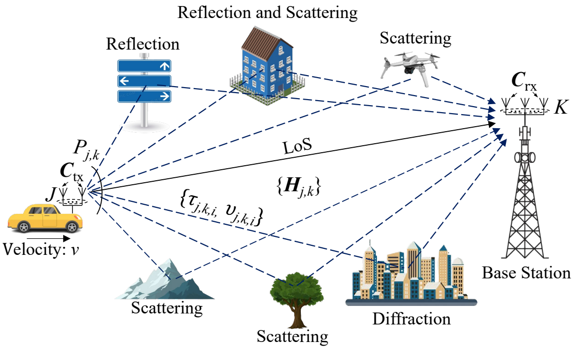
Cross-Domain MAMP (CD-MAMP) Detector



Instead of applying AMP to $\mathbf{A}\Xi$:

- The linear estimator is matched to $\mathbf{A}$ only. Exploit the sparsity of $\mathbf{A}$.
- Ξ is a permuted discrete Fourier transform. Use FFT for efficiency.

Simulation: High-mobility Multipath MIMO Channel



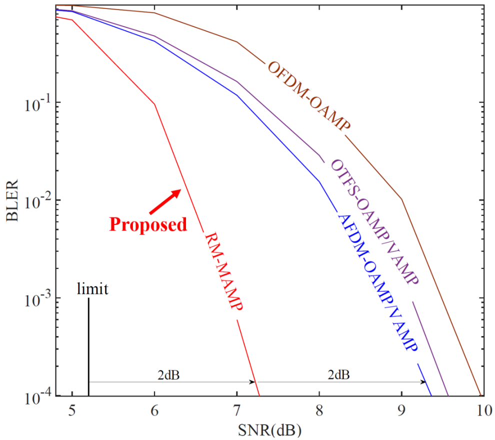
- K receive antennas, J transmit antennas, correlation coefficient $\rho \in \{0, 0.6\}$
- Channel impulse response between $(j,\ k)$-th antennas:

$$H_{j,k}[u,\ell] = \sum_{i=1}^{P_{j,k}} H_{j,k,i} e^{j2\pi\nu_{j,k,i}(uT_s - \ell T_s)} \mathrm{P}_{\mathrm{rc}}(\ell T_s - \tau_{j,k,i}).$$

$P_{j,k} = 5$: number of multipaths; P_{rc}: raised-cosine rolloff filter with factor 0.4;

Carrier frequency: 4 GHz with $\Delta f = 15$ kHz; Velocity: $v \in \{100, 300\}$ km/h

5G-NR LDPC: RM v.s. OFDM, OTFS and AFDM



4×4 MIMO, $\rho = 0.6$, coding rate 0.625, QPSK,

5 paths, $v = 300$km/h, $M = 2048, N = 2048$

5G-NR LDPC:

- The BLER performances of OFDM, OTFS, and AFDM are **comparable**.
- Compared to OFDM, OTFS, and AFDM, RM achieves a gain of **more than 2dB**.

Outline

Sparse Regression (SR) Codes: A Different Route to Shannon Capacity

Sparse regression (SR) codes[1] are an alternative to block codes:

- block codes enforce parity constraints on bits
- SR codes encode information as the support pattern of a sparse vector in a large linear dictionary.

Sparse regression (SR) codes encode information $\mathbf{x}$ to codeword $\mathbf{c}$:

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

over the **real** numbers, rather than a finite field.

Decoding becomes a signal recovery problem. Recover $\mathbf{x}$ from noisy $\mathbf{y}$:

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \text{noise}$$

[1] A. Joseph and A. R. Barron, "Least squares superposition codes of moderate dictionary size are reliable at rates up to capacity," *IEEE Trans. Inf. Theory*, 2012.

Sparse Regression (SR) Codes: A Different Route to Shannon Capacity

Decoding becomes a signal recovery problem. Recover $\mathbf{x}$ from noisy $\mathbf{y}$:

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \text{noise}$$

Approximate Message Passing (AMP) gives a fast, scalable decoder that exploits the high-dimensional linear model and the sparsity structure.

