Unsourced Multiple Access (UMAC): IT & Coding

J.-F. Chamberland, K. Narayanan, and Y. Polyanskiy

ISIT, Melbourne, Australia, 2021

Tutorial outline

1. (YP) Why rethink MAC today?
2. (YP) Review of classical results on MAC
3. (YP) **New UMAC model. IT bounds**
4. (KN) Why standard solutions do not work for UMAC
5. (JFC) **UMAC codes from Compressed Sensing**
6. (KN) **UMAC codes from MAC codes**

Internet-of-Things: Machine-Type Communication

- 5G and 6G: largely bet on new application domains
- Machine-type communication (MTC): main driver of unit sales
- MTC's requirements:
  - huge # of devices
  - grant-free access
  - uncoordinated transmissions

Internet-of-Things: Machine-Type Communication

- 5G and 6G: largely bet on new application domains
- Machine-type communication (MTC): main driver of unit sales
- MTC's requirements:
  - huge # of devices
  - grant-free access
  - uncoordinated transmissions
  - chief: energy efficiency

Internet-of-Things: Machine-Type Communication

- 5G and 6G: largely bet on new application domains
- Machine-type communication (MTC): main driver of unit sales
- MTC's requirements:
  - huge # of devices
  - grant-free access
  - uncoordinated transmissions
  - chief: energy efficiency
- wireless sensor networks were expected to "change the world" every year since ≈1990. What's different now?

Internet-of-Things: Machine-Type Communication

- 5G and 6G: largely bet on new application domains
- Machine-type communication (MTC): main driver of unit sales
- MTC's requirements:
  - huge # of devices
  - grant-free access
  - uncoordinated transmissions
  - chief: energy efficiency
- wireless sensor networks were expected to "change the world" every year since ≈1990. What's different now? **Consumer interest**

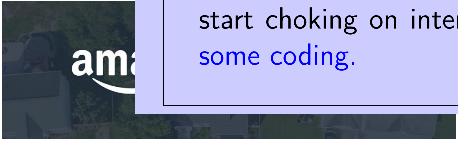
LoRa

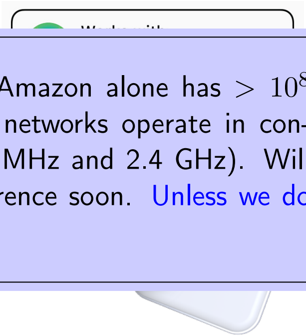
sigfox

Works with
Apple Find My

amazon sidewalk

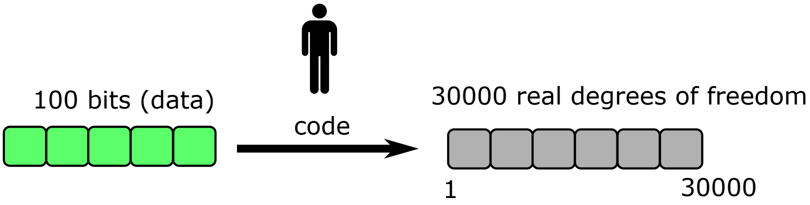
2021: multiple commercial LPWANs



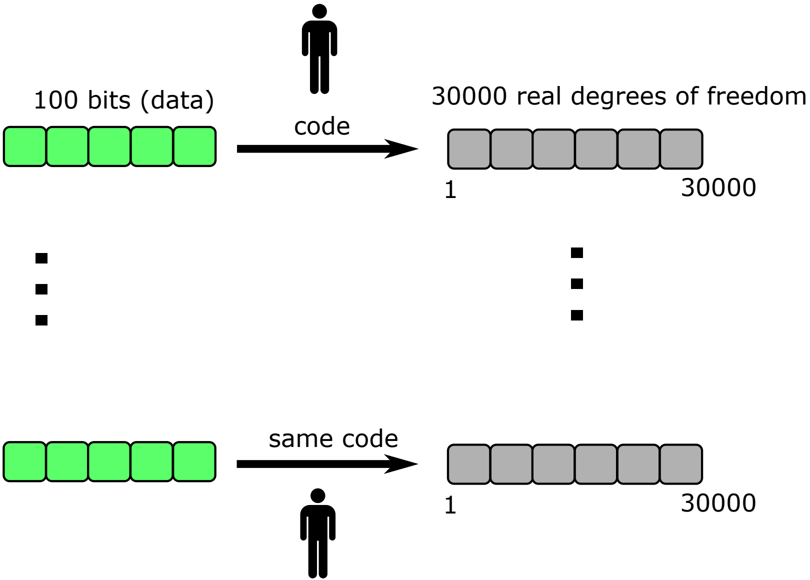


Why they need us? Amazon alone has $> 10^8$ devices already. These networks operate in congested ISM bands (900 MHz and 2.4 GHz). Will start choking on interference soon. Unless we do some coding.

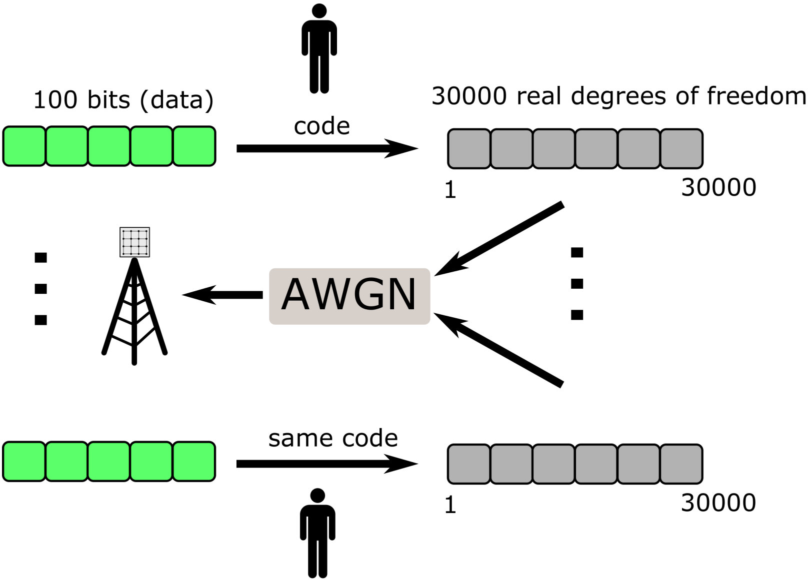
Preview: what this talk is about?



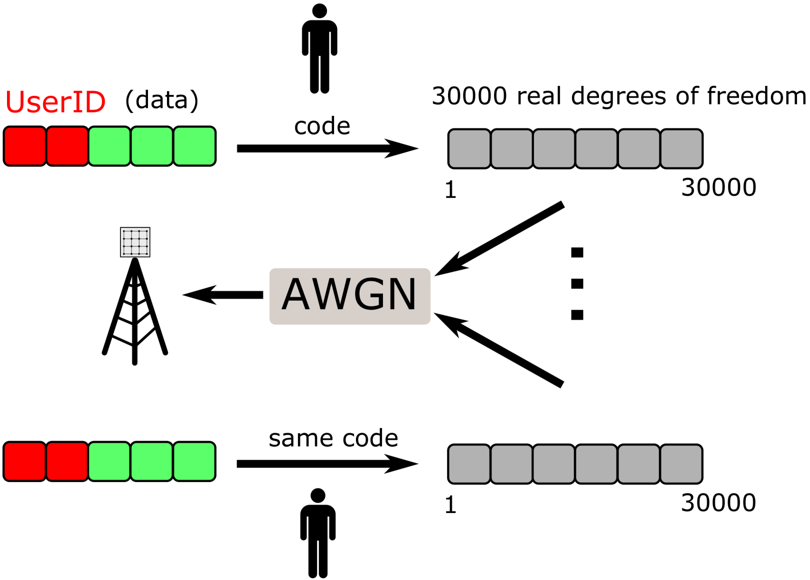
Preview: what this talk is about?

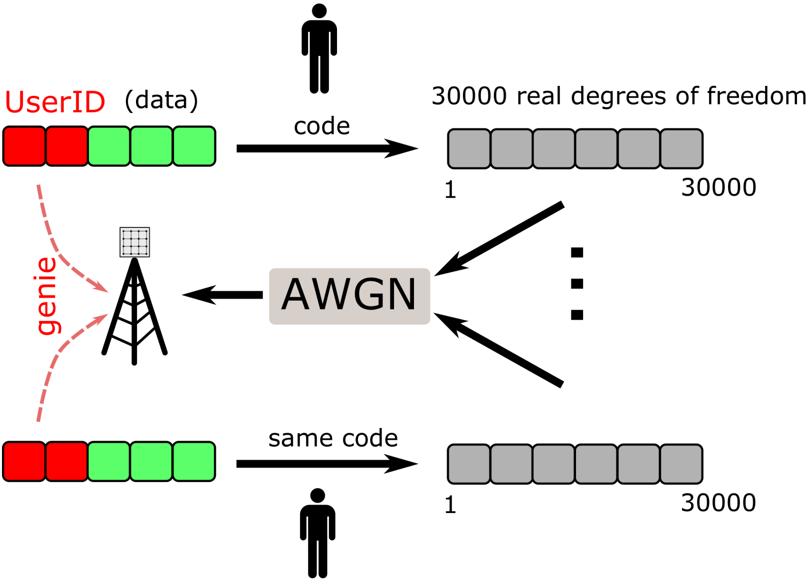


Preview: what this talk is about?

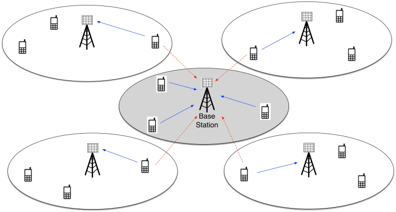


Random-Access (UMAC) vs Classical MAC



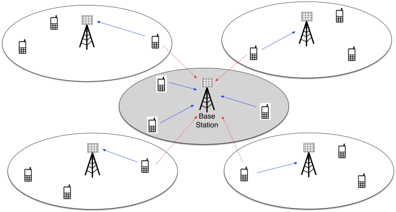

UserID
(data)
code
30000 real degrees of freedom
1
30000
genie
AWGN
same code
1
30000

Transmission costs: classical and new



- **Classical story:**
  - Moving k bits costs energy $E_b \times k$
  - Want to move bits faster (higher spectral efficiency ρ)? You pay more
  - Fundamentally minimal $E_b = N_0 \frac{2^\rho - 1}{\rho}$
  - MAC: Same tradeoff if there are $K > 1$ users
  - **. . . and orthogonalizing access is optimal**

Transmission costs: classical and new

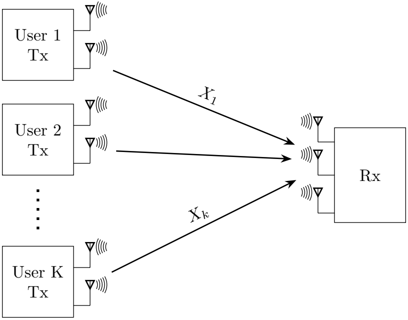


- **Classical story:**
  - Moving k bits costs energy $E_b \times k$
  - Want to move bits faster (higher spectral efficiency ρ)? You pay more
  - Fundamentally minimal $E_b = N_0 \frac{2^\rho - 1}{\rho}$
  - MAC: Same tradeoff if there are $K > 1$ users
  - **. . . and orthogonalizing access is optimal**

- **New story:** (this talk)
  - with $K \gg 1$ of very low-rate users, tradeoff changes (new problem)
  - Math: first-order phase transition
  - Engineering: orthogonalization is bad
  - Business: **free lunch** – adding more users costs nothing (no increase in space-time-frequency resources or energy)

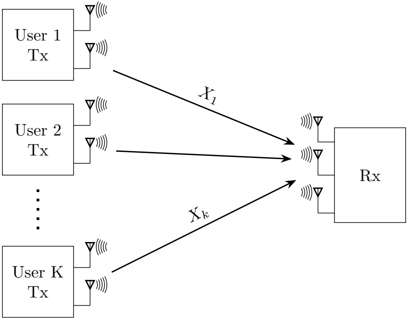
Classical multiple-access IT

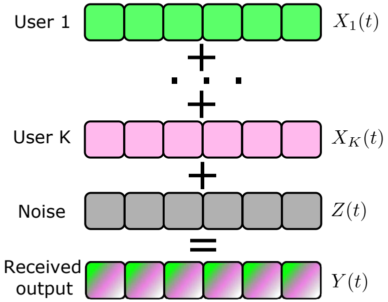
Gaussian MAC



$$Y(t) = X_1(t) + \cdots + X_K(t) + Z(t)$$

Gaussian MAC





$$Y(t) = X_1(t) + \cdots + X_K(t) + Z(t)$$

- Users send coded waveforms $X_j(t)$
- Additive Gaussian noise $Z(t)$
- Base station's job: estimate X_j from the knowledge of $Y(t)$

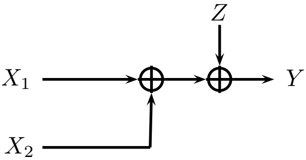
Tech note: synchronized block coding

2-user Gaussian MAC

$$Y = X_1 + X_2 + Z$$

$$Z \overset{iid}{\sim} \mathcal{N}(0,1)$$

$$\mathbb{E}[(X_1)^2] \leq P_1, \mathbb{E}[(X_2)^2] \leq P_2$$

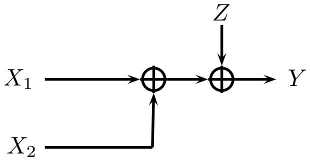


2-user Gaussian MAC

$$Y = X_1 + X_2 + Z$$

$$Z \stackrel{iid}{\sim} \mathcal{N}(0,1)$$

$$\mathbb{E}[(X_1)^2] \le P_1, \mathbb{E}[(X_2)^2] \le P_2$$



- Evaluating capacity region:

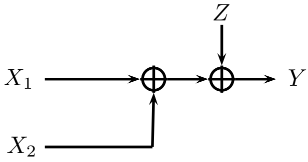
$$\begin{aligned} R_1 + R_2 &\le I(X_1, X_2; Y) \le \frac{1}{2}\log(1 + P_1 + P_2) \\ R_i &\le I(X_i; Y|X_{\hat{i}}) = I(X_i; X_i + Z) \le \frac{1}{2}\log(1 + P_i) \end{aligned}$$

2-user Gaussian MAC

$$Y = X_1 + X_2 + Z$$

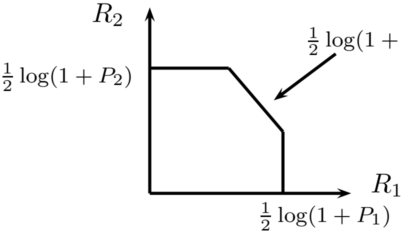
$$Z \overset{iid}{\sim} \mathcal{N}(0,1)$$

$$\mathbb{E}[(X_1)^2] \le P_1, \mathbb{E}[(X_2)^2] \le P_2$$



- Evaluating capacity region:

$$\begin{aligned} R_1 + R_2 &\le I(X_1, X_2; Y) \le \frac{1}{2}\log(1 + P_1 + P_2) \\ R_i &\le I(X_i; Y | X_{\hat{i}}) = I(X_i; X_i + Z) \le \frac{1}{2}\log(1 + P_i) \end{aligned}$$

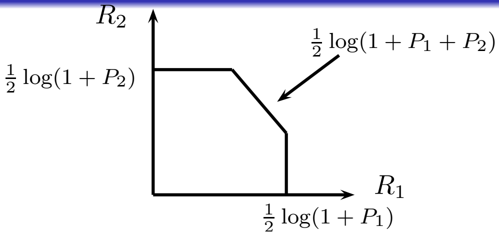


2-GMAC rates for FDMA

$$Y = X_1 + X_2 + Z$$

$$Z \overset{iid}{\sim} \mathcal{N}(0,1)$$

$$\mathbb{E}[(X_1)^2] \leq P_1, \mathbb{E}[(X_2)^2] \leq P_2$$



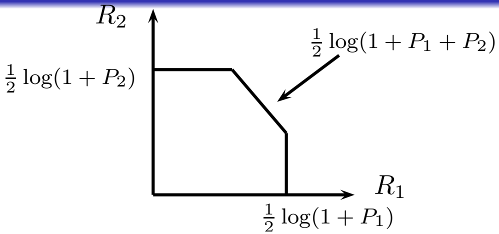
- Here is a FDMA:
  - Use Fourier transform to change n=time to n=frequency.
  - Partition block: $n = \lambda n + (1-\lambda)n$
  - User 1 sends in λn: $R_1 = \frac{\lambda}{2}\log(1+\frac{P_1}{\lambda})$
  - User 2 sends in $\bar{\lambda} n$: $R_2 = \frac{\bar{\lambda}}{2}\log(1+\frac{P_2}{\bar{\lambda}})$

2-GMAC rates for FDMA

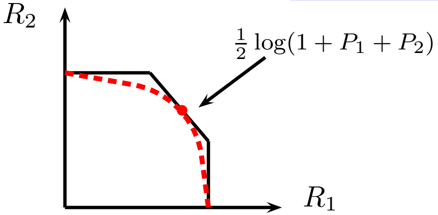
$$Y = X_1 + X_2 + Z$$

$$Z \overset{iid}{\sim} \mathcal{N}(0,1)$$

$$\mathbb{E}[(X_1)^2] \le P_1, \mathbb{E}[(X_2)^2] \le P_2$$



- Here is a FDMA:
  - Use Fourier transform to change n=time to n=frequency.
  - Partition block: $n = \lambda n + (1-\lambda)n$
  - User 1 sends in λn: $R_1 = \frac{\lambda}{2}\log(1+\frac{P_1}{\lambda})$
  - User 2 sends in $\bar{\lambda} n$: $R_2 = \frac{\bar{\lambda}}{2}\log(1+\frac{P_2}{\bar{\lambda}})$

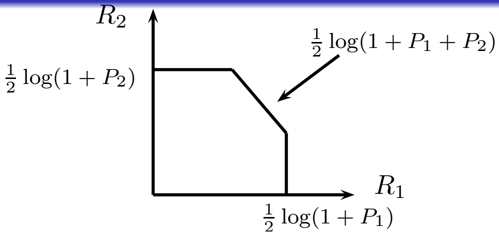


2-GMAC rates for FDMA

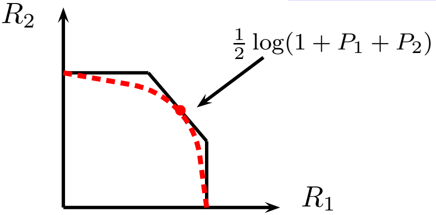
$$Y = X_1 + X_2 + Z$$

$$Z \overset{iid}{\sim} \mathcal{N}(0,1)$$

$$\mathbb{E}[(X_1)^2] \le P_1, \mathbb{E}[(X_2)^2] \le P_2$$



- Here is a FDMA:
  - Use Fourier transform to change n=time to n=frequency.
  - Partition block: $n = \lambda n + (1-\lambda)n$
  - User 1 sends in λn: $R_1 = \frac{\lambda}{2}\log(1+\frac{P_1}{\lambda})$
  - User 2 sends in $\bar{\lambda} n$: $R_2 = \frac{\bar{\lambda}}{2}\log(1+\frac{P_2}{\bar{\lambda}})$



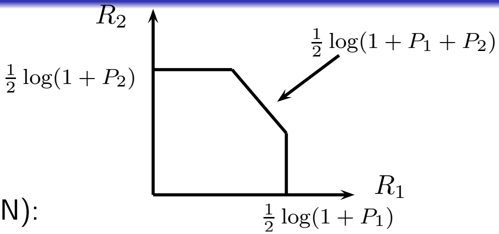
$\lambda^* = \frac{P_1}{P_1+P_2}$ achieves optimal sumrate

2-GMAC rates for TIN

$$Y = X_1 + X_2 + Z$$

$$Z \overset{iid}{\sim} \mathcal{N}(0,1)$$

$$\mathbb{E}[(X_1)^2] \le P_1, \mathbb{E}[(X_2)^2] \le P_2$$



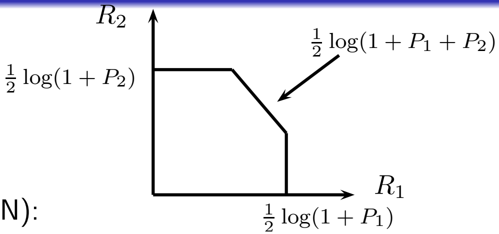
- Treat-interference-as-noise (TIN):
  - Each user treats the other as noise (single-user decoders)
  - Random coding ensures noise is Gaussian.
  - Rates: $R_1 = \frac{1}{2}\log(1+\frac{P_1}{1+P_2}), R_2 = \frac{1}{2}\log(1+\frac{P_2}{1+P_1})$

2-GMAC rates for TIN

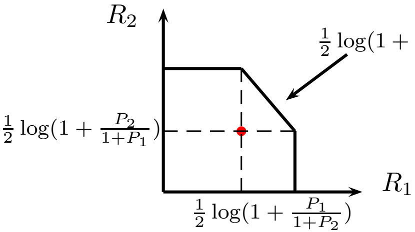
$$Y = X_1 + X_2 + Z$$

$$Z \overset{iid}{\sim} \mathcal{N}(0,1)$$

$$\mathbb{E}[(X_1)^2] \leq P_1, \mathbb{E}[(X_2)^2] \leq P_2$$



- Treat-interference-as-noise (TIN):
  - Each user treats the other as noise (single-user decoders)
  - Random coding ensures noise is Gaussian.
  - Rates: $R_1 = \frac{1}{2}\log(1+\frac{P_1}{1+P_2}), R_2 = \frac{1}{2}\log(1+\frac{P_2}{1+P_1})$



- TIN point can be inside/outside TDMA.

Spectral efficiency vs. $\frac{E_b}{N_0}$

- Spectral efficiency and energy-per-bit:

$$\rho \triangleq \frac{\text{total \# of data bits}}{\text{total real d.o.f.}}$$

$$\frac{E_b}{N_0} \triangleq \frac{\text{total energy spent}}{2 \times \text{total \# bits}} = \frac{nKP}{2nC_{sum}}$$

- Consider K equal-power users:

$$\rho = \frac{1}{2}\log(1+KP), \qquad \frac{E_b}{N_0} = \frac{KP}{\log(1+KP)}$$

- regardless of K: (and any sumrate-optimal arch)

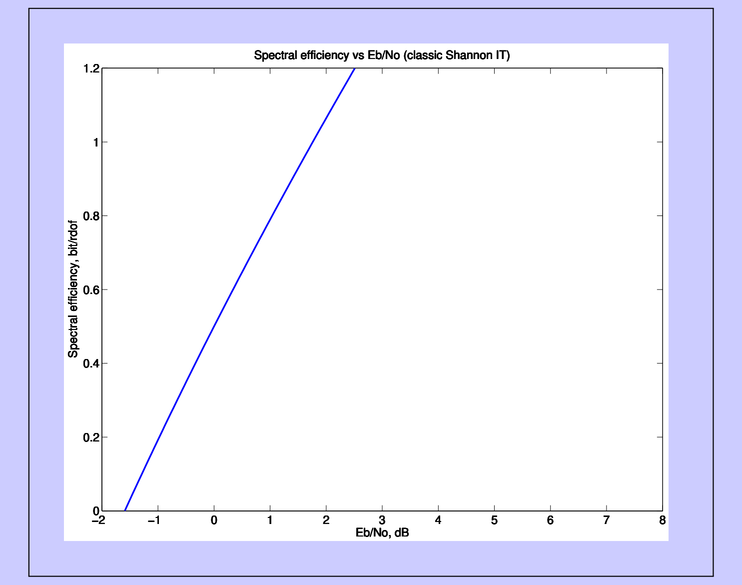
$$\frac{E_b}{N_0} = \frac{2^{2\rho}-1}{2\rho} \geq -1.59\ dB$$

Spectral efficiency vs. $\frac{E_b}{N_0}$

- Spectral efficiency and energy-per-bit:

- Cons

- rega



Spectral efficiency vs. $\frac{E_b}{N_0}$

- Spectral efficiency and energy-per-bit:

$$\rho \triangleq \frac{\text{total \# of data bits}}{\text{total real d.o.f.}}$$
$$\frac{E_b}{N_0} \triangleq \frac{\text{total energy spent}}{2 \times \text{total \# bits}} = \frac{nKP}{2nC_{sum}}$$

- Consider K equal-power users:

$$\rho = \frac{1}{2}\log(1+KP), \qquad \frac{E_b}{N_0} = \frac{KP}{\log(1+KP)}$$

- regardless of K: (and any sumrate-optimal arch)

$$\frac{E_b}{N_0} = \frac{2^{2\rho}-1}{2\rho} \geq -1.59\ dB$$

- Compare to TIN: $\rho = \frac{K}{2}\log_2(1+\frac{P}{1+(K-1)P}) \xrightarrow{K\to\infty} \frac{1}{2\ln 2}\frac{P_{tot}}{1+P_{tot}}$

Spectral efficiency vs. $\frac{E_b}{N_0}$

- Spectral efficiency and energy-per-bit:

$$\rho \triangleq \frac{\text{total \# of data bits}}{\text{total real d.o.f.}}$$

$$\frac{E_b}{N_0} \triangleq \frac{\text{total energy spent}}{2 \times \text{total \# bits}} = \frac{nKP}{2nC_{sum}}$$

- Consider K equal-power users:

$$\rho = \frac{1}{2}\log(1+KP), \qquad \frac{E_b}{N_0} = \frac{KP}{\log(1+KP)}$$

- regardless of K: (and any sumrate-optimal arch)

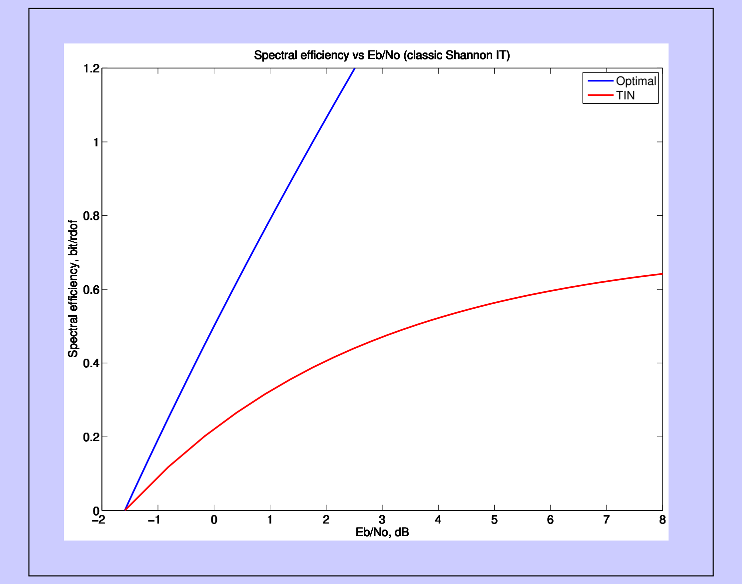
$$\frac{E_b}{N_0} = \frac{2^{2\rho}-1}{2\rho} \geq -1.59\ dB$$

- Compare to TIN: $\rho = \frac{K}{2}\log_2\left(1+\frac{P}{1+(K-1)P}\right) \xrightarrow{K\to\infty} \frac{1}{2\ln 2}\frac{P_{tot}}{1+P_{tot}}$

$$\rho = \frac{1}{2\ln 2}\frac{P_{tot}}{1+P_{tot}}, \qquad \frac{E_b}{N_0} = (1+P_{tot})\ln 2$$

Spectral efficiency vs. $\frac{E_b}{N_0}$

- Spectral efficiency and energy-per-bit:



- Cons

- rega

- Com

Spectral efficiency vs. $\frac{E_b}{N_0}$

- Spectral efficiency and energy-per-bit:

$$\rho \triangleq \frac{\text{total \# of data bits}}{\text{total real d.o.f.}}$$

$$\frac{E_b}{N_0} \triangleq \frac{\text{total energy spent}}{2 \times \text{total \# bits}} = \frac{nKP}{2nC_{sum}}$$

- Consider K equal-power users:

$$\rho = \frac{1}{2}\log(1+KP), \qquad \frac{E_b}{N_0} = \frac{KP}{\log(1+KP)}$$

- regardless of K: (and any sumrate-optimal arch)

$$\frac{E_b}{N_0} = \frac{2^{2\rho}-1}{2\rho} \geq -1.59\ dB$$

- Compare to TIN: $\rho = \frac{K}{2}\log_2(1+\frac{P}{1+(K-1)P}) \xrightarrow{K\to\infty} \frac{1}{2\ln 2}\frac{P_{tot}}{1+P_{tot}}$

$$\rho = \frac{1}{2\ln 2}\frac{P_{tot}}{1+P_{tot}}, \qquad \frac{E_b}{N_0} = (1+P_{tot})\ln 2$$

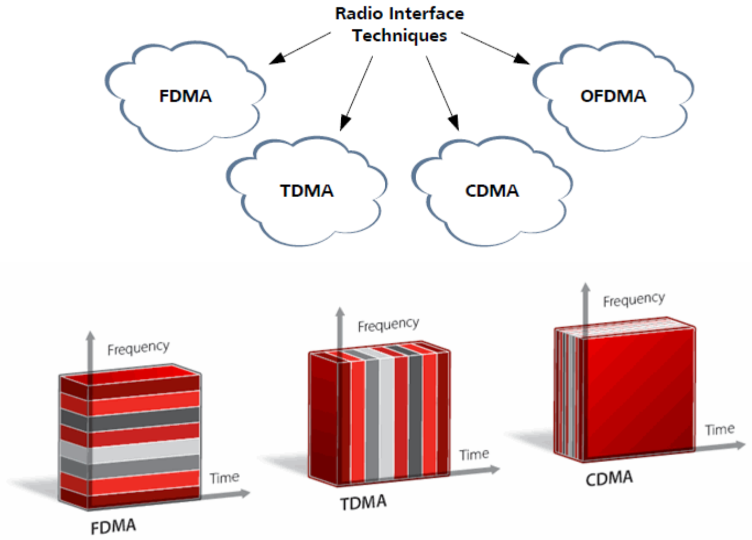
- **For TIN:** $\rho \leq \frac{1}{2\ln 2} = 0.72$ bit/rdof, E_b/N_0 optimal for low sp.eff.

Classical MAC: summary

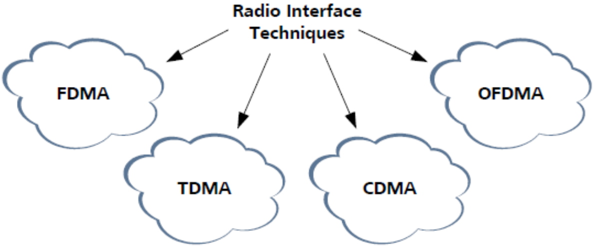
Principles:

- Tradeoff depends on spectral efficiency (aka total rate from all users), i.e. only on product $K \times \frac{\log M}{n}$.
- Orthogonal schemes are **optimal**
- TIN attains minimum $\frac{E_b}{N_0}$ when sp.eff. is low.

Classical (LTE) recipee: avoid multi-user interference!



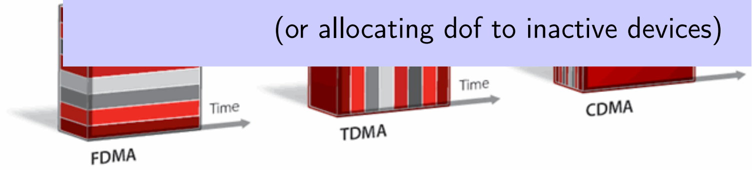
Classical (LTE) recipee: avoid multi-user interference!



These are all orthogonal schemes

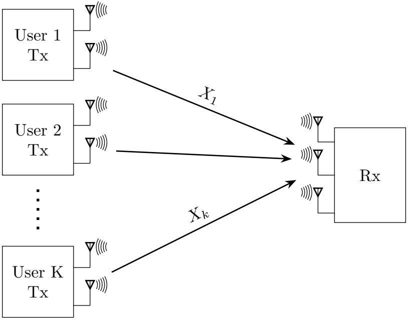
Key problem: they require coordination!

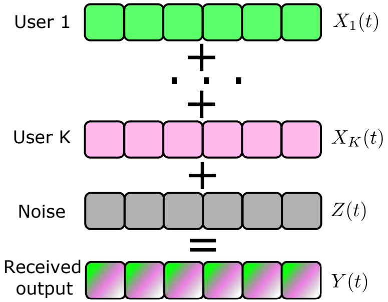
(or allocating dof to inactive devices)



New model: unsourced MAC

The classical model: K-user multiple-access channel

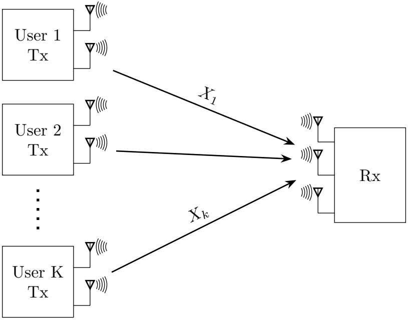


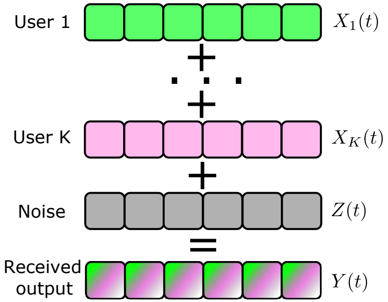


$$Y(t) = X_1(t) + \cdots + X_K(t) + Z(t)$$

- K users, each sends k bits
- **Classic:** $K = \text{small}$ and $k \gg 1$ (coordination cost ammortized)

The classical model: K-user multiple-access channel

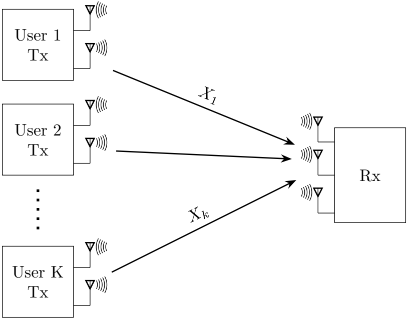


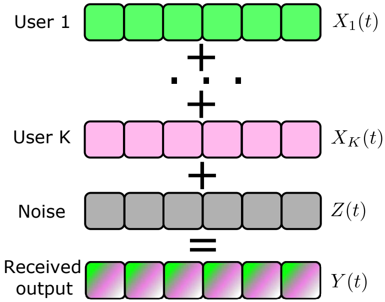


$$Y(t) = X_1(t) + \cdots + X_K(t) + Z(t)$$

- K users, each sends k bits
- Classic: $K = \text{small}$ and $k \gg 1$ (coordination cost ammortized)
- New 1: $k = \text{small}$ and $K \gg 11$

The classical model: K-user multiple-access channel



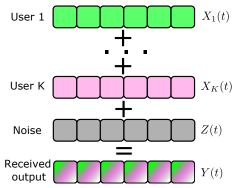


$$Y(t) = X_1(t) + \cdots + X_K(t) + Z(t)$$

- K users, each sends k bits
- **Classic:** $K = \text{small}$ and $k \gg 1$ (coordination cost ammortized)
- **New 1:** $k = \text{small}$ and $K \gg 11$
- **New 2:** Users are indistinguishable (unsourced)

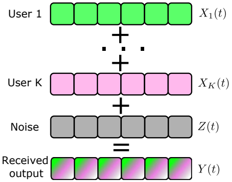
Concept of a UMAC code

- Users: select K_a messages $W_i \overset{iid}{\sim} \text{Uniform}[M]$
- Encoder f : maps W_i to codeword $f(W_i) \in \mathbb{R}^n$
- Channel: $Y = \sum_{i=1}^{K_a} f(W_i) + Z$
- Decoder g: inspects Y and produces a list $g(Y)$ of K_a messages



Concept of a UMAC code

- Users: select K_a messages $W_i \overset{iid}{\sim} \text{Uniform}[M]$
- Encoder f : maps W_i to codeword $f(W_i) \in \mathbb{R}^n$
- Channel: $Y = \sum_{i=1}^{K_a} f(W_i) + Z$
- Decoder g: inspects Y and produces a list $g(Y)$ of K_a messages



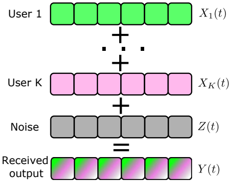
Definition

(f, g) form an (n, M, K_a, P, ϵ) UMAC code if both requirements hold:

- **(energy)**: for each $w \in [M]$: $\|f(w)\|^2 \leq nP$
- **(PUPE)**: for each $i \in [K_a]$: $\mathbb{P}[W_i \notin g(Y)] \leq \epsilon$

Concept of a UMAC code

- Users: select K_a messages $W_i \overset{iid}{\sim} \text{Uniform}[M]$
- Encoder f: maps W_i to codeword $f(W_i) \in \mathbb{R}^n$
- Channel: $Y = \sum_{i=1}^{K_a} f(W_i) + Z$
- Decoder g: inspects Y and produces a list $g(Y)$ of K_a messages



Definition

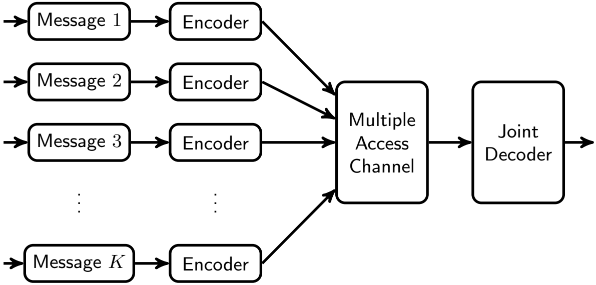
(f, g) form an (n, M, K_a, P, ϵ) UMAC code if both requirements hold:

- **(energy)**: for each $w \in [M]$: $\|f(w)\|^2 \le nP$
- **(PUPE)**: for each $i \in [K_a]$: $\mathbb{P}[W_i \notin g(Y)] \le \epsilon$

- Per-User Probability of Error $= \frac{1}{K_a} \sum_{i=1}^{K_a} \mathbb{P}[\text{User } i\text{th msg lost}]$
- Sometimes, the *message collision* is included in the error event:

$$\mathbb{P}[W_i \notin g(Y) \text{ or } \exists j \ne i : W_j = W_i] \le \epsilon$$

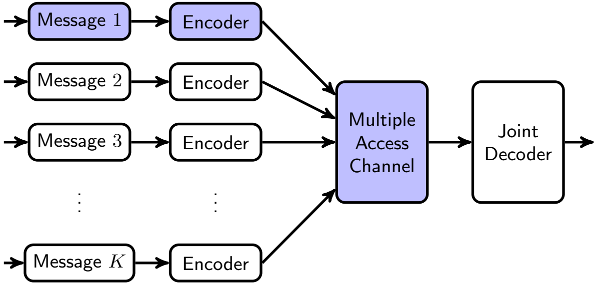
UMAC model explained



Characteristics of UMAC model

- K_a active devices, each with a $\log_2 M$-bit message
- Multiple access channel.

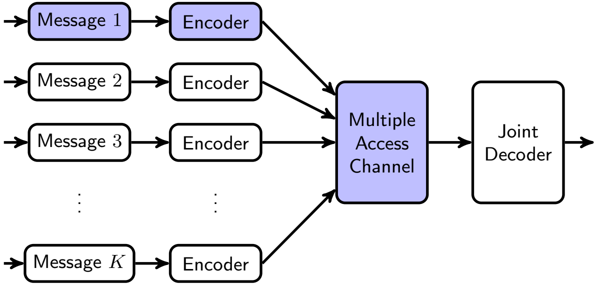
UMAC model explained



Characteristics of UMAC model

- K_a active devices, each with a $\log_2 M$-bit message
- Multiple access channel.
- Every user employs same encoder $f : [M] \to \mathbb{R}^n$
- Decoder only reconstructs an unordered list of messages.

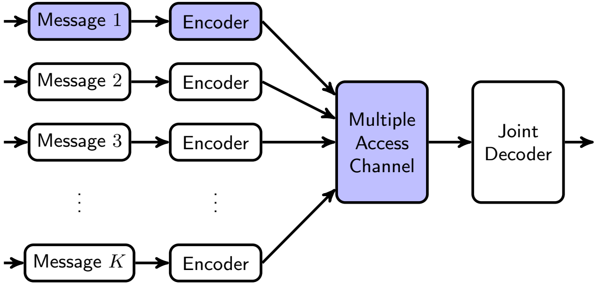
UMAC model explained



Characteristics of UMAC model

- K_a active devices, each with a $\log_2 M$-bit message
- Multiple access channel.
- Every user employs same encoder $f : [M] \to \mathbb{R}^n$
- Decoder only reconstructs an unordered list of messages.
- Goal: decoded list $\approx$ list of sent messages (fraction wrong $\leq$ PUPE)

UMAC model explained



Characteristics of UMAC model

- K_a active devices, each with a $\log_2 M$-bit message
- Multiple access channel.
- Every user employs same encoder $f : [M] \to \mathbb{R}^n$
- Decoder only reconstructs an unordered list of messages.
- Goal: decoded list $\approx$ list of sent messages (fraction wrong $\leq$ PUPE)
- Key: summarizes **the** main challenge of random-access

Achievability bound

Theorem (P.'2017[1])

For any (M, n, ϵ, K_a, P) *and any* $P' < P$ *there exists a UMAC code with*

$$PUPE \le p_0 + \sum_{t=1}^{K_a} \frac{t}{K_a} e^{-nE(t)},$$

where

$$p_0 = \frac{1}{M}\binom{K_a}{2} + K_a \mathbb{P}[\chi^2(n) > \tfrac{nP}{P'}]$$

$$E(t) = \max_{0 \le \rho, \rho_1 \le 1} -\rho\rho_1 t R_1 - \rho_1 R_2 + E_0(\rho, \rho_1, P')$$

$$R_1 = \frac{1}{n}\log\frac{M}{t!}, \quad R_2 = \frac{1}{n}\log\binom{K_a}{t}$$

$E_0 = \cdots$ *(complicated expression)*

[1] Polyanskiy, "A perspective on massive random-access", 2017

Achievability bound: preliminaries I

Probability of a $Z \sim \mathcal{N}(0, aI_n)$ to land in a ball "Gaussian ball" :

$$\mathbb{P}[\|Z+u\| < v] \le e^{-nE_{ball}} .$$

- from Chernoff bound:

$$\mathbb{P}[\|Z+u\| < v] \le e^{-\gamma v^2}\mathbb{E}\left[e^{\gamma\|Z+u\|^2}\right] \qquad \forall\gamma > 0 .$$

- By direct computation: $\mathbb{E}\left[e^{\gamma\|Z+u\|^2}\right] = \frac{e^{-\frac{\gamma\|u\|^2}{1+2a\gamma}}}{(1+2a\gamma)^{\frac{n}{2}}}$.
- Thus, $E_{ball} = \min_{\gamma>0} -\gamma(v^2 + \frac{\|u\|^2}{1+2a\gamma}) + \frac{n}{2}\ln(1+2a\gamma)$.

Achievability bound: preliminaries II

Probability of a union (Gallager's ρ-trick):

$$\mathbb{P}[\cup_j A_j] \le \left(\sum_j \mathbb{P}[A_j]\right)^\rho \qquad \forall 0 < \rho \le 1$$

- Proof is simple: From union bound

$$\mathbb{P}[\cup_j A_j] \le \min\left(\sum_j \mathbb{P}[A_j], 1\right)$$

Now use the fact $\min(x, 1) \le x^\rho$.

Achievability bound: preliminaries II

Probability of a union (Gallager's ρ-trick):

$$\mathbb{P}[\cup_j A_j] \le \left(\sum_j \mathbb{P}[A_j]\right)^\rho \qquad \forall 0 < \rho \le 1$$

- Proof is simple: From union bound

$$\mathbb{P}[\cup_j A_j] \le \min\left(\sum_j \mathbb{P}[A_j], 1\right)$$

  Now use the fact $\min(x,1) \le x^\rho$.
- In applications one usually finds some good random variable V such that

$$\mathbb{P}[A_j|V] \le e^{-nE(V)}\,,$$

  for some computable $E(V)$. And then from the ρ-trick:

$$\mathbb{P}[\cup_{j=1}^m A_j] \le m^\rho \mathbb{E}\left[e^{-n\rho E(V)}\right]$$

Random-coding achievability: Proof I

- Codebook generation:

$$c_i \sim \mathcal{N}(0, P')^{\otimes n}, , \qquad i = 1, \ldots, M \,.$$

- Why generate with power $P' < P$? Because we want to satisfy strict power constraint:

$$\|c_i\|^2 \leq nP \,.$$

Random-coding achievability: Proof I

- Codebook generation:

$$c_i \sim \mathcal{N}(0, P')^{\otimes n}, , \qquad i = 1, \ldots, M\,.$$

- Why generate with power $P' < P$? Because we want to satisfy strict power constraint:

$$\|c_i\|^2 \le nP\,.$$

- So each user before transmitting c_i makes sure that $\|c_i\|^2 \le nP$, otherwise transmits 0.
- With probability $\le p_0$ then all K_a users selected good and distinct codewords:

$$p_0 = \frac{1}{M}\binom{K_a}{2} + K_a \mathbb{P}[\|c_1\|^2 > nP]$$

- Conditioning on this event, and from symmetry we can assume that $c_1, \ldots, c_{K_a}$ were transmitted.
- Proceed to discussing decoder...

Random-coding achievability: Proof II

- Decoder receives
$$Y = c_1 + \cdots + c_{K_a} + Z$$
his job is to recover a subset $S \subset [M]$ of size K_a of those codewords that he believes were sent.
- Define sum-codewords $c(S) \triangleq \sum_{i \in S} c_i$
- We will analyze maximum likelihood decoder:
$$\hat{S} = \arg\min_{S} \|c(S) - Y\|\,.$$
- **Note:** This decoder is not optimal. Why? Because our figure of merit is not to decode all c/w correctly, but rather to decode each one with high probability. (Similar: ML is not optimal for minimizing BER)
- Note that selecting $\hat{S}$ we incur
$$\mathsf{PUPE} = \frac{1}{K_a} |[K_a] \setminus \hat{S}|\,.$$
- So $\{t\text{–misdecoded}\} = \{|[K_a] \setminus \hat{S}| = t\}$.

$$Y = c_1 + \cdots + c_{K_a} + Z$$

- Goal: bound $\mathbb{P}[t\text{–misdecoded}] \le e^{-nE(t)}$.

$$Y = c_1 + \cdots + c_{K_a} + Z$$

- Goal: bound $\mathbb{P}[t\text{–misdecoded}] \le e^{-nE(t)}$.
- Note: t–misdecoded $\iff$ some subset $S_0 \subset [K_a]$ of messages was replaced with $S_0' \subset \{K_a + 1, \ldots, M\}$ and $|S_0| = |S_0'| = t$.

  Denote $F(S_0, S_0') \triangleq \{\|c(S_0) - c(S_0') + Z\| \le \|Z\|\}$. Then:

$$Y = c_1 + \cdots + c_{K_a} + Z$$

- Goal: bound $\mathbb{P}[t\text{–misdecoded}] \le e^{-nE(t)}$.
- Note: t–misdecoded $\iff$ some subset $S_0 \subset [K_a]$ of messages was replaced with $S_0' \subset \{K_a+1,\ldots,M\}$ and $|S_0| = |S_0'| = t$.

Denote $F(S_0, S_0') \triangleq \{\|c(S_0) - c(S_0') + Z\| \le \|Z\|\}$. Then:

$$\mathbb{P}[t\text{–misdecoded}] \le \mathbb{P}\left[\bigcup_{S_0 \in \binom{K_a}{t}} \bigcup_{S_0' \in \binom{M-K_a}{t}} F(S_0, S_0')\right]$$

$$Y = c_1 + \cdots + c_{K_a} + Z$$

- Goal: bound $\mathbb{P}[t\text{–misdecoded}] \le e^{-nE(t)}$.
- Note: t–misdecoded $\iff$ some subset $S_0 \subset [K_a]$ of messages was replaced with $S_0' \subset \{K_a+1, \ldots, M\}$ and $|S_0| = |S_0'| = t$.

  Denote $F(S_0, S_0') \triangleq \{\|c(S_0) - c(S_0') + Z\| \le \|Z\|\}$. Then:

$$\mathbb{P}[t\text{–misdecoded}] \le \mathbb{P}\left[\bigcup_{S_0 \in \binom{K_a}{t}} \bigcup_{S_0' \in \binom{M-K_a}{t}} F(S_0, S_0')\right]$$

- Note that $\mathbb{P}[F(S_0, S_0') | c(S_0), Z] \le e^{-nE_{ball}}$ (Gaussian ball)

$$Y = c_1 + \cdots + c_{K_a} + Z$$

- Goal: bound $\mathbb{P}[t\text{–misdecoded}] \le e^{-nE(t)}$.
- Note: t–misdecoded $\iff$ some subset $S_0 \subset [K_a]$ of messages was replaced with $S_0' \subset \{K_a+1, \dots, M\}$ and $|S_0| = |S_0'| = t$.

  Denote $F(S_0, S_0') \triangleq \{\|c(S_0) - c(S_0') + Z\| \le \|Z\|\}$. Then:

$$\mathbb{P}[t\text{–misdecoded}] \le \mathbb{P}\left[\bigcup_{S_0 \in \binom{K_a}{t}} \bigcup_{S_0' \in \binom{M-K_a}{t}} F(S_0, S_0')\right]$$

- Note that $\mathbb{P}[F(S_0, S_0') | c(S_0), Z] \le e^{-nE_{ball}}$ (Gaussian ball)
- So we use Gallager's ρ-trick twice:
  - Let $F(S_0) = \cup_{S_0'} F(S_0, S_0')$ and bound $\mathbb{P}[F(S_0)|c(S_0), Z] \le \binom{M-K_a}{t}^{\rho} e^{-n\rho E_{ball}} \triangleq e^{-n\tilde{E}(c(S_0),Z)}$
  - Then bound $\mathbb{P}[\cup_{S_0} F(S_0)] \le \binom{K_a}{t}^{\rho_1} \mathbb{E}[e^{-n\rho_1 \tilde{E}(c(S_0),Z)}]$

$$Y = c_1 + \cdots + c_{K_a} + Z$$

- Goal: bound $\mathbb{P}[t\text{–misdecoded}] \le e^{-nE(t)}$.
- Note: t–misdecoded $\iff$ some subset $S_0 \subset [K_a]$ of messages was replaced with $S_0' \subset \{K_a + 1, \ldots, M\}$ and $|S_0| = |S_0'| = t$.

  Denote $F(S_0, S_0') \triangleq \{\|c(S_0) - c(S_0') + Z\| \le \|Z\|\}$. Then:

$$\mathbb{P}[t\text{–misdecoded}] \le \mathbb{P}\left[\bigcup_{S_0 \in \binom{K_a}{t}} \bigcup_{S_0' \in \binom{M-K_a}{t}} F(S_0, S_0')\right]$$

- Note that $\mathbb{P}[F(S_0, S_0')|c(S_0), Z] \le e^{-nE_{ball}}$ (Gaussian ball)
- So we use Gallager's ρ-trick twice:
  - Let $F(S_0) = \cup_{S_0'} F(S_0, S_0')$ and bound $\mathbb{P}[F(S_0)|c(S_0), Z] \le \binom{M-K_a}{t}^\rho e^{-n\rho E_{ball}} \triangleq e^{-n\tilde{E}(c(S_0),Z)}$
  - Then bound $\mathbb{P}[\cup_{S_0} F(S_0)] \le \binom{K_a}{t}^{\rho_1} \mathbb{E}[e^{-n\rho_1 \tilde{E}(c(S_0),Z)}]$
- $\Rightarrow E(t) = \max_{\rho,\rho_1} -\rho\rho_1 t R_1 - \rho_1 R_2 + E_0(\rho, \rho_1)$

Random-coding: comparison to Classical MAC

$$\mathbb{P}\left[\bigcup_{S_0 \in \binom{K_a}{t}} \bigcup_{S_0' \in \binom{M-K_a}{t}} F(S_0, S_0')\right]$$

- S_0 selects those t users were unlucky (got their messages misdecoded)
- S_0' selects the wrong message for the unlucky users

Random-coding: comparison to Classical MAC

$$\mathbb{P}\left[\bigcup_{S_0 \in \binom{K_a}{t}} \bigcup_{S_0' \in \binom{M-K_a}{t}} F(S_0, S_0')\right]$$

- S_0 selects those t users were unlucky (got their messages misdecoded)
- S_0' selects the wrong message for the unlucky users
- In classical MAC we also have $2^K - 1$ different error-events indexed by $S_0 \subset [K]$ – misdecoded users. And

$$\mathbb{P}[F(S_0)] \le e^{n(\sum_{i \in S_0} R_i - \hat{I}(X_{S_0}; Y | X_{S_0^c})},$$

where $\hat{I}$ is the empirical mutual info.