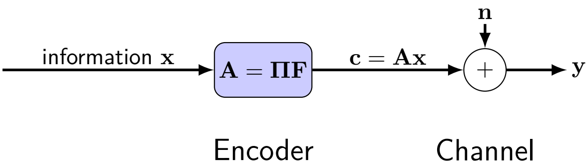
For AMP convergence two important properties:

(1) The matrix $\mathbf{A}$ should be behave like a large random mixing matrix.

(2) The information vector $\mathbf{x}$ must be sparse but structured.

Added bonus: under certain conditions $\mathbf{c} = \mathbf{A}\mathbf{x}$ tends to be approximately Gaussian distributed, matching the AWGN capacity-achieving input distribution.

Standard Sparse Regression (SR) Codes— (1) Encoding



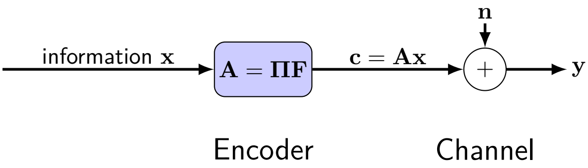
(1) SR codeword $\mathbf{c}$ is encoded as:

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

where $\mathbf{A} = \mathbf{\Pi F}$ is an $M \times N$ randomly sampled discrete cosine transform (DCT) matrix:

- DCT matrices are orthogonal, satisfying AMP requirement
- Use FFT for efficient encoding and decoding
- $M < N$ so $\mathbf{A}$ is a "wide" matrix.

Standard Sparse Regression (SR) Codes— (2) Sparse Information Vector



(2) The information vector $\mathbf{x}$ is sparse:

$$\mathbf{x} = [\, 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0 \mid \underbrace{0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0}_{\text{encode } \log B \text{ bits}} \mid \cdots \mid 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0 \,]$$

with L sections and B entries per section. Code rate is:

$$R = \frac{L}{M} \log B$$

Sparsity aids convergence of the decoder.

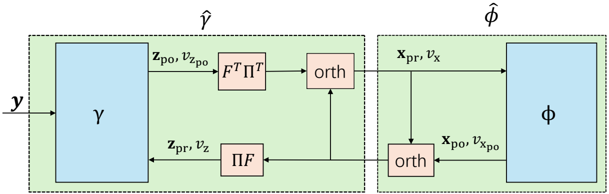
In practice, the $\mathbf{x}$ vector is power-normalized $\mathbf{x} \to \sqrt{B}\mathbf{x}$

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
  - γ: MMSE estimator for $\mathbf{y} = \mathbf{z} + \mathbf{n}$ where $\mathbf{z} \sim \mathcal{N}(\mathbf{z}_{\text{pr}}, v_z \mathbf{I}_M)$
  - ϕ: 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:

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

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

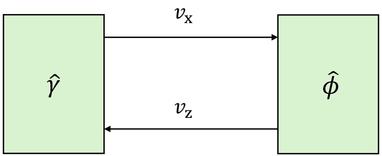


OAMP decoder of standard SR codes

Messages:
- means are vector $\mathbf{z}$ or $\mathbf{x}$
- common variance v

State Evolution (SE)

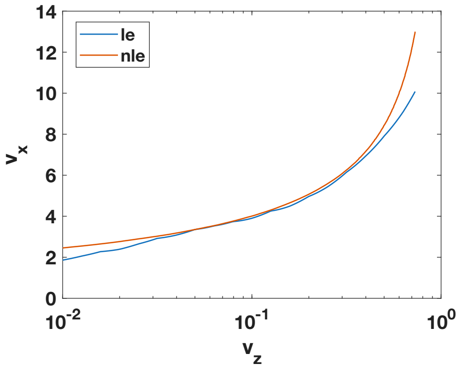
State evolution (SE) tracks the performance of OAMP without running the algorithm



$\widehat{\gamma}$: MMSE estimator for channel

$\widehat{\phi}$: MMSE estimator for sparsity

v_x, v_z: message variances



Variances start upper-right, iterate lower-left.

Curves don't cross→no fixed point→convergence.

Similar to EXIT charts for turbo codes.

Outline

Nonlinear SR Codes — Motivation

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

- Unmodified SR codes have poor finite-length performance

Performance improvement by non-linear operations?