Random-coding: comparison to Classical MAC

$$\mathbb{P}\left[\bigcup_{S_0\in\binom{K_a}{t}}\ \bigcup_{S_0'\in\binom{M-K_a}{t}} F(S_0,S_0')\right]$$

- S_0 selects those t users were unlucky (got their messages misdecoded)
- S_0' selects the wrong message for the unlucky users
- In classical MAC we also have 2^K-1 different error-events indexed by $S_0\subset[K]$ – misdecoded users. And

$$\mathbb{P}[F(S_0)] \le e^{n(\sum_{i\in S_0} R_i - \hat{I}(X_{S_0};Y|X_{S_0^c})}\,,$$

  where $\hat{I}$ is the empirical mutual info.
- Asymptotically: $\hat{I}=I$ and thus $\mathbb{P}[\cup_{S_0}F(S_0)]\to 0$ whenever

$$\sum_{i\in S_0} R_i < I(X_{S_0};Y|X_{S_0^c}) \qquad \forall S_0\subset[K]\,.$$

- The parallel with our bound should be clear.

Converse bound

Theorem

Every (n, K_a, M, P) *UMAC code with PUPE* $\leq \epsilon$ *must satisfy both:*

$$nP \geq \left(Q^{-1}(\frac{K_a}{M}) + Q^{-1}(\epsilon)\right)^2$$

$$\frac{n}{2}\log(1+K_aP) \geq \log\binom{M}{K_a} - K_a(\epsilon\log\frac{Me}{\epsilon K_a} + h(\epsilon))$$

$$= K_a\left((1-\epsilon)\log\frac{eM}{K_a} + 2\epsilon\log\epsilon + \bar{\epsilon}\log\bar{\epsilon} + O(\frac{1}{M})\right)$$

- Here: $Q(x) \triangleq \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} dy$, $h(\epsilon) = \epsilon\log\frac{1}{\epsilon} + \bar{\epsilon}\log\frac{1}{\bar{\epsilon}}$, $\bar{\epsilon} = 1-\epsilon$.
- First bound: almost independent of K_a.
- Second bound: compares sum-capacity with rate-distortion function.

Converse: proof I

- In [PPV'11][2]it was shown that any *single user* channel code over the AWGN with parameters (n, M, P) and BLER ϵ must satisfy

$$nP \geq \left(Q^{-1}(\frac{1}{M}) + Q^{-1}(\epsilon) \right)^2$$

[1] Polyanskiy, Poor, Verdú "Minimum energy to send k bits through the Gaussian channel with and without feedback," 2011

Converse: proof I

- In [PPV'11][2] it was shown that any *single user* channel code over the AWGN with parameters (n, M, P) and BLER ϵ must satisfy

$$nP \geq \left(Q^{-1}(\frac{1}{M}) + Q^{-1}(\epsilon)\right)^2$$

- It is easy to extend their argument to show that list-L decodable codes must satisfy

$$nP \geq \left(Q^{-1}(\frac{L}{M}) + Q^{-1}(\epsilon)\right)^2$$

[1] Polyanskiy, Poor, Verdú "Minimum energy to send k bits through the Gaussian channel with and without feedback," 2011

Converse: proof I

- In [PPV'11][2] it was shown that any *single user* channel code over the AWGN with parameters (n, M, P) and BLER ϵ must satisfy

$$nP \geq \left(Q^{-1}\left(\frac{1}{M}\right) + Q^{-1}(\epsilon)\right)^2$$

- It is easy to extend their argument to show that list-L decodable codes must satisfy

$$nP \geq \left(Q^{-1}\left(\frac{L}{M}\right) + Q^{-1}(\epsilon)\right)^2$$

- UMAC code (n, K_a, M, P) can be used as a list-K_a decodable *single user* code: add $(K_a - 1)$ dummy ones at the receiver and run UMAC decoder.

[1] Polyanskiy, Poor, Verdú "Minimum energy to send k bits through the Gaussian channel with and without feedback," 2011

Converse: proof I

- In [PPV'11][2] it was shown that any *single user* channel code over the AWGN with parameters (n, M, P) and BLER ϵ must satisfy

$$nP \geq \left(Q^{-1}(\frac{1}{M}) + Q^{-1}(\epsilon)\right)^2$$

- It is easy to extend their argument to show that list-L decodable codes must satisfy

$$nP \geq \left(Q^{-1}(\frac{L}{M}) + Q^{-1}(\epsilon)\right)^2$$

- UMAC code (n, K_a, M, P) can be used as a list-K_a decodable *single user* code: add $(K_a - 1)$ dummy ones at the receiver and run UMAC decoder. $\Rightarrow$ the bound above applies with $L = K_a$.

[1] Polyanskiy, Poor, Verdú "Minimum energy to send k bits through the Gaussian channel with and without feedback," 2011

Converse: proof II

- Define vector $U \in \{0,1\}^M$ with $U_i = 1$ iff some $W_j = i$. Similarly $\hat{U}$ is the vector output by decoder.
- $\|\cdot\|$ and $d(\cdot,\cdot)$ denote Hamming weight and Hamming distance, resp.

Converse: proof II

- Define vector $U \in \{0,1\}^M$ with $U_i = 1$ iff some $W_j = i$. Similarly $\hat{U}$ is the vector output by decoder.
- $\|\cdot\|$ and $d(\cdot,\cdot)$ denote Hamming weight and Hamming distance, resp.
- Condition on no message collisions from now on.
  $\Rightarrow U \sim \mathrm{Uniform}[\binom{M}{K_a}]$
- ... and

$$\mathbb{E}[d(U,\hat{U})] \leq 2K_a\epsilon \qquad (*)$$

Converse: proof II

- Define vector $U \in \{0,1\}^M$ with $U_i = 1$ iff some $W_j = i$.
  Similarly $\hat{U}$ is the vector output by decoder.
- $\|\cdot\|$ and $d(\cdot,\cdot)$ denote Hamming weight and Hamming distance, resp.
- Condition on no message collisions from now on.
  $\Rightarrow U \sim \mathrm{Uniform}[\binom{M}{K_a}]$
- ... and

$$\mathbb{E}[d(U,\hat{U})] \le 2K_a\epsilon \qquad (*)$$

- Overall we have the Markov chain:

$$U \to (X_1, \ldots, X_{K_a}) \to Y \to \hat{U}\,.$$

Converse: proof II

- Define vector $U \in \{0,1\}^M$ with $U_i = 1$ iff some $W_j = i$. Similarly $\hat{U}$ is the vector output by decoder.
- $\|\cdot\|$ and $d(\cdot,\cdot)$ denote Hamming weight and Hamming distance, resp.
- Condition on no message collisions from now on.
  $\Rightarrow U \sim \mathrm{Uniform}[\binom{M}{K_a}]$
- ... and

$$\mathbb{E}[d(U,\hat{U})] \le 2K_a\epsilon \qquad (*)$$

- Overall we have the Markov chain:

$$U \to (X_1,\ldots,X_{K_a}) \to Y \to \hat{U}\,.$$

So we can apply capacity vs rate-distortion method:

$$R(\epsilon) \le I(U;\hat{U}) \le I(X_1^{K_a};Y) \le \frac{n}{2}\log(1+PK_a)\,.$$

Converse: proof II

- Define vector $U \in \{0,1\}^M$ with $U_i = 1$ iff some $W_j = i$. Similarly $\hat{U}$ is the vector output by decoder.
- $\|\cdot\|$ and $d(\cdot,\cdot)$ denote Hamming weight and Hamming distance, resp.
- Condition on no message collisions from now on. $\Rightarrow U \sim \mathrm{Uniform}[\binom{M}{K_a}]$
- ... and

$$\mathbb{E}[d(U,\hat{U})] \le 2K_a\epsilon \qquad (*)$$

- Overall we have the Markov chain:

$$U \to (X_1,\ldots,X_{K_a}) \to Y \to \hat{U}\,.$$

So we can apply capacity vs rate-distortion method:

$$R(\epsilon) \le I(U;\hat{U}) \le I(X_1^{K_a};Y) \le \frac{n}{2}\log(1+PK_a)\,.$$

- Final step: compute $R(\epsilon) \triangleq \min\{I(U;\hat{U}) : (*)\text{–holds}\}$

Problem

Fix $w \in [0, m]$ and consider a fixed vector b and a random vector A on Hamming sphere of radius w in $\{0,1\}^m$, i.e. $\|b\| = \|A\| = w$. Find $\max\{H(A) : \mathbb{E}[d(A,b)] \le 2t\}$.

- WLOG $b = (\underbrace{1, \dots, 1}_{w}, \underbrace{0, \dots, 0}_{m-w})$
- By averaging over permutations the problem reduces to maximization over distribution of $S = \sum_{i=1}^{w} A_i$:

$$\max H(A) = \max \left\{ \mathbb{E}\left[\log \binom{w}{S} \binom{m-w}{S} \right] : \mathbb{E}[S] \le t \right\}$$

Problem

Fix $w \in [0, m]$ and consider a fixed vector b and a random vector A on Hamming sphere of radius w in $\{0,1\}^m$, i.e. $\|b\| = \|A\| = w$. Find $\max\{H(A) : \mathbb{E}[d(A,b)] \le 2t\}$.

- WLOG $b = (\underbrace{1, \dots, 1}_{w}, \underbrace{0, \dots, 0}_{m-w})$
- By averaging over permutations the problem reduces to maximization over distribution of $S = \sum_{i=1}^{w} A_i$:

$$\max H(A) = \max \left\{ \mathbb{E}\left[\log \binom{w}{S} \binom{m-w}{S} \right] : \mathbb{E}[S] \le t \right\}$$

  - Bound via Stirling: $\log \binom{m-w}{S} \le \log \frac{m^S}{S!} \le S \log m - S \log \frac{S}{e}$

Converse: proof III

Problem

Fix $w \in [0, m]$ and consider a fixed vector b and a random vector A on Hamming sphere of radius w in $\{0,1\}^m$, i.e. $\|b\| = \|A\| = w$. Find $\max\{H(A) : \mathbb{E}[d(A,b)] \le 2t\}$.

- WLOG $b = (\underbrace{1, \ldots, 1}_{w}, \underbrace{0, \ldots, 0}_{m-w})$
- By averaging over permutations the problem reduces to maximization over distribution of $S = \sum_{i=1}^{w} A_i$:

$$\max H(A) = \max \left\{ \mathbb{E}\left[\log \binom{w}{S} \binom{m-w}{S} \right] : \mathbb{E}[S] \le t \right\}$$

  - Bound via Stirling: $\log \binom{m-w}{S} \le \log \frac{m^S}{S!} \le S \log m - S \log \frac{S}{e}$
  - From Jensen: $\mathbb{E}[S \log S] \ge t \log t$.

Converse: proof III

Problem

Fix $w \in [0, m]$ and consider a fixed vector b and a random vector A on Hamming sphere of radius w in $\{0,1\}^m$, i.e. $\|b\| = \|A\| = w$. Find $\max\{H(A) : \mathbb{E}[d(A,b)] \le 2t\}$.

- WLOG $b = (\underbrace{1, \dots, 1}_{w}, \underbrace{0, \dots, 0}_{m-w})$
- By averaging over permutations the problem reduces to maximization over distribution of $S = \sum_{i=1}^{w} A_i$:

$$\max H(A) = \max\left\{\mathbb{E}\left[\log\binom{w}{S}\binom{m-w}{S}\right] : \mathbb{E}[S] \le t\right\}$$

  - Bound via Stirling: $\log\binom{m-w}{S} \le \log\frac{m^S}{S!} \le S\log m - S\log\frac{S}{e}$
  - From Jensen: $\mathbb{E}[S\log S] \ge t\log t$.
  - $\max\mathbb{E}\left[\log\binom{w}{S}\right] = wh(t/w)$ (max-entropy subject to $\mathbb{E}[S] \le t$)

Converse: proof III

Problem

Fix $w \in [0, m]$ and consider a fixed vector b and a random vector A on Hamming sphere of radius w in $\{0,1\}^m$, i.e. $\|b\| = \|A\| = w$. Find $\max\{H(A) : \mathbb{E}[d(A,b)] \le 2t\}$.

- WLOG $b = (\underbrace{1,\ldots,1}_{w}, \underbrace{0,\ldots,0}_{m-w})$
- By averaging over permutations the problem reduces to maximization over distribution of $S = \sum_{i=1}^{w} A_i$:

$$\max H(A) = \max\left\{\mathbb{E}\left[\log\binom{w}{S}\binom{m-w}{S}\right] : \mathbb{E}[S] \le t\right\}$$

  - Bound via Stirling: $\log\binom{m-w}{S} \le \log\frac{m^S}{S!} \le S\log m - S\log\frac{S}{e}$
  - From Jensen: $\mathbb{E}[S\log S] \ge t\log t$.
  - $\max \mathbb{E}\left[\log\binom{w}{S}\right] = wh(t/w)$ (max-entropy subject to $\mathbb{E}[S] \le t$)
- Overall: $\max H(A) \le t\log\frac{em}{t} + wh(\frac{t}{w})$

Converse: proof IV

Problem (Strange rate-distortion problem[3])

Find $R(\epsilon) \triangleq \min I(U;\hat{U})$ *where* $U \sim \mathrm{Uniform}[\binom{M}{K_a}]$ *and*

$$\mathbb{E}[d(U,\hat{U})] \leq 2K_a\epsilon$$

[3] A very similar problem was considered in: Reeves-Gastpar, "Approximate Sparsity Pattern Recovery: Information-Theoretic Lower Bounds," 2013

Converse: proof IV

Problem (Strange rate-distortion problem[3])

Find $R(\epsilon) \triangleq \min I(U;\hat{U})$ *where* $U \sim \mathrm{Uniform}[\binom{M}{K_a}]$ *and*

$$\mathbb{E}[d(U,\hat{U})] \le 2K_a\epsilon$$

- $I(U;\hat{U}) = H(U) - H(U|\hat{U})$

[3] A very similar problem was considered in: Reeves-Gastpar, "Approximate Sparsity Pattern Recovery: Information-Theoretic Lower Bounds," 2013

Problem (Strange rate-distortion problem[3])

Find $R(\epsilon) \triangleq \min I(U;\hat{U})$ *where* $U \sim \mathrm{Uniform}[\binom{M}{K_a}]$ *and*

$$\mathbb{E}[d(U,\hat{U})] \le 2K_a\epsilon$$

- $I(U;\hat{U}) = H(U) - H(U|\hat{U})$
- $H(U) = \log \binom{M}{K_a}$
- From previous: $H(U|\hat{U}) \le \epsilon K_a \log \frac{eM}{\epsilon K_a} + K_a h(\epsilon)$

[3] A very similar problem was considered in: Reeves-Gastpar, "Approximate Sparsity Pattern Recovery: Information-Theoretic Lower Bounds," 2013

Converse: proof IV

Problem (Strange rate-distortion problem[3])

Find $R(\epsilon) \triangleq \min I(U;\hat{U})$ *where* $U \sim \mathrm{Uniform}[\binom{M}{K_a}]$ *and*

$$\mathbb{E}[d(U,\hat{U})] \le 2K_a\epsilon$$

- $I(U;\hat{U}) = H(U) - H(U|\hat{U})$
- $H(U) = \log \binom{M}{K_a}$
- From previous: $H(U|\hat{U}) \le \epsilon K_a \log \frac{eM}{\epsilon K_a} + K_a h(\epsilon)$
- $R(\epsilon) \ge \log \binom{M}{K_a} - K_a[\log \frac{eM}{\epsilon K_a} + h(\epsilon)]$

[3] A very similar problem was considered in: Reeves-Gastpar, "Approximate Sparsity Pattern Recovery: Information-Theoretic Lower Bounds," 2013

Converse: proof IV

Problem (Strange rate-distortion problem[3])

Find $R(\epsilon) \triangleq \min I(U;\hat{U})$ *where* $U \sim \mathrm{Uniform}[\binom{M}{K_a}]$ *and*

$$\mathbb{E}[d(U,\hat{U})] \le 2K_a\epsilon$$

- $I(U;\hat{U}) = H(U) - H(U|\hat{U})$
- $H(U) = \log\binom{M}{K_a}$
- From previous: $H(U|\hat{U}) \le \epsilon K_a \log\frac{eM}{\epsilon K_a} + K_a h(\epsilon)$
- $R(\epsilon) \ge \log\binom{M}{K_a} - K_a[\log\frac{eM}{\epsilon K_a} + h(\epsilon)]$
- This completes the proof: Every UMAC code must satisfy

$$\frac{n}{2}\log(1+K_aP) \ge \log\binom{M}{K_a} - K_a\left(\epsilon\log\frac{Me}{\epsilon K_a} + h(\epsilon)\right)$$

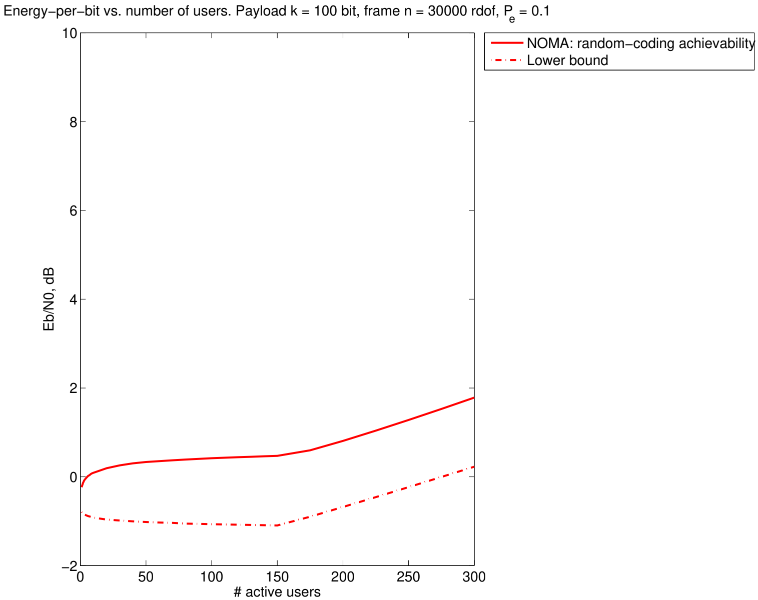
[3] A very similar problem was considered in: Reeves-Gastpar, "Approximate Sparsity Pattern Recovery: Information-Theoretic Lower Bounds," 2013

- Let us evaluate numerically these bounds.
- Need to fix parameters (n, k, K_a).

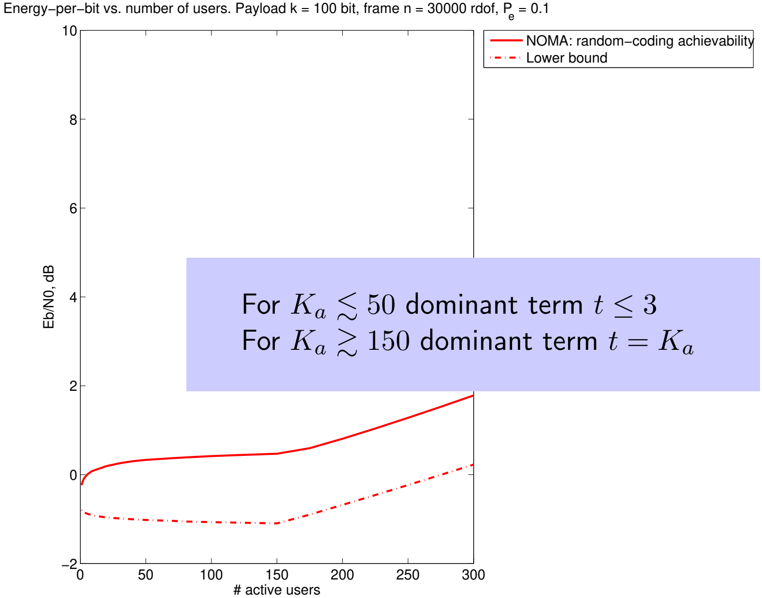
- Let us evaluate numerically these bounds.
- Need to fix parameters (n, k, K_a).
- Consider a typical LoRa network:
  - payloads $k \approx 100$ bit
  - a message in SF11 occupies $\approx k\frac{2^{11}}{11}$ complex d.o.f. $\Rightarrow n \approx 30000$.

- Let us evaluate numerically these bounds.
- Need to fix parameters (n, k, K_a).
- Consider a typical LoRa network:
  - payloads $k \approx 100$ bit
  - a message in SF11 occupies $\approx k\frac{2^{11}}{11}$ complex d.o.f. $\Rightarrow n \approx 30000$.
- Our choices from now on:
  - Frame length $n = 30000$ (real d.o.f.)
  - User payload: $k = 100$ bits
  - Active users: $K_a = 1 \ldots 300$ (variable)
  - Target error PUPE = 0.1 or 0.001
  - Goal: Find minimal $\frac{E_b}{N_0} \triangleq \frac{nP}{2k}$.

IT bounds evaluation: PUPE=0.1



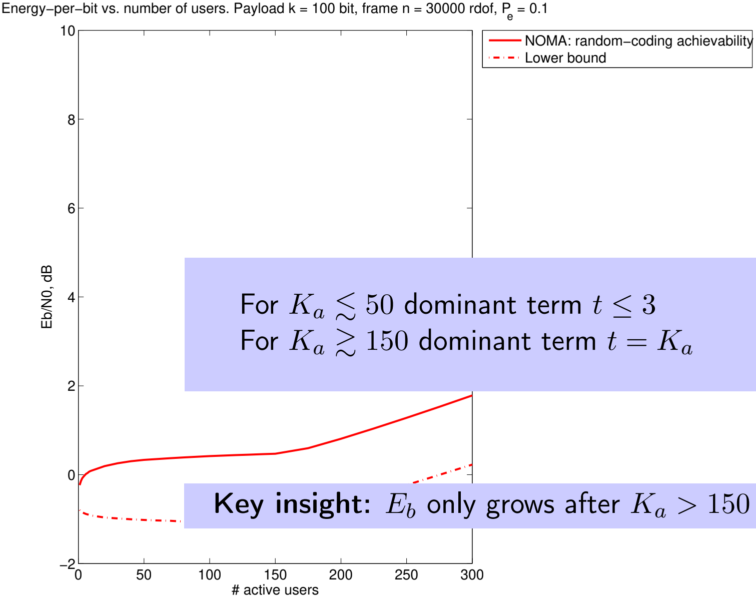
IT bounds evaluation: PUPE=0.1



For $K_a \lesssim 50$ dominant term $t \leq 3$

For $K_a \gtrsim 150$ dominant term $t = K_a$

IT bounds evaluation: PUPE=0.1



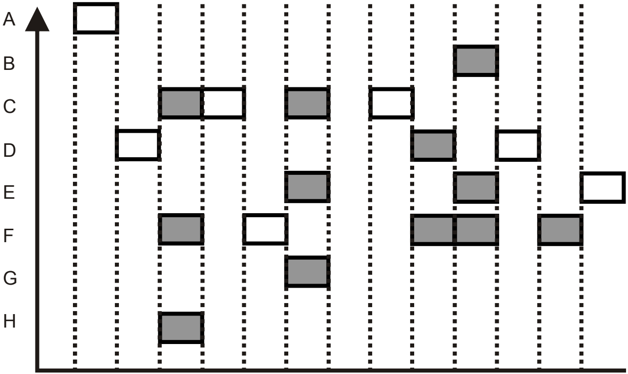
For $K_a \lesssim 50$ dominant term $t \leq 3$
For $K_a \gtrsim 150$ dominant term $t = K_a$

Key insight: E_b only grows after $K_a > 150$ (!!)

- **Surprise:** The energy-per-bit stays almost constant
  ... as if only 1 user were sending!
- But this is for an "optimal" system (random-coding).

- **Surprise:** The energy-per-bit stays almost constant
  ... as if only 1 user were sending!
- But this is for an "optimal" system (random-coding).
- What about performance of practically employed schemes?
- We will consider two:
  - ALOHA
  - Treat-interference-as-Noise (TIN)

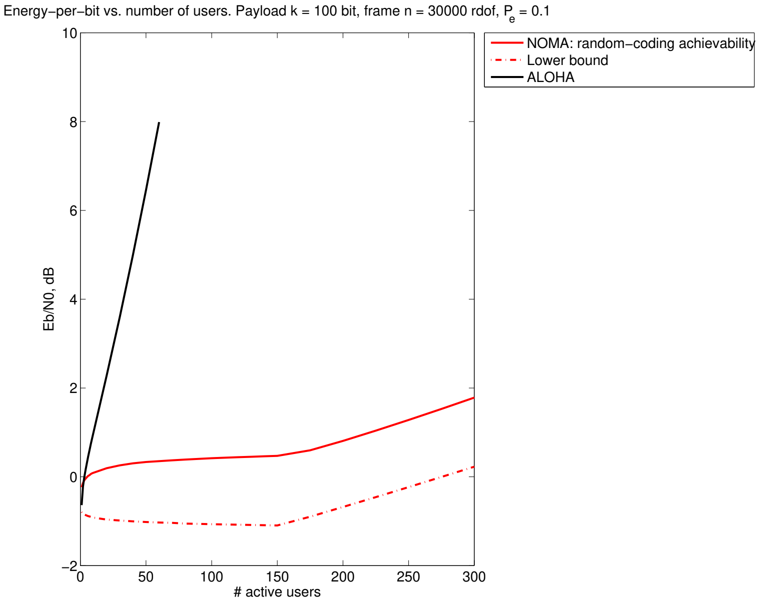
Mother of all random-access: ALOHA



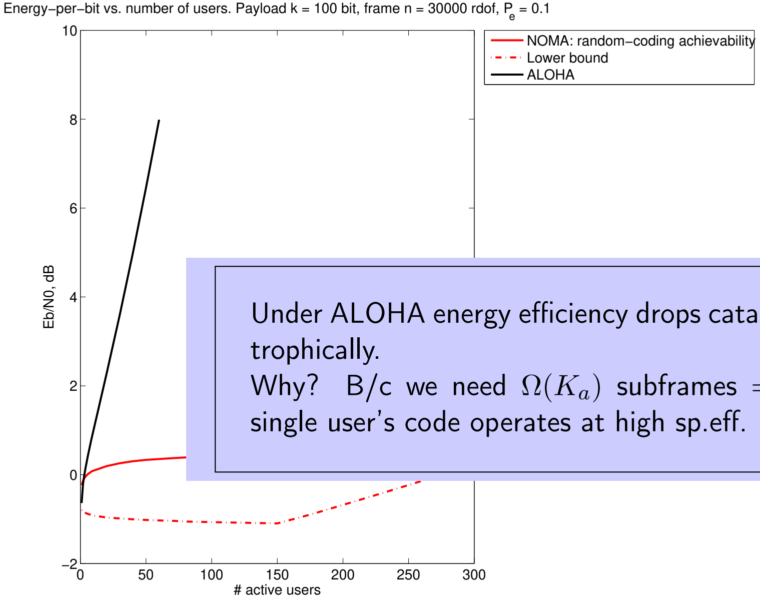
Slotted ALOHA protocol (shaded slots indicate collision)

- Each user places his n_1-codeword into one of L subframes.
- If two users select same subframe: both are lost.

Fundamental limits vs. ALOHA



Fundamental limits vs. ALOHA



Under ALOHA energy efficiency drops catastrophically.
Why? B/c we need $\Omega(K_a)$ subframes $\Rightarrow$ single user's code operates at high sp.eff.

Treat interference as noise (TIN)

Theorem (DT-TIN bound)

There exists $\mathcal{C} \subset B(0, \sqrt{nP})$ *of size* M *such that*

$$\text{PUPE} \leq \mathbb{E}\left[e^{-|i(X_1;Y)-\log M|^+}\right] + \mathbb{P}[\chi^2(n) > n\frac{P}{P'}]$$

where $Y = \sum_{i=1}^{K_a} X_i + Z$, $X_i \sim \mathcal{N}(0, P'I_n)^{\otimes n}$ *and* $Z \sim \mathcal{N}(0, I_n)$ *and* $i(x;y) = nC_{TIN}(P') + \frac{\log e}{2}\left[\frac{\|y\|^2}{1+K_aP'} - \frac{\|y-x\|^2}{1+(K_a-1)P'}\right]$.

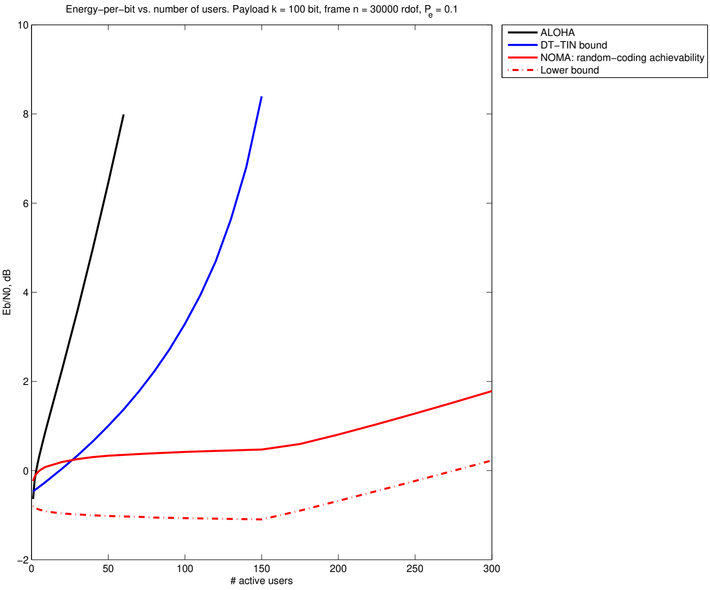
Remarks:

- Decoder outputs K_a closest codewords: PUPE $\leq \mathbb{P}[X_1 \notin \{\text{top-}K_a \text{ closest c/w to } Y\}]$
- Achieves about $\log M \approx nC_{TIN}(P) - \sqrt{nV_{TIN}(P)}Q^{-1}(\epsilon)$

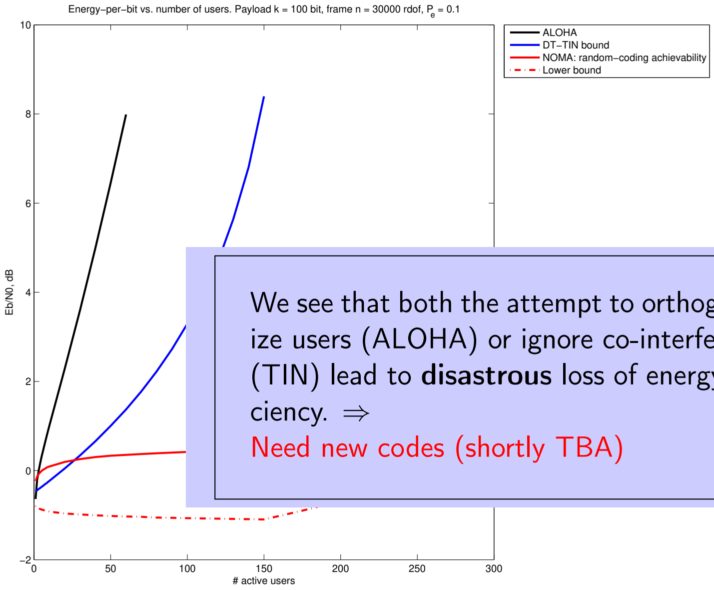
$$C_{TIN}(P) = \frac{1}{2}\log\left(1 + \frac{P}{1+(K_a-1)P}\right), \quad V_{TIN}(P) = \frac{P\log^2 e}{1+K_aP}.$$

- Spectral efficiency as $K_a \to \infty$ is bounded by $\frac{\log_2 e}{2} \approx 0.72$ bit.

Treat interference as noise (TIN): evaluation



Treat interference as noise (TIN): evaluation



We see that both the attempt to orthogonalize users (ALOHA) or ignore co-interference (TIN) lead to **disastrous** loss of energy efficiency. $\Rightarrow$

Need new codes (shortly TBA)

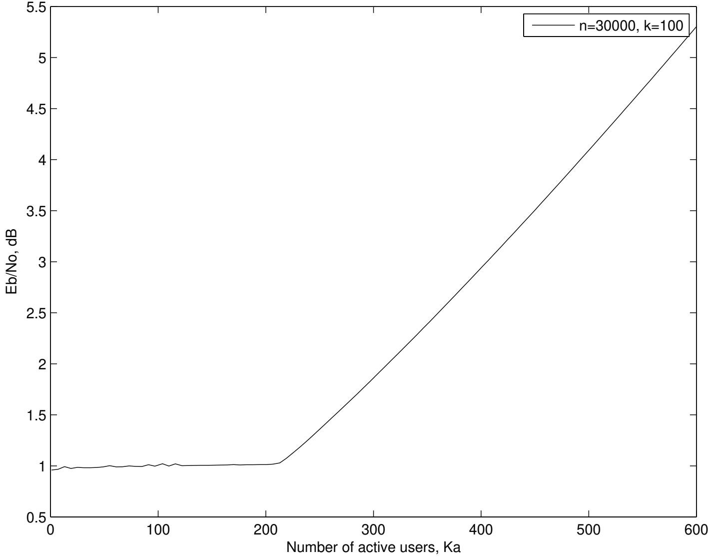
Effect of n and k

- Good engineer: *How do these curves change with blocklength* n *and payload size* k*?*
- Good info-theorist: *Can we formulate an asymptotic question* $n \to \infty$*?*

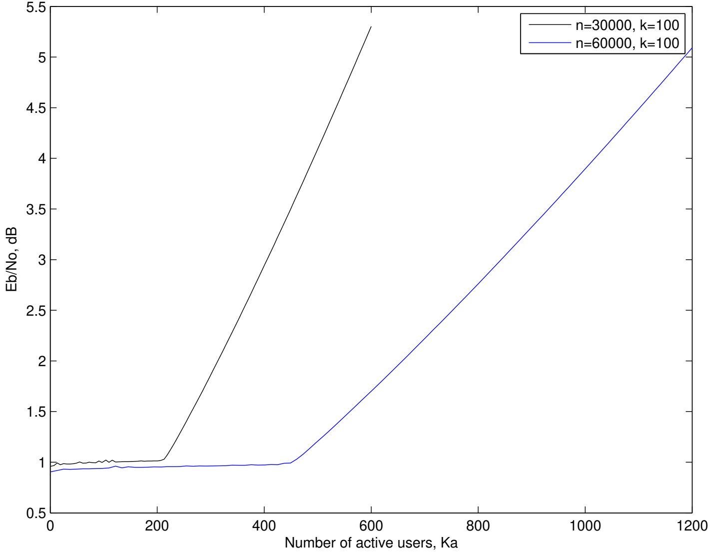
Effect of n and k

- Good engineer: *How do these curves change with blocklength n and payload size k?*
- Good info-theorist: *Can we formulate an asymptotic question $n \to \infty$?*
- Let us evaluate the bounds for various n...

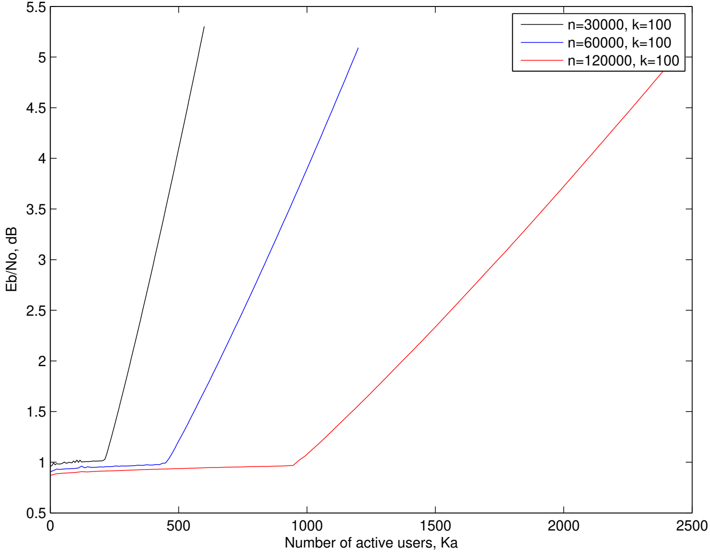
Comparing random coding bound at different n



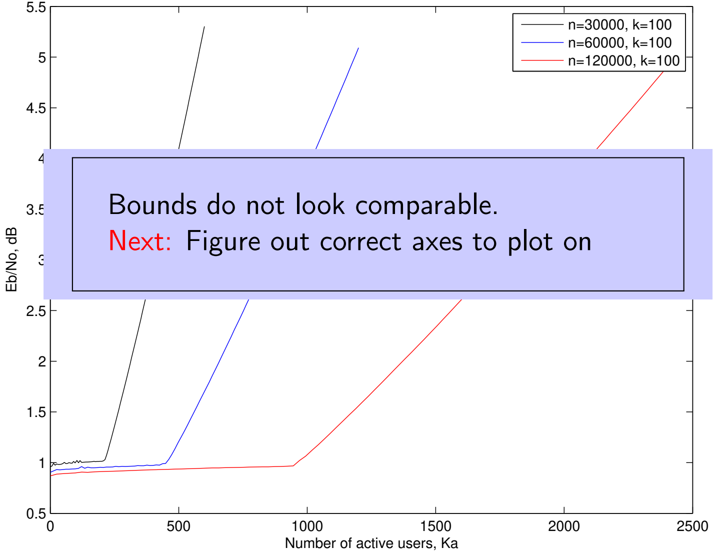
Comparing random coding bound at different n



Comparing random coding bound at different n



Comparing random coding bound at different n



Bounds do not look comparable.

Next: Figure out correct axes to plot on

Effective $\frac{E_b}{N_0}$ and spectral efficiency

- So far we used axes:

$$K_a \text{ vs } \frac{E_b}{N_0} \triangleq \frac{nP}{2k}\,.$$

- **Problem:** As n changes there is no "convergence to the limit".

Effective $\frac{E_b}{N_0}$ and spectral efficiency

- So far we used axes:

$$K_a \text{ vs } \frac{E_b}{N_0} \triangleq \frac{nP}{2k}\,.$$

- **Problem:** As n changes there is no "convergence to the limit".
- The issue is that we have not defined our quantities correctly. It turns out the key quantities in this problem are:
  - Effective number of total bits $k_{tot} = \log_2 \binom{M}{K_a} \approx K_a \log_2 \frac{eM}{K_a}$
    (total information transmitted)

Effective $\frac{E_b}{N_0}$ and spectral efficiency

- So far we used axes:

$$K_a \text{ vs } \frac{E_b}{N_0} \triangleq \frac{nP}{2k}\,.$$

- **Problem:** As n changes there is no "convergence to the limit".
- The issue is that we have not defined our quantities correctly. It turns out the key quantities in this problem are:
    - Effective number of total bits $k_{tot} = \log_2 \binom{M}{K_a} \approx K_a \log_2 \frac{eM}{K_a}$ (total information transmitted)
    - $\Rightarrow$ effective spectral efficiency and energy-per-bit are

$$\rho \triangleq \frac{k_{tot}}{n}, \quad \left(\frac{E_b}{N_0}\right)_{eff} = \frac{nPK_a}{2k_{tot}}$$

Effective $\frac{E_b}{N_0}$ and spectral efficiency

- So far we used axes:

$$K_a \text{ vs } \frac{E_b}{N_0} \triangleq \frac{nP}{2k}\,.$$

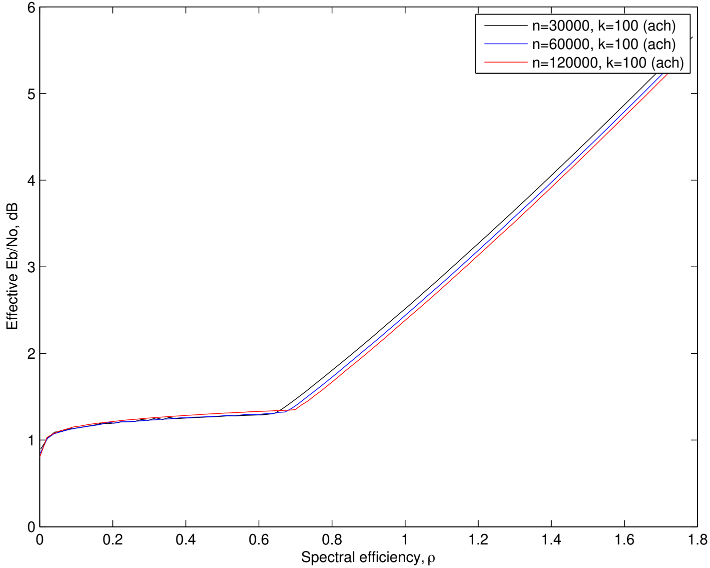
- **Problem:** As n changes there is no "convergence to the limit".
- The issue is that we have not defined our quantities correctly. It turns out the key quantities in this problem are:
  - Effective number of total bits $k_{tot} = \log_2 \binom{M}{K_a} \approx K_a \log_2 \frac{eM}{K_a}$ (total information transmitted)
  - $\Rightarrow$ effective spectral efficiency and energy-per-bit are

$$\rho \triangleq \frac{k_{tot}}{n}, \quad \left(\frac{E_b}{N_0}\right)_{eff} = \frac{nPK_a}{2k_{tot}}$$

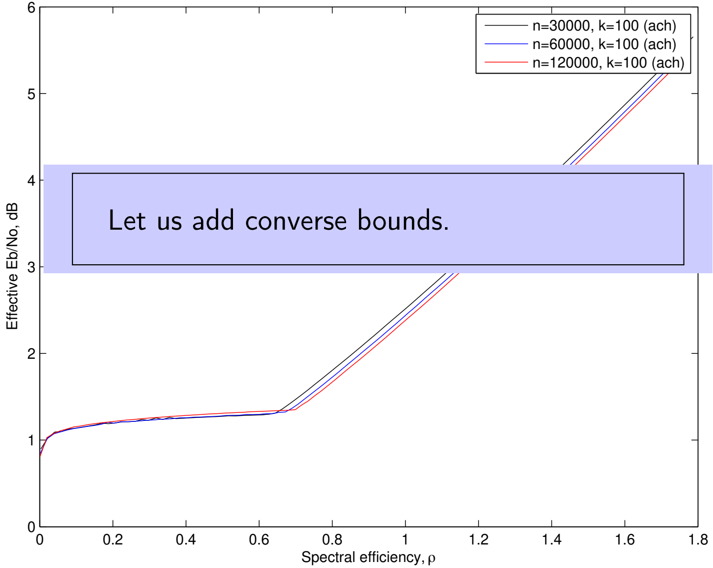
- Let us try replotting in these new axes:

$$\rho \text{ vs } \left(\frac{E_b}{N_0}\right)_{eff}$$

Effective E_b/N_0 vs spectral efficiency: different n $(k = 100)$

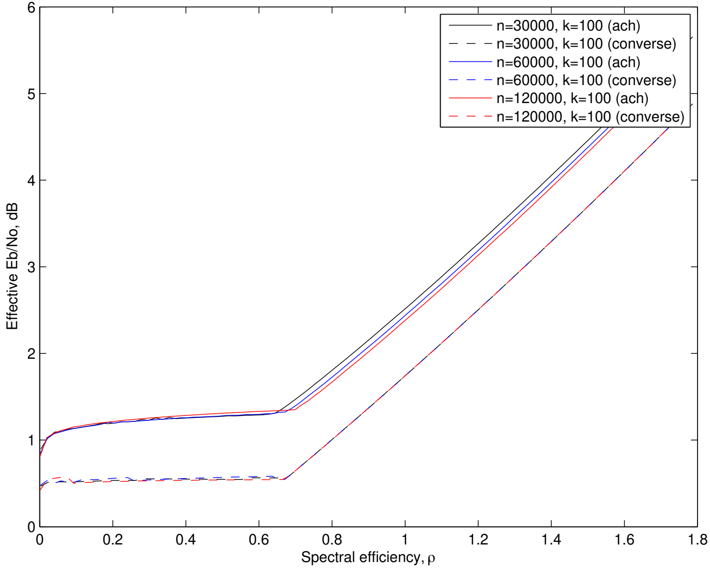


Effective E_b/N_0 vs spectral efficiency: different n $(k = 100)$

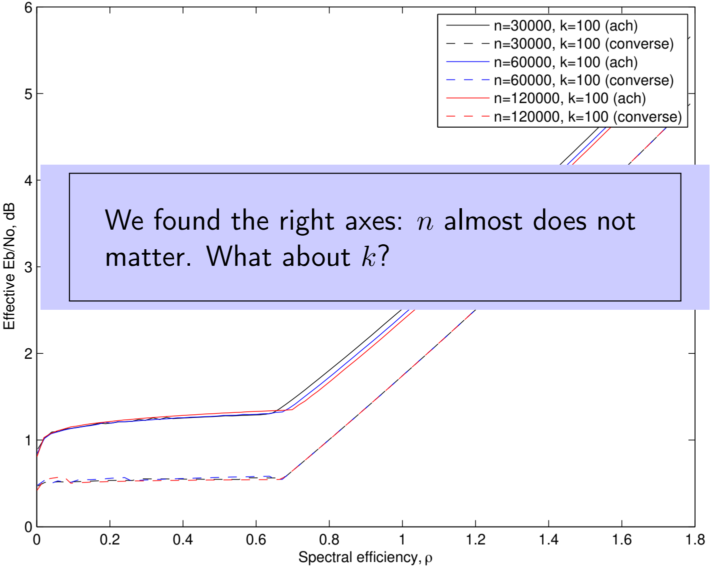


Let us add converse bounds.

Effective E_b/N_0 vs spectral efficiency: different n $(k = 100)$

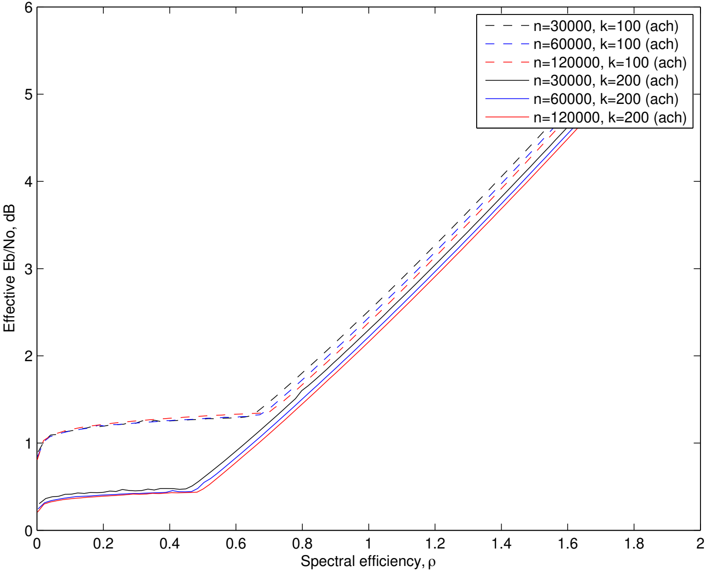


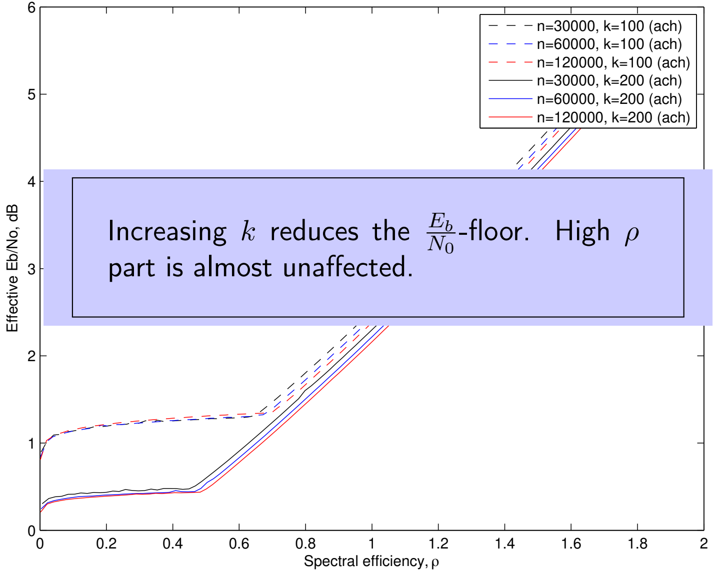
Effective E_b/N_0 vs spectral efficiency: different n ($k = 100$)



We found the right axes: n almost does not matter. What about k?

Effective E_b/N_0 vs spectral efficiency: different k





Increasing k reduces the $\frac{E_b}{N_0}$-floor. High ρ part is almost unaffected.

Asymptotics and open problems

- As good info-theorists we should be excited: curves seem to converge to some limit as $n \to \infty$.

Asymptotics and open problems

- As good info-theorists we should be excited: curves seem to converge to some limit as $n \to \infty$.
- To identify this limit, let us notice that our problem is in fact equivalent to support recovery in compressed sensing.

Same-codebook codes = compressed sensing

- UMAC = all users share same codebook
- UMAC = decoder only reconstructs list of messages (i.e. vector $\{0,1\}^M$ of weight K_a)
- Equivalent to compressed-sensing (CS) [Jin-Kim-Rao'11]

Same-codebook codes = compressed sensing

- UMAC = all users share same codebook
- UMAC = decoder only reconstructs list of messages (i.e. vector $\{0,1\}^M$ of weight K_a)
- Equivalent to compressed-sensing (CS) [Jin-Kim-Rao'11]
- Let same-codebook (column) vectors be $c_1, \ldots c_j$.

$$X = \begin{pmatrix} c_1 & | & \cdots & | & c_M \end{pmatrix}$$

- Let $\beta \in \{0,1\}^M$ with $\beta_j = 1$ if codeword j was transmitted
- Then the problem is:

$$Y = X\beta + Z, \qquad \text{Goal: } \mathbb{E}[\|\beta - \hat{\beta}(Y)\|] \to \min$$

(linear regression with sparsity $\|\beta\|_0 = K_a$ aka CS).

Same-codebook codes = compressed sensing

- UMAC = all users share same codebook
- UMAC = decoder only reconstructs list of messages (i.e. vector $\{0,1\}^M$ of weight K_a)
- Equivalent to compressed-sensing (CS) [Jin-Kim-Rao'11]
- Let same-codebook (column) vectors be $c_1, \dots c_j$.

$$X = \begin{pmatrix} c_1 & | & \cdots & | & c_M \end{pmatrix}$$

- Let $\beta \in \{0,1\}^M$ with $\beta_j = 1$ if codeword j was transmitted
- Then the problem is:

$$Y = X\beta + Z, \qquad \text{Goal: } \mathbb{E}[\|\beta - \hat{\beta}(Y)\|] \to \min$$

(linear regression with sparsity $\|\beta\|_0 = K_a$ aka CS).

- Suppose the entries of X are iid $\mathcal{N}(0, P)$. Then we get Gaussian random design CS (GCS).

Same-codebook codes = compressed sensing

- UMAC = all users share same codebook
- UMAC = decoder only reconstructs list of messages (i.e. vector $\{0,1\}^M$ of weight K_a)
- Equivalent to compressed-sensing (CS) [Jin-Kim-Rao'11]
- Let same-codebook (column) vectors be $c_1, \dots c_j$.