- Clipping was shown to have modest performance improvement[2].
- Irregular clipping can improve finite-length performance[3].
- Quantization has further performance improvements over clipping at higher rates[4].
- Give some insight into *why* clipping can improve performance

We see clipping and quantization as special cases of a nonlinear operation.

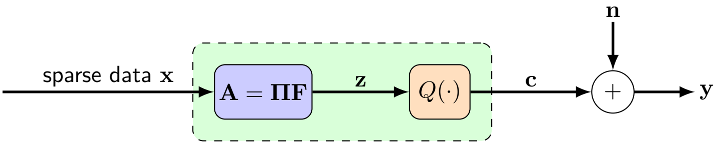
[2] S. Liang, J. Ma, and L. Ping, "Clipping can improve the performance of spatially coupled sparse superposition codes," IEEE Commun. Lett., Dec. 2017.

[3] W. Li, L. Liu, B. Kurkoski, "Irregularly Clipped Sparse Regression Codes," IEEE ITW, 2021.

[4] C. Luo, L. Liu, B. Kurkoski, under review.

Nonlinear SR Codes: Clipping and Quantization

General Nonlinear SR codes: apply a nonlinear function $Q(\cdot)$ to the codeword:



Q is 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 ϵ is the clipping threshold.

Q is quantization:

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

Q discrete: $\hat{z}_1, \ldots, \hat{z}_L$ quantization levels; found using K-means/Lloyd-Max

Irregular SR Code

- An **irregular nonlinear** SR code is

$$\mathbf{c} = \begin{bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \vdots \\ \mathbf{c}_K \end{bmatrix} = \begin{bmatrix} Q_1(\mathbf{z}_1) \\ Q_2(\mathbf{z}_2) \\ \vdots \\ Q_K(\mathbf{z}_K) \end{bmatrix}$$

  Nonlinear functions $Q_1, \ldots, Q_K$ are distinct.
  λ_i is a fraction.
  The length of $\mathbf{c}_i$ is $\lambda_i M$, i.e. λ_i is a degree distribution, $\sum_i \lambda_i = 1$
- In other words, apply Q_i to fraction λ_i of the code symbols.
- In addition, power normalization is performed: $Q_i(\mathbf{z}_i) \rightarrow \alpha_i Q_i(\mathbf{z}_i)$

OAMP for Nonlinear SR Codes

- γ: 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 $\gamma_1, \ldots, \gamma_K$ for $Q_1, \ldots, Q_K$.
- ϕ: same as standard SR codes

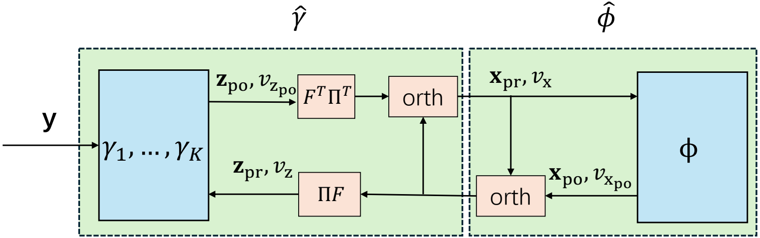


Figure 4: OAMP decoder of irregular SR codes

State Evolution Optimization for Irregular Quantization

The "gap" is :

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

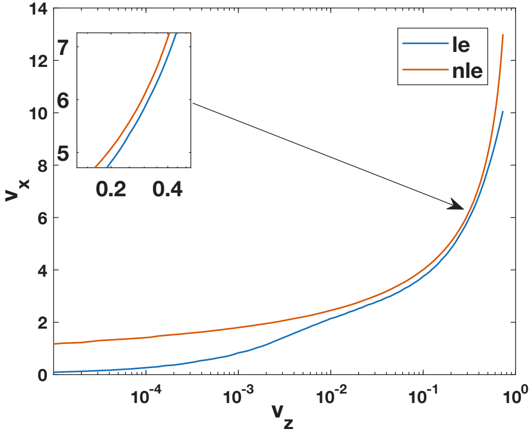
Optimization is to minimize $\mathcal{G}_{\text{irr}}(v_z, \boldsymbol{\lambda})$ over v_z and maximize over $\boldsymbol{\lambda}$:

$$\max_{\boldsymbol{\lambda}} \min_{v_z \in \mathcal{V}} \mathcal{G}_{\text{irr}}(v_z, \boldsymbol{\lambda}),$$

$$s.t. \sum_{k=1}^{K} \lambda_k = 1,$$

$$0 \leq \lambda_k \leq 1.$$

Showed that $\mathcal{G}_{\text{irr}}$ is a convex function.