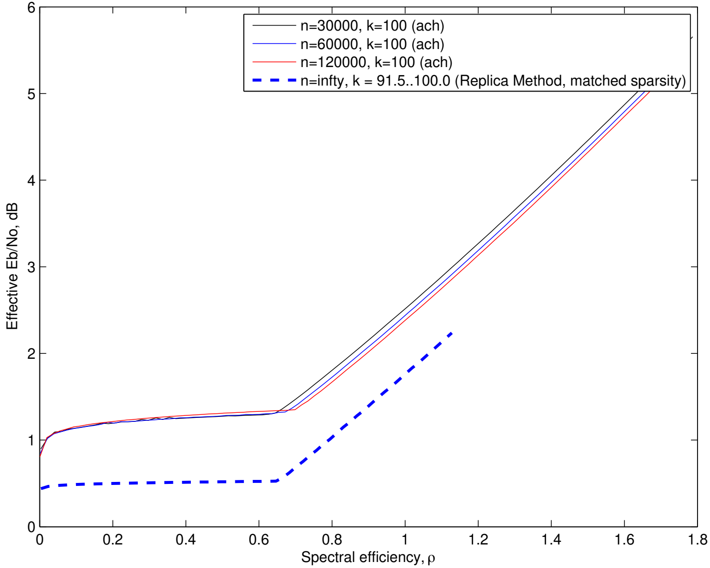
$$X = \begin{pmatrix} c_1 & | & \cdots & | & c_M \end{pmatrix}$$

- Let $\beta \in \{0,1\}^M$ with $\beta_j = 1$ if codeword j was transmitted
- Then the problem is:

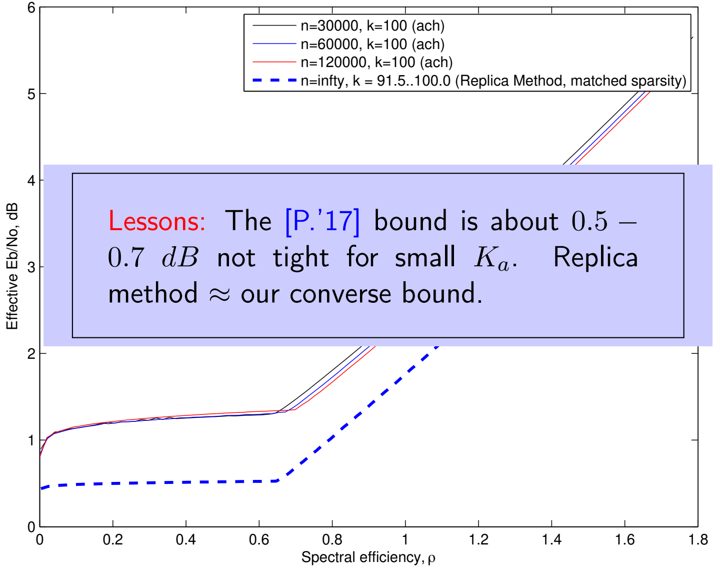
$$Y = X\beta + Z, \qquad \text{Goal: } \mathbb{E}[\|\beta - \hat{\beta}(Y)\|] \to \min$$

(linear regression with sparsity $\|\beta\|_0 = K_a$ aka CS).
- Suppose the entries of X are iid $\mathcal{N}(0, P)$. Then we get Gaussian random design CS (GCS).
- Fundamental limits of GCS were studied in the limit of $n \to \infty$ at a fixed aspect ratio $\delta = \frac{n}{M}$ and sparsity $\pi = \frac{K_a}{M}$. The minimal PUPE in this limit is given by replica method.

Finite blocklength bound vs. $n = \infty$ asymptotics



Finite blocklength bound vs. $n = \infty$ asymptotics



Lessons: The [P.'17] bound is about $0.5 - 0.7\ dB$ not tight for small K_a. Replica method $\approx$ our converse bound.

Extra: replica method[4]

[4] More details in Section V.A of Kowshik-Polyanskiy, "Fundamental limits of many-user MAC with finite payloads and fading", 2021

Asymptotics of random-access

- We say that $\mathcal{E}$ is asymptotically achievable effective E_b/N_0 at (M_{eff}, μ, ϵ) if $\exists(n, M, K_a, \epsilon)$ RA-code with $M = M_{eff} K_a$, $K_a = \mu n$ and codewords of energy

$$\|c\|_2^2 \leq 2\mathcal{E} \log_2 M_{eff}$$

  for all $n \to \infty$.

- Asymptotic fundamental limit: minimal achievable $\mathcal{E}$, i.e.

$$E^*_\infty(M_{eff}, \mu, \epsilon) = \limsup_{n\to\infty} \frac{\log_2 M}{\log M_{eff}} E^*_b(n, M, K_a, \epsilon)$$

Asymptotics of RA and CS

- Recall connection to the compressed sensing.
- Call $E > 0$ feasible at a given ratio p/n and sparsity π if:

$$Y = \sqrt{E} X\beta + Z, \qquad Z \sim \mathcal{N}(0, I_n), \beta \in R^p$$

  - Columns of X are of unit energy
  - $\beta \in \{0,1\}^p$ and $\|\beta\|_0 = \pi p$,
  - $\exists \hat{\beta}(Y, X)$ such that

$$\begin{aligned} \|\hat{\beta}\|_0 &\leq \mu n \qquad \text{(FDR)} \\ \|\hat{\beta} - \beta\|_0 &\leq 2\epsilon \|\beta\|_0 \end{aligned}$$

- Then we have $E^*_\infty = \min \frac{E}{2\log_2 M_{eff}}$

Asymptotics of RA and CS

- Recall connection to the compressed sensing.
- Call $E > 0$ feasible at a given ratio p/n and sparsity π if:

$$Y = \sqrt{E} X\beta + Z, \qquad Z \sim \mathcal{N}(0, I_n), \beta \in R^p$$

  - Columns of X are of unit energy
  - $\beta \in \{0,1\}^p$ and $\|\beta\|_0 = \pi p$,
  - $\exists \hat{\beta}(Y, X)$ such that

$$\begin{aligned} \|\hat{\beta}\|_0 &\leq \mu n \qquad (\text{FDR}) \\ \|\hat{\beta} - \beta\|_0 &\leq 2\epsilon \|\beta\|_0 \end{aligned}$$

- Then we have $E^*_\infty = \min \frac{E}{2 \log_2 M_{eff}}$
- When $X \overset{iid}{\sim} \mathcal{N}(0, 1/n)$ this is well studied in stat. physics.

Replica method prediction

- Consider a scalar problem:

$$B = \sqrt{E_1} A + N\,, \qquad A \sim \mathrm{Ber}(\pi) \perp\!\!\!\perp N \sim \mathcal{N}(0,1)$$

- Define $I_1(E_1) = I(A; B)$ and

$$p^*(E_1, \pi) = \min_{\hat{A}} \left\{ \mathbb{P}[A = 0 | \hat{A} = 1] : \mathbb{P}[\hat{A} = 1] = \pi \right\}$$

- It can be seen that p^* is a solution of

$$\sqrt{E_1} = Q^{-1}(p^*) + Q^{-1}\left(\frac{\pi p^*}{1-\pi}\right)\,.$$

Replica method prediction

- Consider a scalar problem:

$$B = \sqrt{E_1} A + N\,, \qquad A \sim \mathrm{Ber}(\pi) \perp\!\!\!\perp N \sim \mathcal{N}(0,1)$$

- Define $I_1(E_1) = I(A;B)$ and

$$p^*(E_1,\pi) = \min_{\hat{A}} \left\{ \mathbb{P}[A=0|\hat{A}=1] : \mathbb{P}[\hat{A}=1] = \pi \right\}$$

- It can be seen that p^* is a solution of

$$\sqrt{E_1} = Q^{-1}(p^*) + Q^{-1}\left(\frac{\pi p^*}{1-\pi}\right) .$$

- Stat. physics predicts that inference in

$$Y = \sqrt{E} X\beta + Z, \qquad X \stackrel{iid}{\sim} \mathcal{N}(0,1/n), \beta \sim \mathrm{Ber}^{\otimes p}(\pi)$$

  is asymptotically equivalent to a scalar problem with $E_1 = E\eta$
- $\eta \in [0,1]$ (the multi-user efficiency) is given as a solution of

$$\eta = \operatorname*{argmin}_x \left[\frac{p}{n} I_1(xE) + \frac{1}{2}(x - 1 - \ln x) \right]$$

$$B = \sqrt{\eta E} A + N, \qquad A \sim \mathrm{Ber}(\pi) \perp\!\!\!\perp N \sim \mathcal{N}(0,1)$$

$$Y = \sqrt{E} X\beta + Z, \qquad X \overset{iid}{\sim} \mathcal{N}(0, 1/n), \beta \sim \mathrm{Ber}^{\otimes p}(\pi)$$

Theorem (Replica formula exact for binary β)

Consider a sequence of random variables

$$V_n = \mathbb{P}[\beta_1 = 1 | Y, X] \in [0,1]$$

as $p, n \to \infty$ *with* $p/n = \mathrm{const}$. *Then*

$$V_n \overset{(d)}{\to} \mathbb{P}[A = 1 | B] \,.$$

$$B = \sqrt{\eta E} A + N, \qquad A \sim \mathrm{Ber}(\pi) \perp\!\!\!\perp N \sim \mathcal{N}(0,1)$$

$$Y = \sqrt{E} X \beta + Z, \qquad X \overset{iid}{\sim} \mathcal{N}(0, 1/n), \beta \sim \mathrm{Ber}^{\otimes p}(\pi)$$

Theorem (Replica formula exact for binary β)

Consider a sequence of random variables

$$V_n = \mathbb{P}[\beta_1 = 1 | Y, X] \in [0,1]$$

as $p, n \to \infty$ *with* $p/n = \mathrm{const}$. *Then*

$$V_n \overset{(d)}{\to} \mathbb{P}[A = 1 | B] \,.$$

- Pfister-Reeves and Barbier-Macris have shown that

$$\mathrm{Var}[\beta_1 | Y, X] \to \mathrm{Var}[A | B]$$

- This is not enough to conclude the proof.

$$B = \sqrt{\eta E} A + N, \qquad A \sim \mathrm{Ber}(\pi) \perp\!\!\!\perp N \sim \mathcal{N}(0,1)$$
$$Y = \sqrt{E} X\beta + Z, \qquad X \overset{iid}{\sim} \mathcal{N}(0, 1/n), \beta \sim \mathrm{Ber}^{\otimes p}(\pi)$$

Theorem (Replica formula exact for binary β)

Consider a sequence of random variables

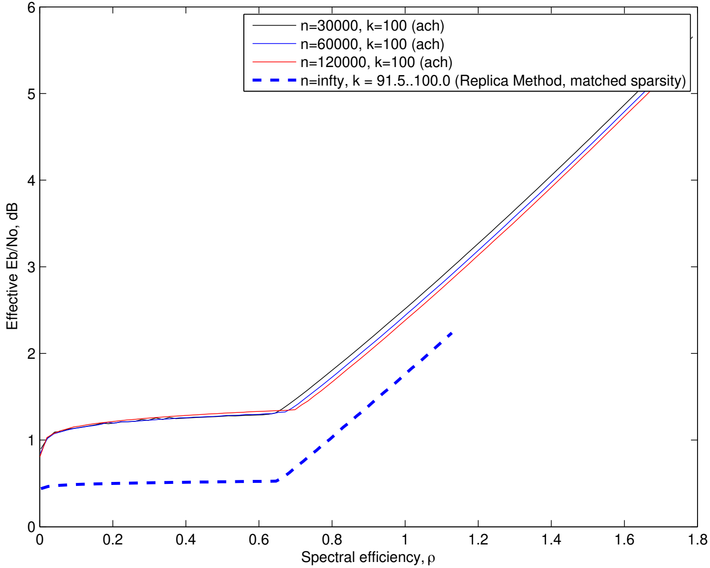
$$V_n = \mathbb{P}[\beta_1 = 1 | Y, X] \in [0,1]$$

as $p, n \to \infty$ with $p/n = \mathrm{const}$. Then

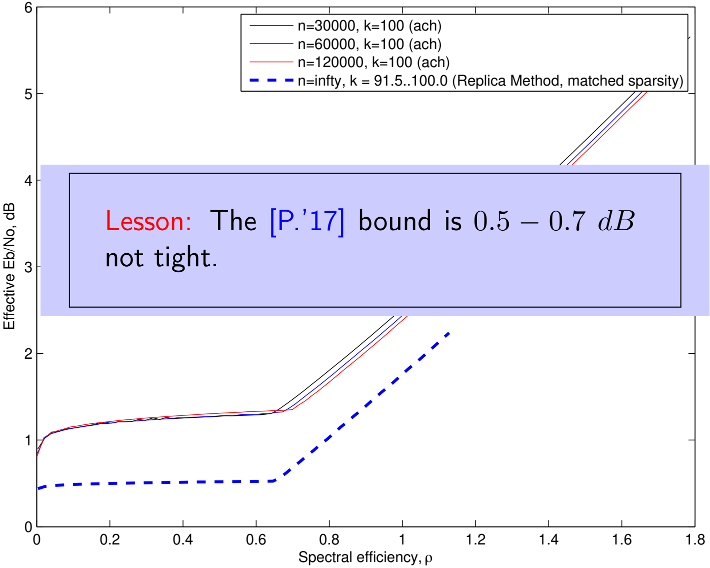
$$V_n \overset{(d)}{\to} \mathbb{P}[A = 1 | B] \,.$$

- Possible to argue indirectly for binary β **only**.
- If we have some sequence $G_n = G_n(Y, X) \in [0,1]$ s.t. $\mathbb{E}[(G_n - \beta_1)^2] \to \mathrm{Var}[\beta_1 | Y, X]$ then $G_n \overset{(d)}{\to} \mathbb{E}[\beta_1 | Y, X]$.
  For binary, this is $= \mathbb{P}[\beta_1 = 1 | X, Y]$.
- AMP started at true β yields such a G_n. The law of G_n is known to converge to $\mathbb{P}[A = 1 | B]$.

Finite blocklength bound vs. $n = \infty$ asymptotics

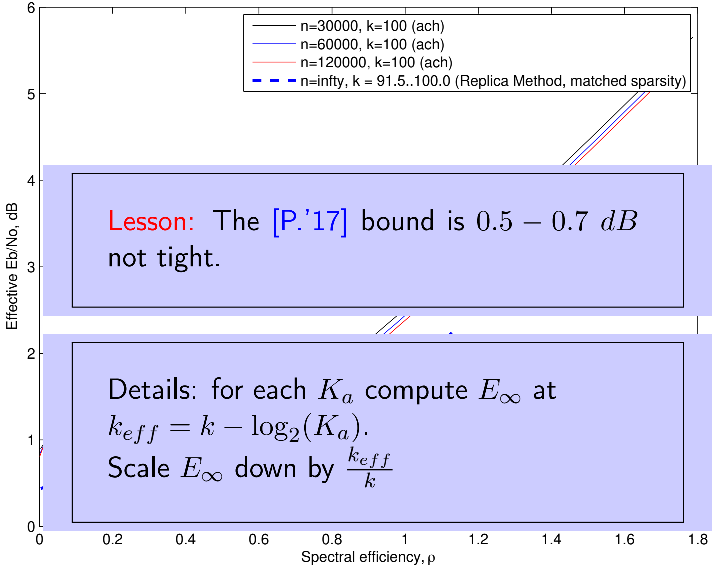


Finite blocklength bound vs. $n = \infty$ asymptotics



Lesson: The [P.'17] bound is $0.5 - 0.7\ dB$ not tight.

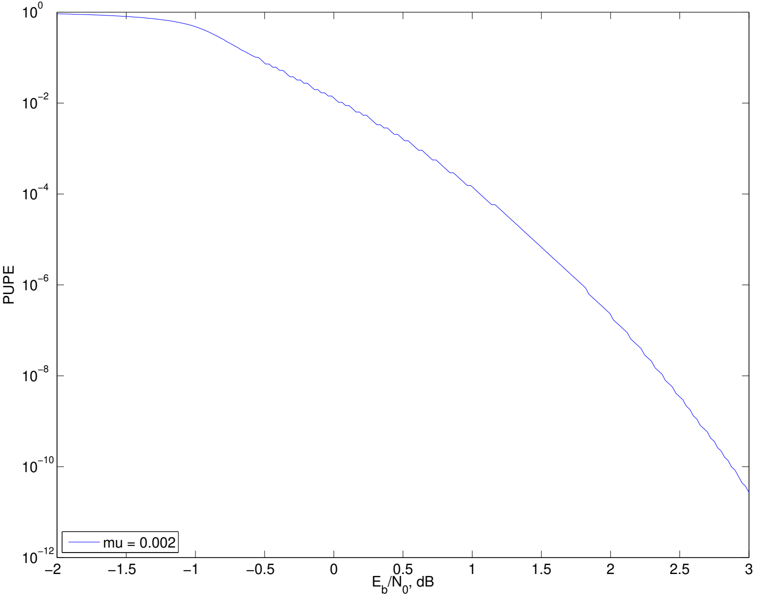
Finite blocklength bound vs. $n = \infty$ asymptotics



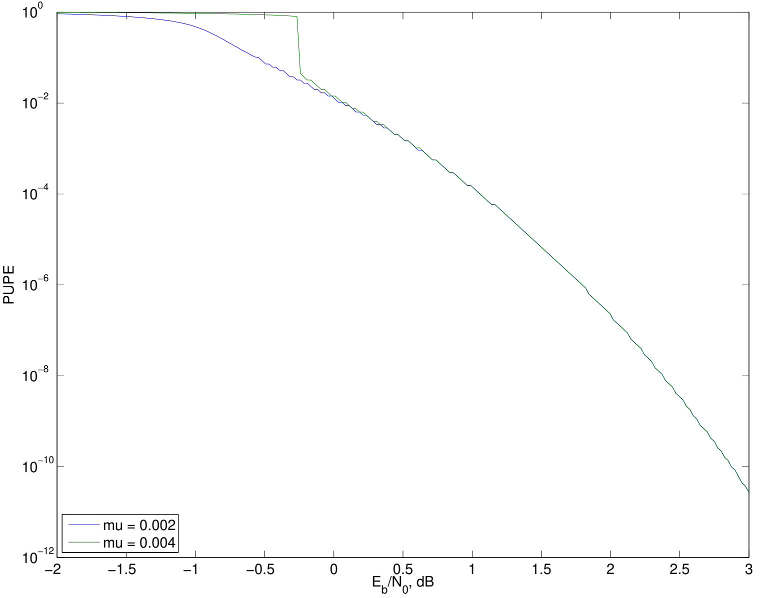
Lesson: The [P.'17] bound is $0.5 - 0.7\ dB$ not tight.

Details: for each K_a compute E_∞ at $k_{eff} = k - \log_2(K_a)$.
Scale E_∞ down by $\frac{k_{eff}}{k}$

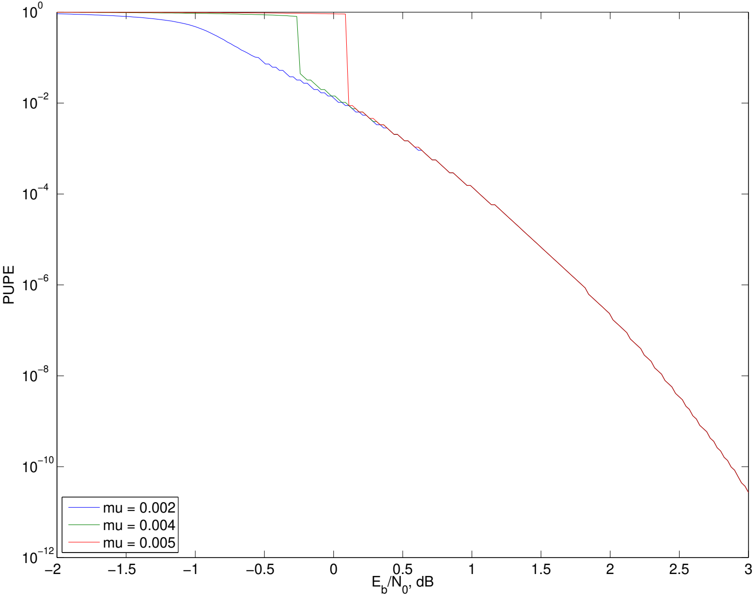
Perfect multi-user interference elimination



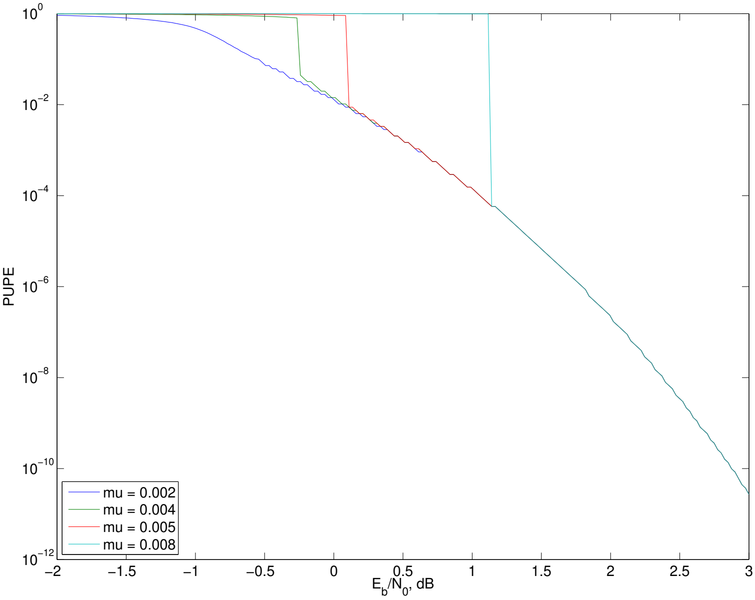
Perfect multi-user interference elimination



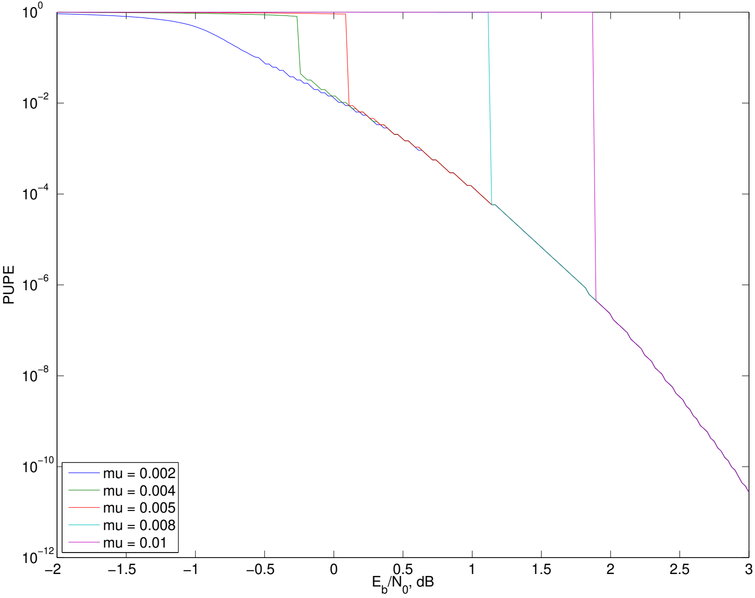
Perfect multi-user interference elimination



Perfect multi-user interference elimination



Perfect multi-user interference elimination



Summary

UMAC framework:

- To save battery: sensors sleep all the time, except transmissions.
- ... uncoordinated transmissions.
- Single shot: devices wake up, blast the packet, go back to sleep.
- There exist low E_b/N_0 schemes with high # of users.
- ... but standard ideas (orthogonalize, TIN) lead to sharp E_b/N_0 growth as # users grows.

Summary

UMAC framework:

- To save battery: sensors sleep all the time, except transmissions.
- ... uncoordinated transmissions.
- Single shot: devices wake up, blast the packet, go back to sleep.
- There exist low E_b/N_0 schemes with high # of users.
- ... but standard ideas (orthogonalize, TIN) lead to sharp E_b/N_0 growth as # users grows.

Next steps:

1. Failure of standard coding solutions
2. Coded Compressed Sensing
3. Non-CS methods for UMAC

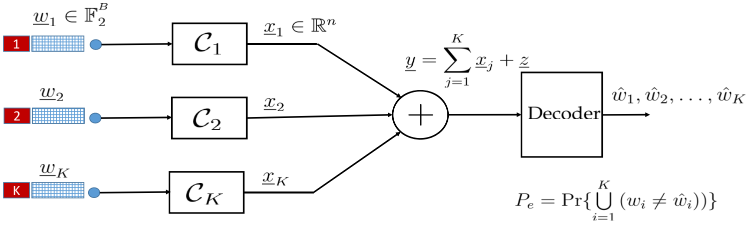
Supporting 10 users at 1Mbps is much easier than 1M users at 10bps.

Coding for Unsourced Random Access

- Brief review of coding for the Gaussian MAC (GMAC)
- Why codes for GMAC cannot be directly used for URA
- Approaches to designing codes for URA

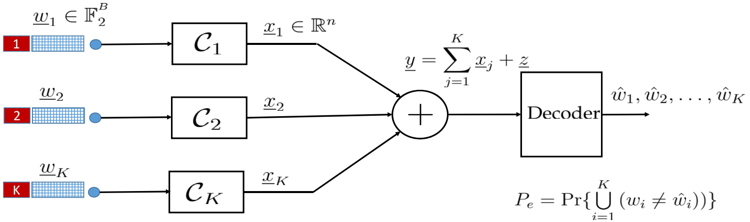
Traditional Gaussian multiple access channel (GMAC)

- K users, each user has a B-bit message
- n channel uses
- Classical information theory - fix K and let $n, B \to \infty$



Traditional Gaussian multiple access channel (GMAC)

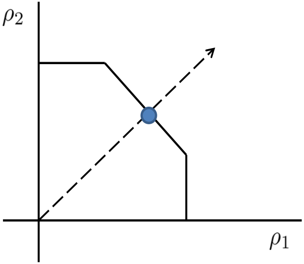
- K users, each user has a B-bit message
- n channel uses
- Classical information theory - fix K and let $n, B \to \infty$



Assumptions

- User identity is conveyed separately
- Resources are allocated based on identity
- Codebooks are different but assumed to be known at the decoder

Coding for the Traditional GMAC



Achievable sum rate

$$\sum_{i=1}^{K} \rho_i < \frac{1}{2} \log\left(1 + \frac{KP}{\sigma^2}\right)$$

Achieving points on the GMAC region

- Corner points can be achieved using successive interference cancellation
- Any point can be achieved through rate-splitting
- These require coordination among users
- Equal rate point is harder to achieve without coordination

Coding Schemes for the Equal Rate Point

- Time/Frequency/Code Division Multiple Access (T/F/CDMA)
- Ping *et al.* - Interleave division multiple access (IDMA)
- Yedla, Pfister, N. ' 11 - Spatially coupled LDPC
- Truhachev, Schlegel - Spatially coupled MA
- Sasoglu *et al.*'13 - Polar codes for MAC

All these schemes require coordination between users to pick parameters

TDMA/FDMA/CDMA

- TDMA/FDMA
  - Requires coordinated allocation of time/frequency slots
  - Without coordination, there will be collisions

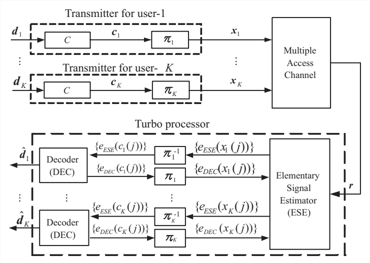
TDMA/FDMA/CDMA

- TDMA/FDMA
  - Requires coordinated allocation of time/frequency slots
  - Without coordination, there will be collisions

- Orthogonal CDMA
  - Users need to be 'assigned' spreading sequences
  - $K_{tot} \gg K$ - spreading sequence length will be too large
  - $K_{tot} \approx 10000$, $n = 30000$ and $B = 100$
  - Not enough dimensions for coding

Interleave Division Multiple Access - Ping *et al.*' 06

- Each user encodes with the same code & picks a different interleaver
- Message passing decoding and demodulation
- Close to capacity performance for small number of users



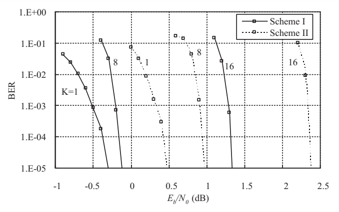
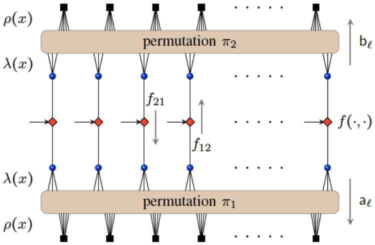


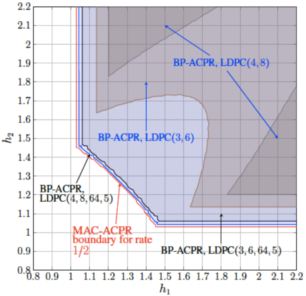
Fig. 7. Performance of IDMA systems based on the turbo-Hadamard code [31] and turbo code over AWGN channels. $N_r = 1$, $It = 30$, $N_{\text{info}} = 4095$ for Scheme I and $N_{\text{info}} = 4096$ for Scheme II.

- *The interleavers have to be different and known to the receiver*
- *Performance is not very good for large number of users*

SC-LDPC for GMAC - Yedla, Pfister, N '11

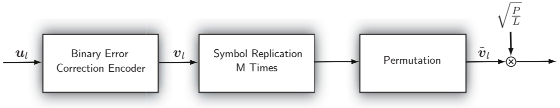
- Spatially coupled LDPC codes with different interleavers
- Empirically shown to be universal for MAC

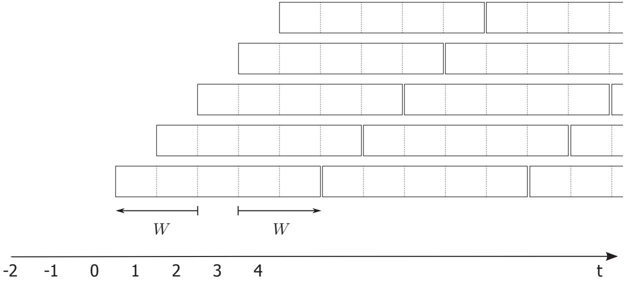




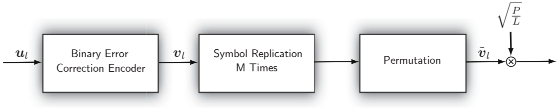
- *Interleavers need to be chosen in a coordinated manner*
- *Interleavers need to be known at the receiver*
- *Not a good solution for short block lengths*

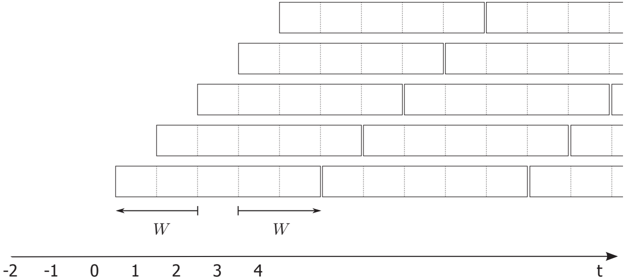
Coupling data transmission.. - Truhachev & Schlegel '12





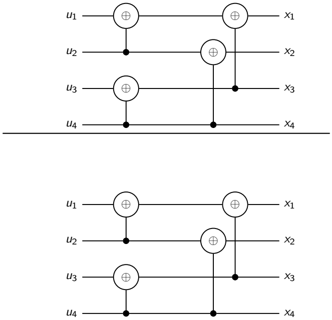
Coupling data transmission.. - Truhachev & Schlegel '12





- *Requires coordination to choose offsets*
- *Not a good solution for short block lengths*

Polar Codes for MAC - Sasoglu'13



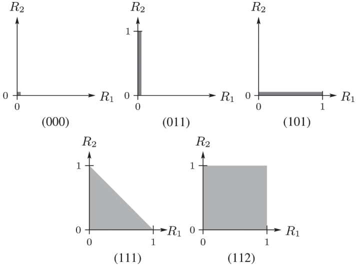


Fig. 1. Capacity regions of the five extremal MACs.

- Polar codes can be optimized for MAC
- *Frozen bits have to be chosen in a coordinated fashion*

Takeaways

Main points from this part

- Traditional GMAC channel model is not suitable for modeling IoT
- Existing coding schemes for GMAC need to be modified
- Finite Block Length achievability bounds serve as a good benchmark

Takeaways

Main points from this part

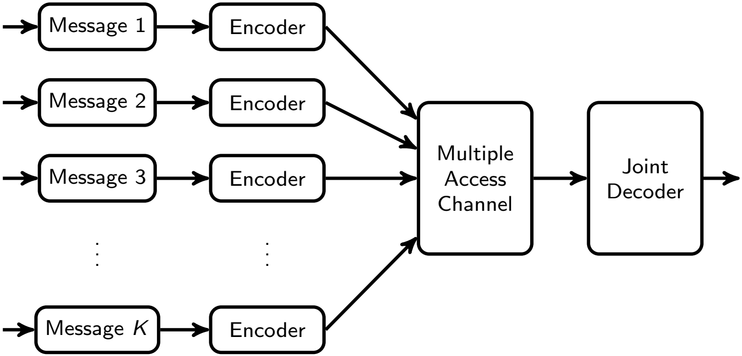
- Traditional GMAC channel model is not suitable for modeling IoT
- Existing coding schemes for GMAC need to be modified
- Finite Block Length achievability bounds serve as a good benchmark

Rest of the talk - Two main approaches to coding for URA

- *Connections between Unsourced MAC and Compressed Sensing*
- *Modifying codes for GMAC to make them work for URA*

Unsourced Random Access as Compressed Sensing

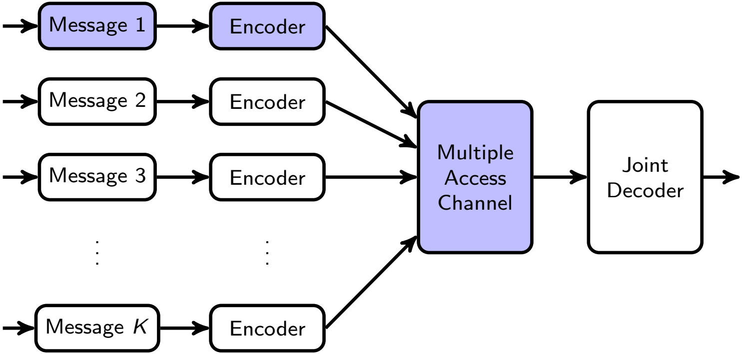
Unsourced Random Access – Encoding Function



Characteristics of URA framework

- K active devices, each with a B-bit message
- Multiple access channel

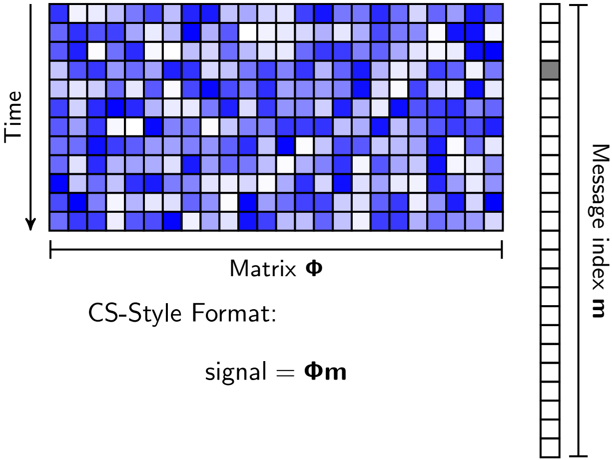
Unsourced Random Access – Encoding Function



Characteristics of URA framework

- Every device employs the same encoder $f : \{0, 1\}^B \to \mathbb{R}^n$
- Decoder must produce an unordered list of messages

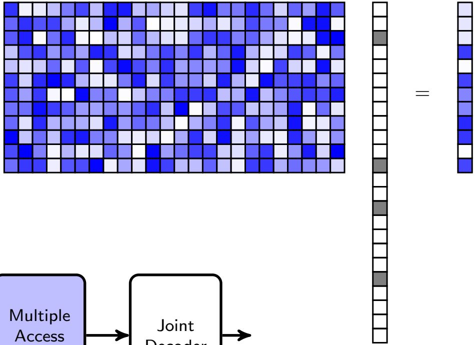
Unsourced Random Access – Index Representation

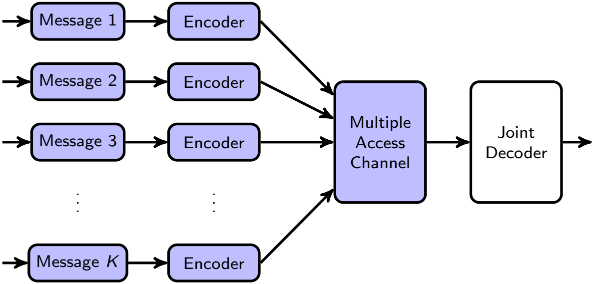


CS-Style Format:

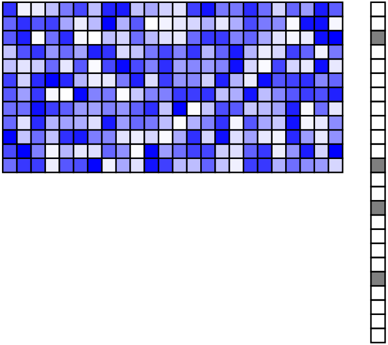
$$\text{signal} = \mathbf{\Phi}\mathbf{m}$$

Unsourced Random Access – CS Analogy





Abstract CS Challenge



Problem setting

- Noisy compressed sensing

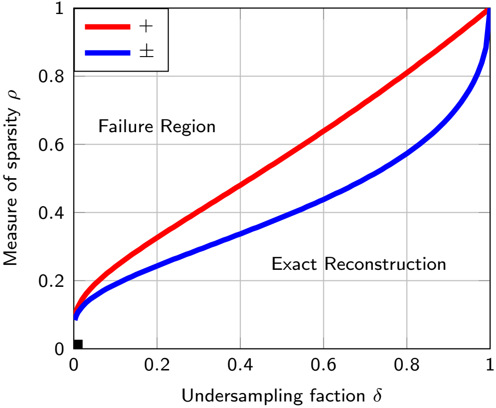
$$\mathbf{y} = \mathbf{\Phi s} + \mathbf{z}$$

  where $\mathbf{s}$ is K sparse
- $\mathbf{s}$ has non-negative integer entries
- $\mathbf{\Phi}.\mathtt{shape} \approx 32,768 \times 2^{128}$
- $\mathbf{z}$ is additive Gaussian noise

Practical issue

- Width of sensing matrix is huge
- Existing CS solvers will not execute at that scale

Matrix Width & Sparsity Undersampling Tradeoff



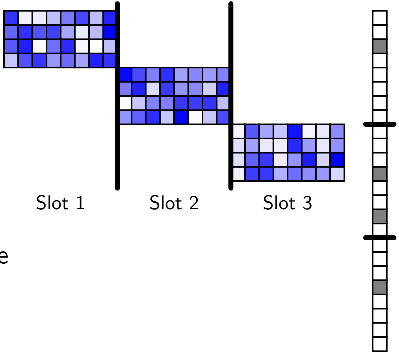
- Undersampling fraction

$$\delta = \frac{32,768}{2^{128}} = 2^{-113}$$

- Measure of sparsity

$$\rho = \frac{256}{32,768} = 2^{-7}$$

Time-Division Unsourced Random Access



Slot partitioning

- Observations become

$$\mathbf{y}_\ell = \mathbf{\Phi}_\ell \mathbf{s}_\ell + \mathbf{z}_\ell$$

  where ℓ is slot label
- Device gets slot based on message
- Channel uses divided among slots

Drawbacks

- Matrices remain wide $2^{128}/L$
- Devices assigned randomly within slots

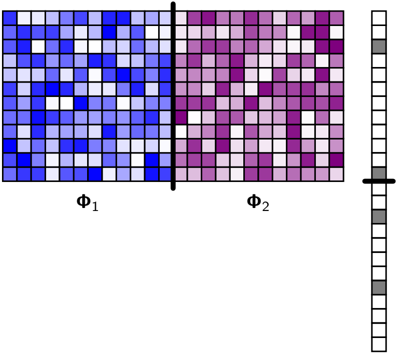
Classical Coding Techniques

Multi-User Coding

- Matrix becomes codebooks

$$\mathbf{y} = \mathbf{\Phi}_1 \mathbf{s}_1 + \mathbf{\Phi}_2 \mathbf{s}_2 + \mathbf{z}$$

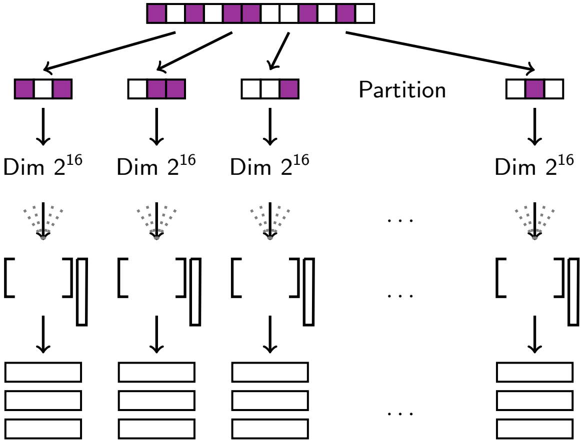
- Device picks code based on bits
- Well-studied for single user
- Fast decoding for large dictionary



Drawbacks

- Low complexity joint multi-user decoders are not available
- Devices may collide within codebook selection

Data Fragmentation



Drawbacks

- Unordered lists of fragments
- Need to perform disambiguation

Pertinent References

- Y. Polyanskiy. A perspective on massive random-access. *Proc. Int. Symp. on Information Theory (ISIT)*, 2017.
- O. Ordentlich and Y. Polyanskiy. Low complexity schemes for the random access Gaussian channel. *Proc. Int. Symp. on Information Theory (ISIT)*, 2017.
- A. Vem, K. R. Narayanan, J.-F. Chamberland, and J. Cheng. A user-independent successive interference cancellation based coding scheme for the unsourced random access Gaussian channel. *IEEE Trans. on Communications*, 2019.
- V. K. Amalladinne, J.-F. Chamberland, and K. R. Narayanan. A coded compressed sensing scheme for unsourced multiple access. *IEEE Trans. on Information Theory*, 2020.
- R. Calderbank and A. Thompson. CHIRRUP: A practical algorithm for unsourced multiple access. *Information and Inference*, December 2019.

A Quest for Low-Complexity: Coded Compressed Sensing

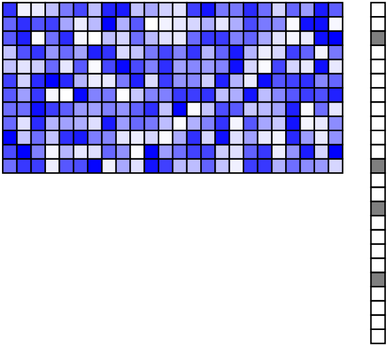
Abstract CS Challenge

Problem setting

- Noisy compressed sensing

$$\mathbf{y} = \mathbf{\Phi s} + \mathbf{z}$$

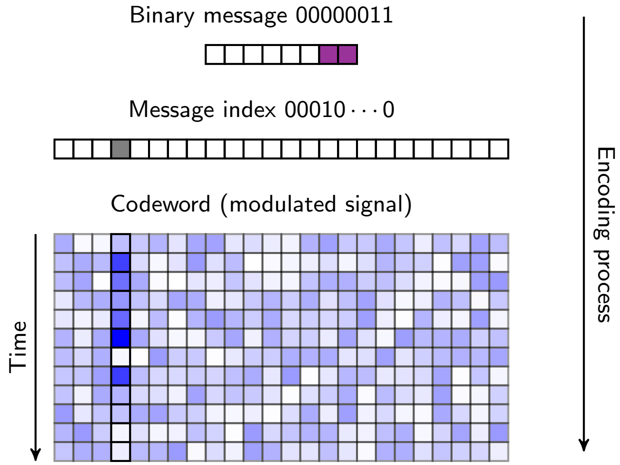
  where $\mathbf{s}$ is K sparse
- $\mathbf{s}$ has non-negative integer entries
- $\mathbf{\Phi}.\texttt{shape} \approx 32,768 \times 2^{128}$
- $\mathbf{z}$ is additive Gaussian noise



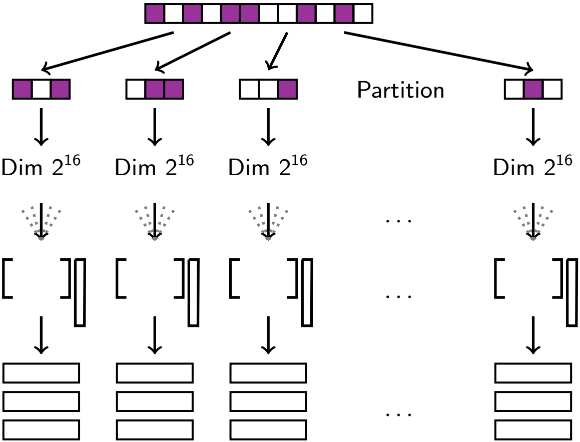
Practical issue and potential direction

- Width of sensing matrix is huge
- Undersampling fraction and sparsity are very small

Unsourced Random Access – Index Representation



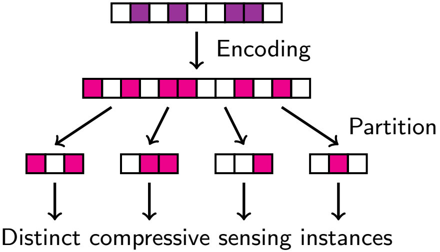
Data Fragmentation



Drawbacks

- Unordered lists of fragments
- Need to perform disambiguation

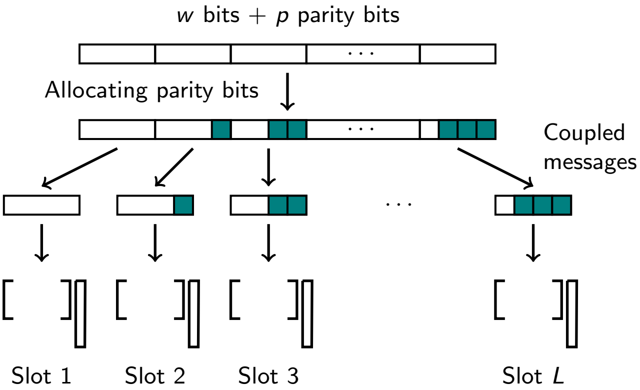
Fragmentation with Disambiguation



Stitching through outer code

- Split problem into sub-components suitable for CS framework
- Get lists of sub-packets, one list for every slot
- Stitch pieces of one packet together using error correction

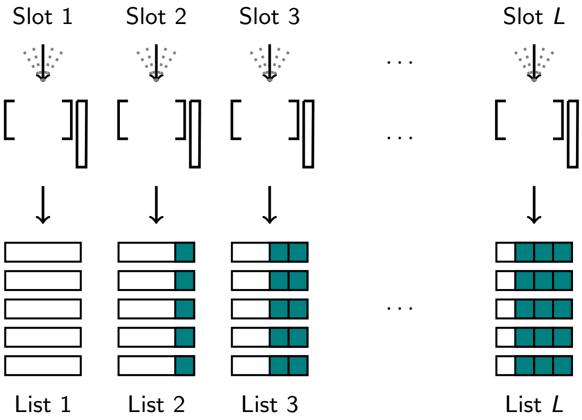
Coded Compressive Sensing – Device Perspective



- Collection of L CS matrices and 1-sparse vectors
- Each CS generated signal is sent in specific time slot

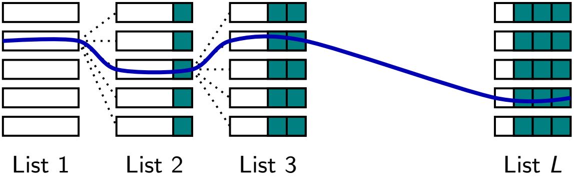
V. K. Amalladinne, J.-F. Chamberland, and K. R. Narayanan. *A coded compressed sensing scheme for unsourced multiple access*. IEEE Transactions on Information Theory, 2020.