SER Performance Comparisons

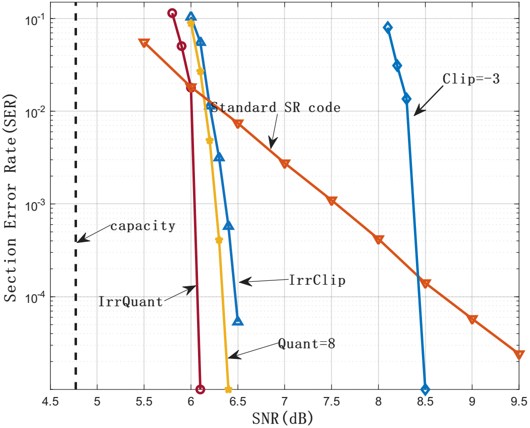


Figure 5: Comparisons between SER performance of regular quantized ($k = 8$) and irregular quantized SR codes ($k = 4$ and $k = 20$ combined), standard SR codes, regular clipped SR codes, irregular Clipped SR codes, section length $B = 64$, number of sections $L = 2048$, code rate $R = 1$, codeword length $M = 12288$, max iteration number=60.

Irregular Quantization vs Irregular Clip

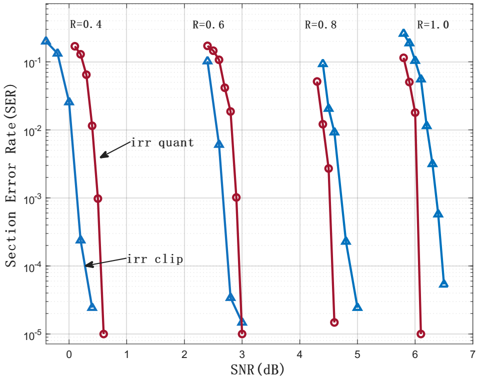
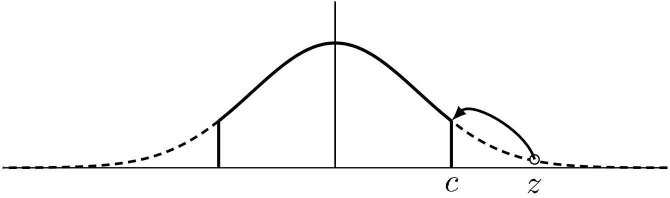


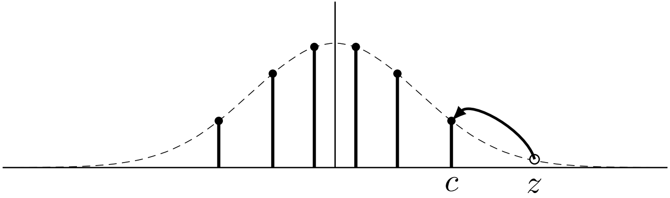
Figure 6: SERs of optimized irregular clipped SR codes (Triangles) vs. irregular quantized SR codes (Circles) with different code rates.

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

Rate	SNR	Levels	λ
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 α 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 $R = 1$? Hypothesis: $R = 1 \rightarrow$ more compression for this level of sparsity $\rightarrow$ AMP is less optimal (finite length) $\rightarrow$ quantization overcomes this suboptimality.

Conclusion

AMP methods performs optimal signal recovery of the problem:

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}$$

when $\mathbf{A}$ and $\mathbf{x}$ satisfy certain conditions. Presented recent results using AMP:

Random multiplexing

- Replace structured modulations (OFDM, OTFS) with a random unitary transform
- The equivalent channel is dense and statistically well behaved
- System is in the universality class where AMP detectors have predictable performance
- Numerical results show 2–3 dB performance gain over OFDM/OTFS/AFDM

Clipped/Quantized sparse regression codes

- These codes outperform conventional versions
- When rate is high $R \geq 0.8$, quantization outperforms clipping
- When rate is low $R \leq 0.6$ clipping outperforms quantization.