Coded Compressive Sensing – Multiple Access



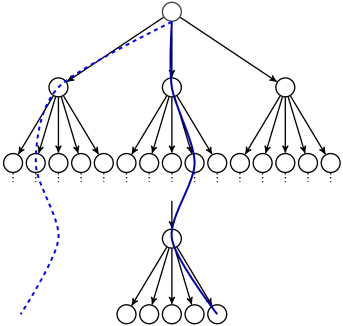
- L instances of CS problem, each solved with non-negative LS
- Produces L lists of K decoded sub-packets (with parity)
- Must piece sub-packets together using tree decoder

Coded Compressive Sensing – Stitching Process

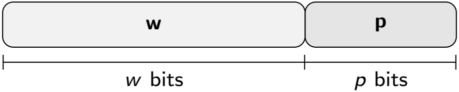


Tree decoding principles

- Every parity is linear combination of bits in preceding blocks
- Late parity bits offer better performance
- Early parity bits decrease decoding complexity
- Correct fragment is on list



Coded Compressive Sensing – Understanding Parity Bits



- Consider binary information vector $\mathbf{w}$ of length k
- Systematically encoded using generator matrix $\mathbf{G}$, with $\mathbf{p} = \mathbf{w}\mathbf{G}$
- Suppose alternate vector $\mathbf{w}_{\mathrm{r}}$ is selected at random from $\{0, 1\}^k$

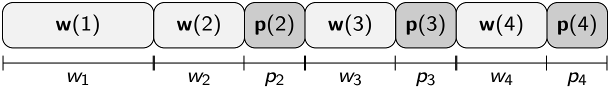
Lemma

Probability that randomly selected information vector $\mathbf{w}_{\mathrm{r}}$ produces same parity sub-component is given by

$$\Pr(\mathbf{p} = \mathbf{p}_{\mathrm{r}}) = 2^{-\operatorname{rank}(\mathbf{G})}$$

Proof: $\{\mathbf{p} = \mathbf{p}_{\mathrm{r}}\} = \{\mathbf{w}\mathbf{G} = \mathbf{w}_{\mathrm{r}}\mathbf{G}\} = \{\mathbf{w} + \mathbf{w}_{\mathrm{r}} \in \operatorname{nullspace}(\mathbf{G})\}$

Coded Compressive Sensing – General Parity Bits



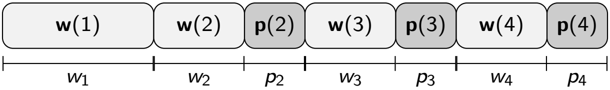
- True vector $(\mathbf{w}_{i_1}(1), \mathbf{w}_{i_1}(2), \mathbf{w}_{i_1}(3), \mathbf{w}_{i_1}(4))$
- Consider alternate vector with information sub-block $(\mathbf{w}_{i_1}(1), \mathbf{w}_{i_2}(2), \mathbf{w}_{i_3}(3), \mathbf{w}_{i_4}(4))$ pieced from lists
- To survive stage 4, candidate vector must fulfill parity equations

$$(\mathbf{w}_{i_1}(1) - \mathbf{w}_{i_2}(1)) \left[\mathbf{G}_{1,2} \right] = \mathbf{0}$$

$$(\mathbf{w}_{i_1}(1) - \mathbf{w}_{i_3}(1), \mathbf{w}_{i_2}(2) - \mathbf{w}_{i_3}(2)) \begin{bmatrix} \mathbf{G}_{1,3} \\ \mathbf{G}_{2,3} \end{bmatrix} = \mathbf{0}$$

$$(\mathbf{w}_{i_1}(1) - \mathbf{w}_{i_4}(1), \mathbf{w}_{i_2}(2) - \mathbf{w}_{i_4}(2), \mathbf{w}_{i_3}(3) - \mathbf{w}_{i_4}(3)) \begin{bmatrix} \mathbf{G}_{1,4} \\ \mathbf{G}_{2,4} \\ \mathbf{G}_{3,4} \end{bmatrix} = \mathbf{0}$$

Coded Compressive Sensing – General Parity Bits



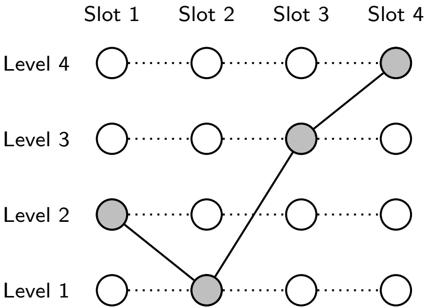
- When indices are not repeated in $(\mathbf{w}_{i_1}(1), \mathbf{w}_{i_2}(2), \mathbf{w}_{i_3}(3), \mathbf{w}_{i_4}(4))$, probability is governed by

$$\operatorname{rank}\left(\begin{bmatrix} \mathbf{G}_{1,2} & \mathbf{G}_{1,3} & \mathbf{G}_{1,4} \\ \mathbf{0} & \mathbf{G}_{2,3} & \mathbf{G}_{2,4} \\ \mathbf{0} & \mathbf{0} & \mathbf{G}_{3,4} \end{bmatrix} \right)$$

- But, when indices are repeated, sub-blocks may disappear

$$\operatorname{rank}\left(\begin{bmatrix} \mathbf{G}_{1,2}\mathbf{1}_{\{i_2 \neq i_1\}} & \mathbf{G}_{1,3}\mathbf{1}_{\{i_3 \neq i_1\}} & \mathbf{G}_{1,4}\mathbf{1}_{\{i_4 \neq i_1\}} \\ \mathbf{0} & \mathbf{G}_{2,3}\mathbf{1}_{\{i_3 \neq i_2\}} & \mathbf{G}_{2,4}\mathbf{1}_{\{i_4 \neq i_2\}} \\ \mathbf{0} & \mathbf{0} & \mathbf{G}_{3,4}\mathbf{1}_{\{i_4 \neq i_3\}} \end{bmatrix} \right)$$

Candidate Paths and Bell Numbers

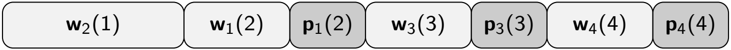


Probability that wrong path is consistent with parities is

$$\Pr(\mathbf{p} = \mathbf{p}_{\mathrm{r}}) = 2^{-\,\mathrm{rank}(\mathbf{G})}$$

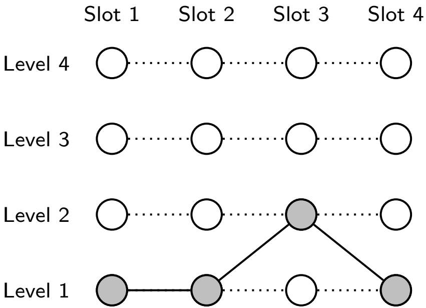
where

$$\mathbf{G} = \begin{bmatrix} \mathbf{G}_{1,2} & \mathbf{G}_{1,3} & \mathbf{G}_{1,4} \\ \mathbf{0} & \mathbf{G}_{2,3} & \mathbf{G}_{2,4} \\ \mathbf{0} & \mathbf{0} & \mathbf{G}_{3,4} \end{bmatrix}$$



When Levels Do NOT Repeat

Candidate Paths and Bell Numbers

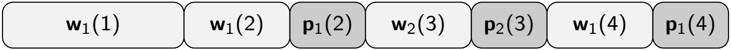


Probability that wrong path is consistent with parities is

$$\Pr(\mathbf{p} = \mathbf{p}_{\mathrm{r}}) = 2^{-\operatorname{rank}(\mathbf{G})}$$

where

$$\mathbf{G} = \begin{bmatrix} \mathbf{0} & \mathbf{G}_{1,3} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_{2,3} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{G}_{3,4} \end{bmatrix}$$

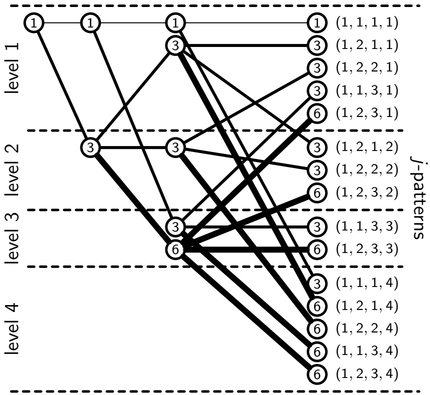


When Levels Repeat

Bell Numbers and *j*-patterns

Integer Sequences

- K^L paths
- Reduce complexity through equivalence
- Online Encyclopedia of Integer Sequences (OEIS) A000110
- Bell numbers grow rapidly
- Hard to compute expected number of surviving paths



Need Approximation

Allocating Parity Bits (approximation)

- p_ℓ: # parity bits in sub-block $\ell \in 2, \ldots, L$,
- P_ℓ: # erroneous paths that survive stage $\ell \in 2, \ldots, L$,
- Complexity C_{tree}: # nodes on which parity check constraints verified

Expressions for $\mathbb{E}[P_\ell]$ and C_{tree}

- $P_\ell | P_{\ell-1} \sim B((P_{\ell-1}+1)K-1, \rho_\ell)$, $\rho_\ell = 2^{-p_\ell}$, $q_\ell = 1-\rho_\ell$

$$\begin{aligned}
\mathbb{E}[P_\ell] &= \mathbb{E}[\mathbb{E}[P_\ell | P_{\ell-1}]] \\
&= \mathbb{E}[((P_{\ell-1}+1)K-1)\rho_\ell] \\
&= \rho_\ell K \mathbb{E}[P_{\ell-1}] + \rho_\ell (K-1) \\
&= \sum_{r=1}^{\ell} K^{\ell-r}(K-1) \prod_{j=r}^{\ell} \rho_j
\end{aligned}$$

- $C_{\text{tree}} = K + \sum_{\ell=2}^{L-1} [(P_\ell + 1)K]$
- $\mathbb{E}[C_{\text{tree}}]$ can be computed using the expression for $\mathbb{E}[P_\ell]$

Optimization of Parity Lengths

- p_ℓ: # parity bits in sub-block $\ell \in 2, \ldots, L$,
- P_ℓ: # erroneous paths that survive stage $\ell \in 2, \ldots, L$,

Relaxed geometric programming optimization

$$
\begin{aligned}
\underset{(p_2,\ldots,p_L)}{\text{minimize}} \quad & \mathbb{E}[C_{\text{tree}}] & \\
\text{subject to} \quad & \Pr(P_L \geq 1) \leq \varepsilon_{\text{tree}} & \text{Erroneous paths} \\
& \sum_{\ell=2}^{L} p_\ell = M - B & \text{Total \# parity bits} \\
& p_\ell \in \{0, \ldots, N/L\} \quad \forall\, \ell \in 2, \ldots, L & \text{Integer constraints}
\end{aligned}
$$

- Solved using standard convex solver, e.g., CVX

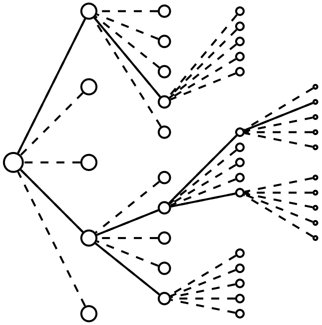
Choice of Parity Lengths

- $K = 200$, $L = 11$, $N/L = 15$

$\varepsilon_{\text{tree}}$	$\mathbb{E}[C_{\text{tree}}]$	Parity Lengths $p_2, \ldots, p_L$
0.006	Infeasible	Infeasible
0.0061930	3.2357×10^{11}	0, 0, 0, 0, 15, 15, 15, 15, 15, 15
0.0061931	3357300	0, 3, 8, 8, 8, 8, 10, 15, 15, 15
0.0061932	1737000	0, 4, 8, 8, 8, 8, 9, 15, 15, 15
0.0061933	926990	0, 5, 8, 8, 8, 8, 8, 15, 15, 15
0.0061935	467060	1, 8, 8, 8, 8, 8, 8, 11, 15, 15
0.0062	79634	1, 8, 8, 8, 8, 8, 8, 11, 15, 15
0.007	7357.8	6, 8, 8, 8, 8, 8, 8, 8, 13, 15
0.008	6152.7	7, 8, 8, 8, 8, 8, 8, 8, 12, 15
0.02	5022.9	6, 8, 8, 9, 9, 9, 9, 9, 9, 14
0.04	4158	7, 8, 8, 9, 9, 9, 9, 9, 9, 13
0.6378	3066.3	9, 9, 9, 9, 9, 9, 9, 9, 9, 9

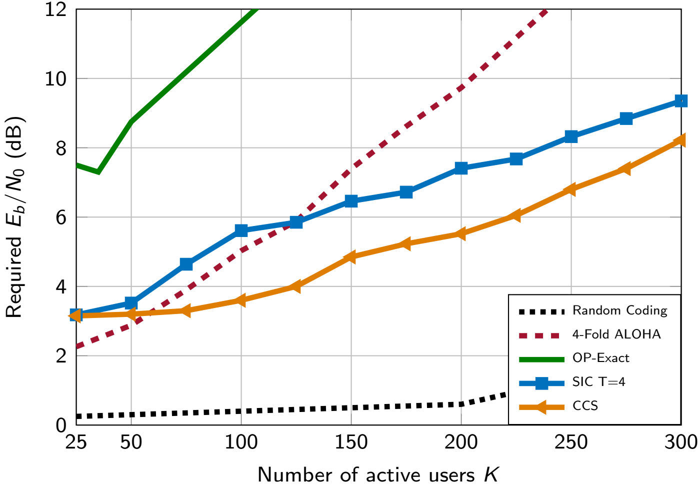
Choice of Parity Lengths

- $K = 200$, $L = 11$, $N/L = 15$



Parity Lengths $p_2, \ldots, p_L$
$0, 0, 0, 0, 15, 15, 15, 15, 15, 15$
$0, 3, 8, 8, 8, 8, 10, 15, 15, 15$
$0, 4, 8, 8, 8, 8, 9, 15, 15, 15$
$0, 5, 8, 8, 8, 8, 8, 15, 15, 15$
$1, 8, 8, 8, 8, 8, 8, 11, 15, 15$
$1, 8, 8, 8, 8, 8, 8, 11, 15, 15$
$6, 8, 8, 8, 8, 8, 8, 8, 13, 15$
$7, 8, 8, 8, 8, 8, 8, 8, 12, 15$
$6, 8, 8, 9, 9, 9, 9, 9, 9, 14$
$7, 8, 8, 9, 9, 9, 9, 9, 9, 13$
$9, 9, 9, 9, 9, 9, 9, 9, 9, 9$

Performance of CCS and Previous Schemes



Leveraging CCS Framework

CHIRRUP: a practical algorithm for unsourced multiple access

Robert Calderbank, Andrew Thompson

(Submitted on 2 Nov 2018)

Unsourced multiple access abstracts grantless simultaneous communication of a large number of devices (messages) each of which transmits (is transmitted) infrequently. It provides a model for machine-to-machine communication in the Internet of Things (IoT), including the special case of radio-frequency identification (RFID), as well as neighbor discovery in ad hoc wireless networks. This paper presents a fast algorithm for unsourced multiple access that scales to 2^{100} devices (arbitrary 100 bit messages). The primary building block is multiuser detection of binary chirps which are simply codewords in the second order Reed Muller code. The chirp detection algorithm originally presented by Howard et al. is enhanced and integrated into a peeling decoder designed for a patching and slotting framework. In terms of both energy per bit and number of transmitted messages, the proposed algorithm is within a factor of 2 of state of the art approaches. A significant advantage of our algorithm is its computational efficiency. We prove that the worst-case complexity of the basic chirp reconstruction algorithm is $\mathcal{O}[nK(\log_2 n + K)]$, where n is the codeword length and K is the number of active users, and we report computing times for our algorithm. Our performance and computing time results represent a benchmark against which other practical algorithms can be measured.

Subjects: **Signal Processing (eess.SP)**
Cite as: **arXiv:1811.00879 [eess.SP]**
(or **arXiv:1811.00879v1 [eess.SP]** for this version)

Submission history
From: Andrew Thompson [view email]
[v1] Fri, 2 Nov 2018 14:25:46 UTC (470 KB)

- Hadamard matrix based compressing scheme + CSS
- Ultra-low complexity decoding algorithm

S. D. Howard, A. R. Calderbank, S. J. Searle. *A Fast Reconstruction Algorithm for Deterministic Compressive Sensing using Second Order Reed-Muller Codes.* CISS 2008

Example: CHIRRUP

- Sensing matrix based on 2nd-order Reed-Muller functions,

$$\phi_{R,b}(a) = \frac{(-1)^{\mathrm{wt}(b)}}{\sqrt{2^m}} i^{(2b+Ra)^T a}$$

  R is binary symmetric matrix with zeros on diagonal, wt represent weight, and $i = \sqrt{-1}$

- Every column of form

$$\begin{matrix} | \\ \mathbf{x}_{R,b} \\ | \end{matrix} = \begin{bmatrix} \phi_{R,b}([0]_2) \\ \phi_{R,b}([1]_2) \\ \vdots \\ \phi_{R,b}([2^m - 1]_2) \end{bmatrix}$$

  $[\cdot]_2$ is integer expressed in radix of 2

- Information encoded into R and b
- **Fast recovery:** Inner-products, Hardmard project onto Walsh basis, get R row column at a time, dechirp, Hadamard project to b

Enhanced Coded Compressed Sensing

An enhanced decoding algorithm for coded compressed sensing

Vamsi K. Amalladinne, Jean-Francois Chamberland, Krishna R. Narayanan

Coded compressed sensing is an algorithmic framework tailored to sparse recovery in very large dimensional spaces. This framework is originally envisioned for the unsourced multiple access channel, a wireless paradigm attuned to machine-type communications. Coded compressed sensing uses a divide-and-conquer approach to break the sparse recovery task into sub-components whose dimensions are amenable to conventional compressed sensing solvers. The recovered fragments are then stitched together using a low complexity decoder. This article introduces an enhanced decoding algorithm for coded compressed sensing where fragment recovery and the stitching process are executed in tandem, passing information between them. This novel scheme leads to gains in performance and a significant reduction in computational complexity. This algorithmic opportunity stems from the realization that the parity structure inherent to coded compressed sensing can be used to dynamically restrict the search space of the subsequent recovery algorithm.

Comments: Submitted to ICASSP2020
Subjects: **Information Theory (cs.IT)**; Signal Processing (eess.SP)
Cite as: arXiv:1910.09704 **[cs.IT]**
(or arXiv:1910.09704v1 **[cs.IT]** for this version)

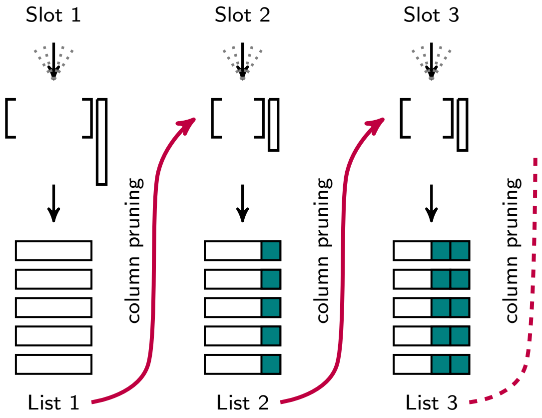
Bibliographic data
[Enable Bibex (What is Bibex?)]

Submission history
From: Vamsi Amalladinne [view email]
[v1] Tue, 22 Oct 2019 00:17:37 UTC (65 KB)

Leverage algorithmic opportunity

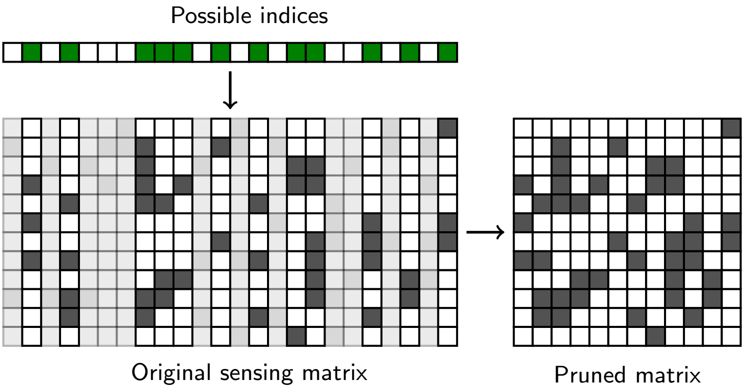
- Extending CCS framework by integrating tree code
- Decisions at early stages inform later parts
- Algorithmic performance improvements

Coded Compressive Sensing with Column Pruning



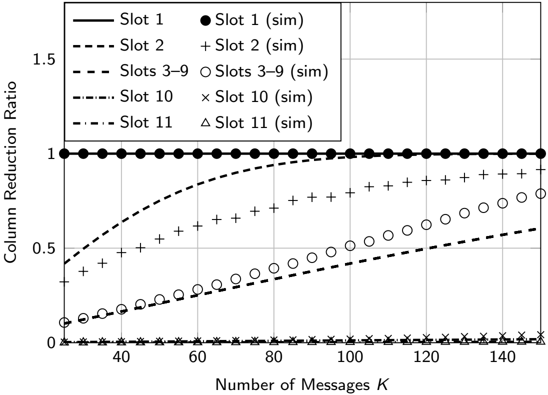
- Active partial paths determine possible parity patterns
- Admissible indices for next slot determined by possible parities
- Inadmissible columns can be pruned before CS algorithm

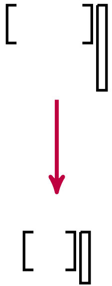
Coded Compressive Sensing with Column Pruning



- For K small, width of sensing matrix is greatly reduced
- Actual sensing matrix is determined dynamically at run time
- Complexity of CS algorithm becomes much smaller

Expected Column Reduction Ratio





- Parity allocation parameters, with $w_\ell + p_\ell = 15$,

$$(p_1, p_2, \ldots, p_{10}) = (6, 8, 8, 8, 8, 8, 8, 8, 13, 15)$$

- Pruning is more pronounced at later stages
- Effective width of sensing matrix is greatly reduced

Leveraging CCS Framework

Non-Bayesian Activity Detection, Large-Scale Fading Coefficient Estimation, and Unsourced Random Access with a Massive MIMO Receiver

Alexander Fengler, Saeid Haghighatshoar, Peter Jung, Giuseppe Caire

In this paper, we study the problem of user activity detection and large-scale fading coefficient estimation in a random access wireless uplink with a massive MIMO base station with a large number M of antennas and a large number of wireless single-antenna devices (users). We consider a block fading channel model where the M-dimensional channel vector of each user remains constant over a coherence block containing L signal dimensions in time-frequency. In the considered setting, the number of potential users K_{tot} is much larger than L but at each time slot only $K_a << K_{\text{tot}}$ of them are active. Previous results, based on compressed sensing, require that $K_a \leq L$, which is a bottleneck in massive deployment scenarios such as Internet-of-Things and unsourced random access. In this work we show that such limitation can be overcome when the number of base station antennas M is sufficiently large. We also provide two algorithms. One is based on Non-Negative Least-Squares, for which the above scaling result can be rigorously proved. The other consists of a low-complexity iterative componentwise minimization of the likelihood function of the underlying problem. Finally, we use the proposed approximated ML algorithm as the decoder for the inner code in a concatenated coding scheme for unsourced random access, where all users make use of the same codebook, and the massive MIMO base station must come up with the list of transmitted messages irrespectively of the identity of the transmitters. We show that reliable communication is possible at any E_b/N_0 provided that a sufficiently large number of base station antennas is used, and that a sum spectral efficiency in the order of $\mathcal{O}(L\log(L))$ is achievable.

Comments: 50 pages, 8 figures, submitted to IEEE Trans. Inf. Theory
Subjects: **Information Theory (cs.IT)**
Cite as: arXiv:1910.11266 **[cs.IT]**
(or arXiv:1910.11266v1 **[cs.IT]** for this version)

Bibliographic data

[Enable Bibex (What is Bibex?)]

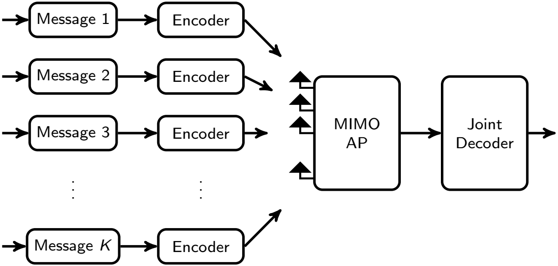
Submission history

From: Alexander Fengler [view email]
[v1] Thu, 24 Oct 2019 16:32:30 UTC (661 KB)

Which authors of this paper are endorsers? | Disable MathJax (What is MathJax?)

- Activity detection in random access
- Massive MIMO Receiver

Massive MIMO-URA



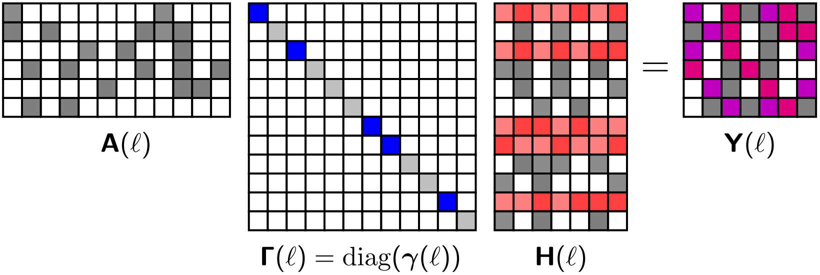
Signal model

- Signal received at time instant t with slot ℓ

$$\mathbf{y}(t,\ell) = \sum_{k=1}^{K} \mathbf{x}_k(t,\ell)\mathbf{h}_k(\ell) + \mathbf{z}(t,\ell)$$

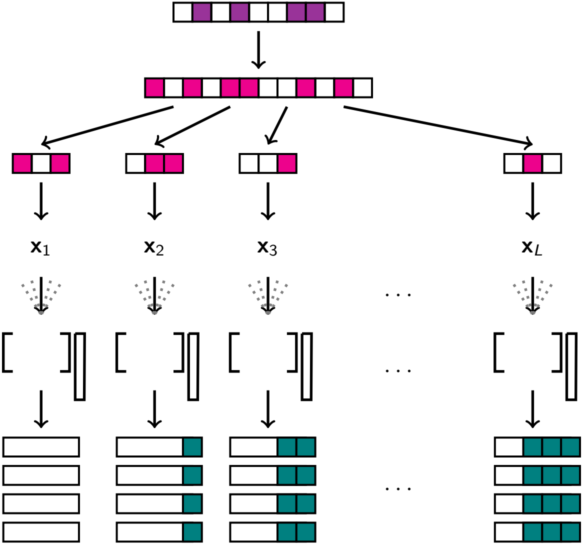
- Number of receive antennas $M \gg 1$
- Block fading – channel does not change within CCS slot
- Spatial correlation negligible – $\mathbf{h}_k(\ell) \sim \mathcal{CN}(0, \mathbf{I}_M)$

Multiple Measurement Vector – CS Interpretation



- Received signal during slot ℓ: $\mathbf{Y}(\ell) = \mathbf{A}(\ell)\mathbf{\Gamma}(\ell)\mathbf{H}(\ell) + \mathbf{Z}(\ell)$
- Column $\mathbf{y}_i(\ell)$ of $\mathbf{Y}(\ell)$ is the signal received at antenna i during slot ℓ
- $\mathbf{H}(\ell)$ has entries drawn i.i.d. from $\mathcal{CN}(0, 1)$

Coded Compressed Sensing – Summary

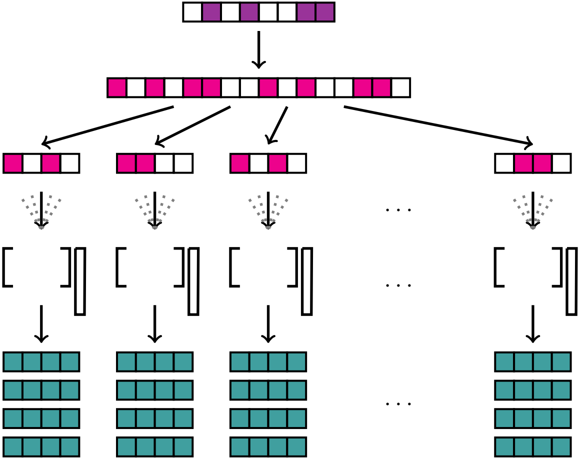


Pertinent References

- V. K. Amalladinne, J.-F. Chamberland, and K. R. Narayanan. A coded compressed sensing scheme for unsourced multiple access. *IEEE Trans. on Information Theory*, 2020.
- R. Calderbank and A. Thompson. CHIRRUP: A practical algorithm for unsourced multiple access. *Information and Inference: A Journal of the IMA*, 2018.
- V. K. Amalladinne, J.-F. Chamberland, and K. R. Narayanan. An enhanced decoding algorithm for coded compressed sensing. In *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, May 2020.
- A. Fengler, S. Haghighatshoar, P. Jung, and G. Caire. Non-Bayesian activity detection, large-scale fading coefficient estimation, and unsourced random access with a massive MIMO receiver. *IEEE Trans. on Information Theory*, 2021.

Connecting Coding and Compressed Sensing via Approximate Message Passing

Coded Compressive Sensing – Divide and Conquer



- Data fragmentation and indexing
- Outer encoding for disambiguation

CCS – Approximate Message Passing

SPARCs for Unsourced Random Access

Alexander Fengler, Peter Jung, Giuseppe Caire

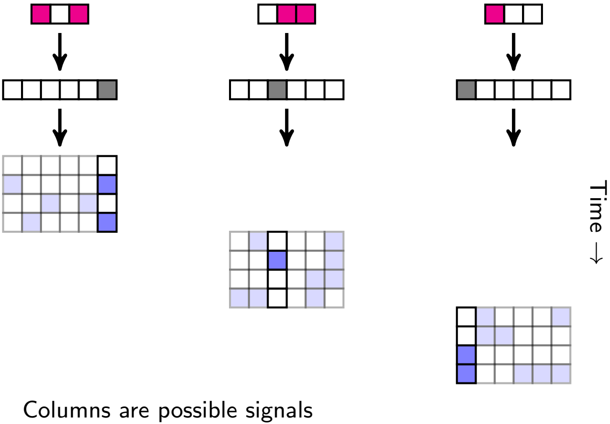
(Submitted on 18 Jan 2019)

This paper studies the optimal achievable performance of compressed sensing based unsourced random-access communication over the real AWGN channel. "Unsourced" means, that every user employs the same codebook. This paradigm, recently introduced by Polyanskiy, is a natural consequence of a very large number of potential users of which only a finite number is active in each time slot. The idea behind compressed sensing based schemes is that each user encodes his message into a sparse binary vector and compresses it into a real or complex valued vector using a random linear mapping. When each user employs the same matrix this creates an effective binary inner multiple-access channel. To reduce the complexity to an acceptable level the messages have to be split into blocks. An outer code is used to assign the symbols to individual messages. This division into sparse blocks is analogous to the construction of sparse regression codes (SPARCs), a novel type of channel codes, and we can use concepts from SPARCs to design efficient random-access codes. We analyze the asymptotically optimal performance of the inner code using the recently rigorized replica symmetric formula for the free energy which is achievable with the approximate message passing (AMP) decoder with spatial coupling. An upper bound on the achievable rates of the outer code is derived by classical Shannon theory. Together this establishes a framework to analyse the trade-off between SNR, complexity and achievable rates in the asymptotic infinite blocklength limit. Finite blocklength simulations show that the combination of AMP decoding, with suitable approximations, together with an outer code recently proposed by Amalladinne et. al. outperforms state of the art methods in terms of required energy-per-bit at lower decoding complexity.

Comments: 16 pages, 7 Figures
Subjects: **Information Theory (cs.IT)**
Cite as: **arXiv:1901.06234 [cs.IT]**
(or **arXiv:1901.06234v1 [cs.IT]** for this version)

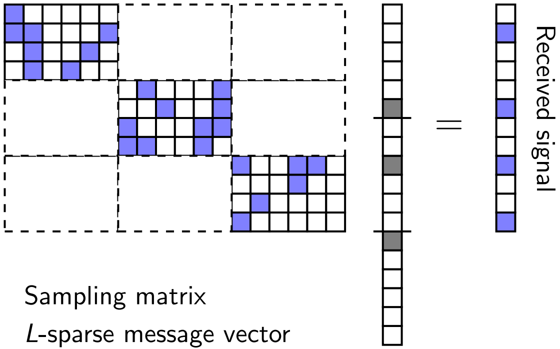
- Connection between CCS indexing and sparse regression codes
- Circumvent slotting under CCS and dispersion effects
- Introduce denoiser tailored to CCS

CCS Revisited



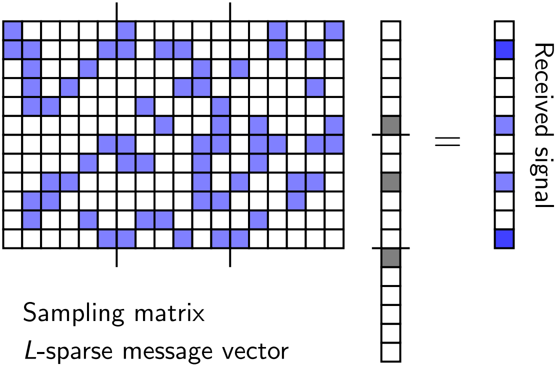
- Bit sequence split into L fragments
- Each bit + parity block converted to index in $[0, 2^{m/L} - 1]$
- Stack sub-codewords into $(n/L) \times 2^{m/L}$ sensing matrices

Coded Compressed Sensing – Unified View



- Slots produce block diagonal (unified) matrix
- Message is one-sparse per section
- Width of $\mathbf{A}$ is smaller: $L2^{m/L}$ instead of 2^m

CCS – Full Sensing Matrix



- Complexity reduction due to narrower **A**
- Use full sensing matrix **A**
- Decode inner code with low-complexity AMP

CCS – Approximate Message Passing

Governing Equations

- AMP algorithm iterates through

$$\mathbf{z}^{(t)} = \mathbf{y} - \mathbf{A}\mathbf{D}\boldsymbol{\eta}_t\left(\mathbf{r}^{(t)}\right) + \underbrace{\frac{\mathbf{z}^{(t-1)}}{n} \operatorname{div} \mathbf{D}\boldsymbol{\eta}_t\left(\mathbf{r}^{(t)}\right)}_{\text{Onsager correction}}$$

$$\mathbf{r}^{(t+1)} = \mathbf{A}^{\mathsf{T}}\mathbf{z}^{(t)} + \mathbf{D}\underbrace{\boldsymbol{\eta}_t\left(\mathbf{r}^{(t)}\right)}_{\text{Denoiser}}$$

Initial conditions $\mathbf{z}^{(0)} = \mathbf{0}$ and $\eta_0\left(\mathbf{r}^{(0)}\right) = \mathbf{0}$

- Application falls within framework for non-separable functions

Task

- Define denoiser and compute Onsager correction term

Marginal Posterior Mean Estimate (PME)

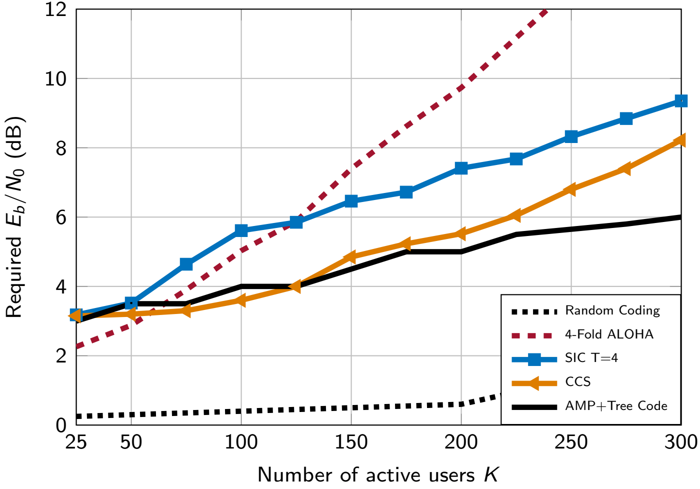
Proposed Denoiser (Fengler, Jung, and Caire)

- State estimate based on Gaussian model

$$\begin{aligned} \hat{s}^{\mathrm{OR}}(q, r, \tau) &= \mathbb{E}\left[s \middle| \sqrt{P_\ell} s + \tau \zeta = r \right] \\ &= \frac{q \exp\left(-\frac{\left(r - \sqrt{P_\ell}\right)^2}{2\tau^2} \right)}{(1-q)\exp\left(-\frac{r^2}{2\tau^2} \right) + q \exp\left(-\frac{\left(r - \sqrt{P_\ell}\right)^2}{2\tau^2} \right)} \end{aligned}$$

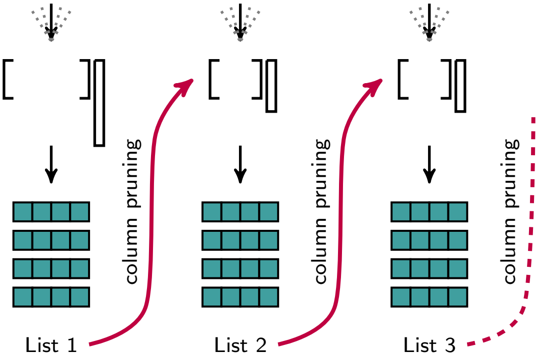
  with (essentially) uninformative prior $q = K/m$ fixed
- $\boldsymbol{\eta}_t\left(\mathbf{r}^{(t)}\right)$ is aggregate of PME values
- τ_t is obtained from state evolution or $\tau_t^2 = \|\mathbf{z}^{(t)}\|^2/n$

Performance of CCS-AMP versus Previous Schemes



Incorporating Lessons from Enhanced CCS

- Integrate outer code structure into inner decoding



Challenges

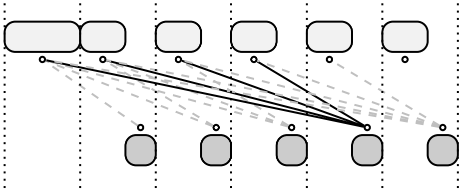
- CCS-AMP inner decoding is not a sequence of hard decisions
- List size for CCS-AMP is effective length of index vector

V. K. Amalladinne, A. K. Pradhan, C. Rush, J.-F. Chamberland, K. R. Narayanan. *On approximate message passing for unsourced access with coded compressed sensing.* ISIT 2020

Redesigning Outer Code

Properties of Original Outer Code

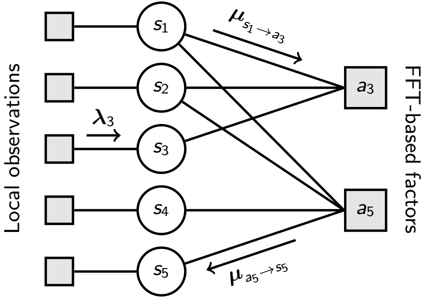
- Aimed at stitching message fragments together
- Works on short lists of K fragments
- Parities allocated to control growth and complexity



Challenges to Integrate into AMP

1. Must compute beliefs for all possible 2^v fragments
2. Must provide pertinent information to inner AMP decoder
3. Should maintain ability to stitch outer code

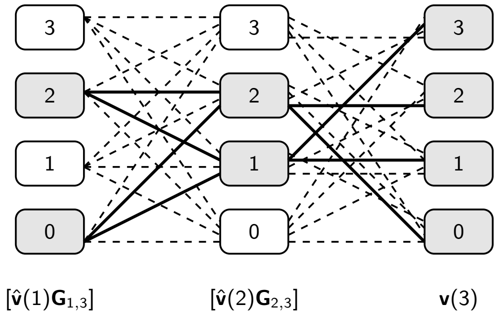
Factor Graph Interpretation of Outer Code



- Outer code with circular convolution structure

$$\boldsymbol{\mu}_{a_p \to s_\ell}\left([\hat{\mathbf{v}}(\ell)]_2\right) \propto \frac{1}{\left\|\mathbf{g}_{\ell,p}^{(g)}\right\|_0} \left(\mathsf{FFT}^{-1} \left(\prod_{s_j \in N(a_p) \setminus s_\ell} \mathsf{FFT}\left(\boldsymbol{\lambda}_{j,p}\right) \right) \right)(g)$$

Outer Code and Mixing

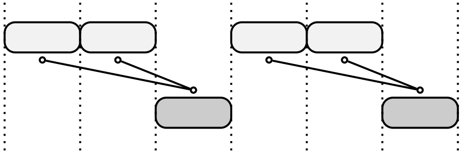


- Multiple devices on same graph
- Parity factor mix concentrated values
- Suggests triadic outer structure

Redesigning Outer Code

Solutions to Integrate into AMP

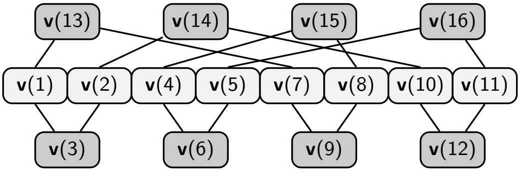
- Parity bits are generated over Abelian group amenable to FWHT or FFT
- Discrimination power proportional to # parities



New Design Strategy

1. Information sections with parity bits interspersed in-between
2. Parity over two blocks (triadic dependencies)

Belief Propagation – Message Passing Rules



- Message from check node a_p to variable node $s \in N(a_p)$:

$$\boldsymbol{\mu}_{a_p \to s}(k) = \sum_{\mathbf{k}_{a_p}: k_p = k} \mathcal{G}_{a_p}\left(\mathbf{k}_{a_p}\right) \prod_{s_j \in N(a_p) \setminus s} \boldsymbol{\mu}_{s_j \to a_p}(k_j)$$

- Message from variable node s_ℓ to check node $a \in N(s)$:

$$\boldsymbol{\mu}_{s_\ell \to a}(k) \propto \boldsymbol{\lambda}_\ell(k) \prod_{a_p \in N(s_\ell) \setminus a} \boldsymbol{\mu}_{a_p \to s_\ell}(k)$$

- Estimated marginal distribution

$$p_{s_\ell}(k) \propto \boldsymbol{\lambda}_\ell(k) \prod_{a \in N(s_\ell)} \boldsymbol{\mu}_{a \to s_\ell}(k)$$

Approximate Message Passing Algorithm

Updated Equations

AMP two-step algorithm

$$\mathbf{z}^{(t)} = \mathbf{y} - \mathbf{A}\mathbf{D}\eta_t\left(\mathbf{r}^{(t)}\right) + \underbrace{\frac{\mathbf{z}^{(t-1)}}{n} \operatorname{div} \mathbf{D}\eta_t\left(\mathbf{r}^{(t)}\right)}_{\text{Correction}}$$

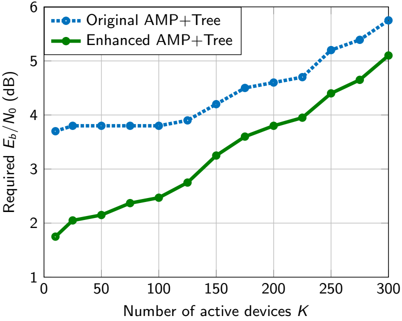
$$\mathbf{r}^{(t+1)} = \mathbf{A}^{\mathsf{T}}\mathbf{z}^{(t)} + \mathbf{D}\,\underbrace{\eta_t\left(\mathbf{r}^{(t)}\right)}_{\text{Denoiser}}$$

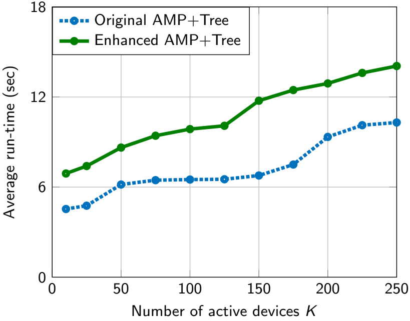
Initial conditions $\mathbf{z}^{(0)} = \mathbf{0}$ and $\eta_0\left(\mathbf{r}^{(0)}\right) = \mathbf{0}$

- Denoiser is BP estimate from factor graph
- Message passing uses fresh effective observation $\mathbf{r}$
- Fewer rounds than shortest cycle on factor graph
- Close to PME, but incorporating beliefs from outer code

R. Berthier, A. Montanari, and P.-M. Nguyen. *State Evolution for Approximate Message Passing with Non-Separable Functions*. Information and Inference: A Journal of the IMA 2020

Preliminary Performance Enhanced CCS



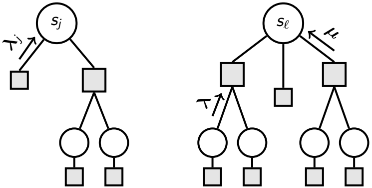


- Performance improves significantly with enhanced CCS-AMP decoding
- Computational complexity is approximately maintained
- Reparametrization may offer additional gains in performance?

CCS and AMP Summary

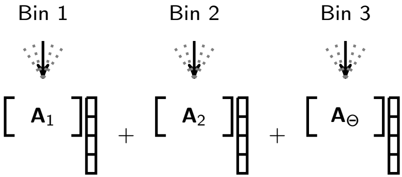
Summary

- New connection between CCS and AMP
- Natural application of BP on factor graph as denoiser
- Outer code design depends on sparsity
  1. Degree distributions (small graph)
  2. Message size (birthday problem)
  3. Final step is disambiguation
- Many theoretical and practical challenges/opportunities exist

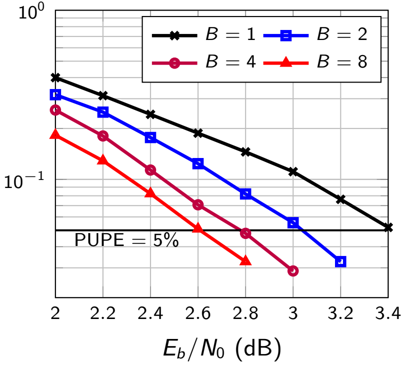


Coding plays increasingly central role in large-scale CS

Coded Demixing for Single-Class URA



- Create multiple bins with (incoherent) matrices
- Devices pick a bucket randomly and use CCS-AMP encoding
- Perform joint demixing CCS-AMP decoding at access point



J. R. Ebert, V. K. Amalladinne, S. Rini, J.-F. Chamberland, K. R. Narayanan. *Stochastic Binning and Coded Demixing for Unsourced Random Access.* arXiv:2104.05686

Pertinent References

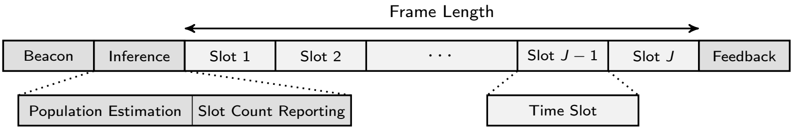
- A. Fengler, P. Jung, and G. Caire. SPARCs and AMP for Unsourced Random Access. In *International Symposium on Information Theory (ISIT)*, 2019.
- V. K. Amalladinne, A. K. Pradhan, C. Rush, J.-F. Chamberland, K. R. Narayanan. On approximate message passing for unsourced access with coded compressed sensing. In *International Symposium on Information Theory (ISIT)*, 2020.
- V. K. Amalladinne, A. Hao, S. Rini, J.-F. Chamberland. Multi-Class Unsourced Random Access via Coded Demixing. In *International Symposium on Information Theory (ISIT)*, 2021.
- A. Joseph, and A. R. Barron. Least squares superposition codes of moderate dictionary size are reliable at rates up to capacity *IEEE Trans. on Information Theory*, 2012.
- C. Rush, A. Greig, and R. Venkataramanan. Capacity-achieving sparse superposition codes via approximate message passing decoding. *IEEE Trans. on Information Theory*, 2017.
- R. Berthier, A. Montanari, and P.-M. Nguyen. State Evolution for Approximate Message Passing with Non-Separable Functions. *Information and Inference: A Journal of the IMA*, 2020.

What this part is about

- Review of Slotted ALOHA with interference cancellation
- Extension to the Unsourced Gaussian MAC
- Sparse IDMA for Unsourced multiple access

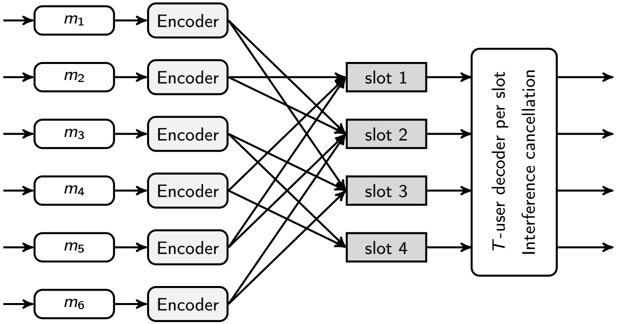
Uncoordinated MAC Frame Structure

- K active devices out of many, many devices



- Beacon employed for coarse synchronization
- Same devices transmit within frame
- Focus is on what happens within the Frame Length
- Each device may or may not use slots within the frame

Unsourced MAC – SIC UGMAC Scheme

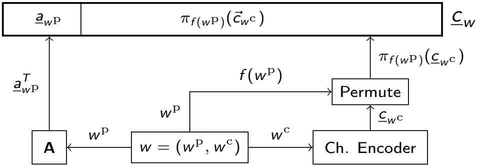


Key Features

- Schedule selected based on **message bits**
- Devices can transmit in multiple sub-blocks
- Scheme facilitates peeling decoder

A. Vem, K. Narayanan, J. Cheng, JFC. *A User-Independent Successive Interference Cancellation Based Coding Scheme for the Unsourced Random Access Gaussian Channel*. IEEE Trans on Comm, 2019

What Happens within a Slot?

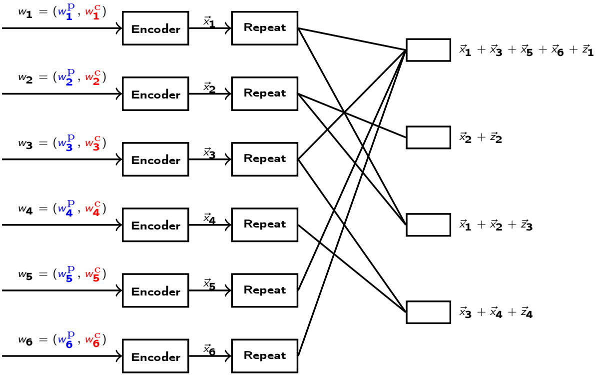


Implementation Notes

- Message is partitioned into two parts $w = (w_p, w_c)$
- Every device uses identical codebook built from LDPC-type codes tailored to T-user real-adder channel
- w_p dictate permutation on encoder and recovered through CS
- Non-negative ℓ_1-regularized LASSO

A. Vem, K. Narayanan, J. Cheng, JFC. *A User-Independent Successive Interference Cancellation Based Coding Scheme for the Unsourced Random Access Gaussian Channel*. IEEE Trans on Comm, 2019

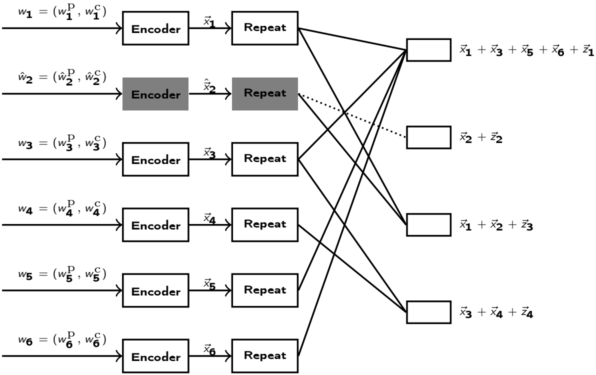
Unsourced MAC – SIC UGMAC Scheme for $T = 2$



Key Features

- Devices repeat codewords in multiple slots based on w_p
- Schedule selected based on **message bits**
- Scheme facilitates peeling decoder

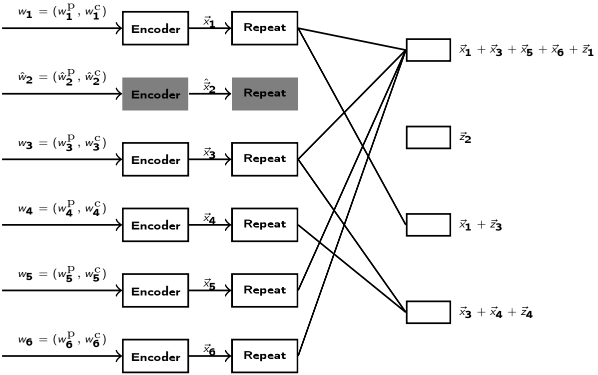
Unsourced MAC – SIC UGMAC Scheme for $T = 2$



Key Features

- Devices repeat codewords in multiple slots based on w_p
- Schedule selected based on **message bits**
- Scheme facilitates peeling decoder

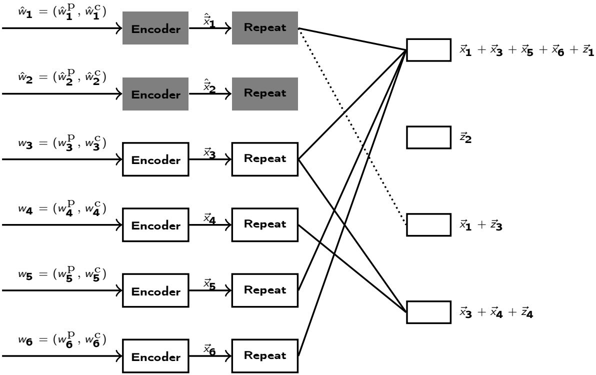
Unsourced MAC – SIC UGMAC Scheme for $T = 2$



Key Features

- Devices repeat codewords in multiple slots based on w_p
- Schedule selected based on **message bits**
- Scheme facilitates peeling decoder

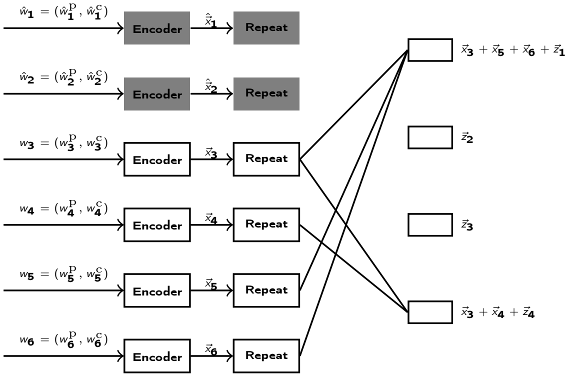
Unsourced MAC – SIC UGMAC Scheme for $T = 2$



Key Features

- Devices repeat codewords in multiple slots based on w_p
- Schedule selected based on **message bits**
- Scheme facilitates peeling decoder

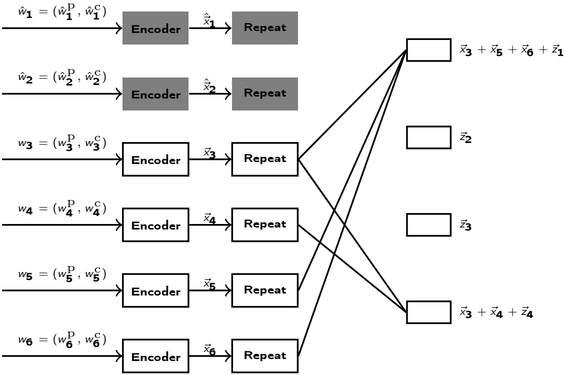
Unsourced MAC – SIC UGMAC Scheme for $T = 2$



Key Features

- Devices repeat codewords in multiple slots based on w_p
- Schedule selected based on **message bits**
- Scheme facilitates peeling decoder

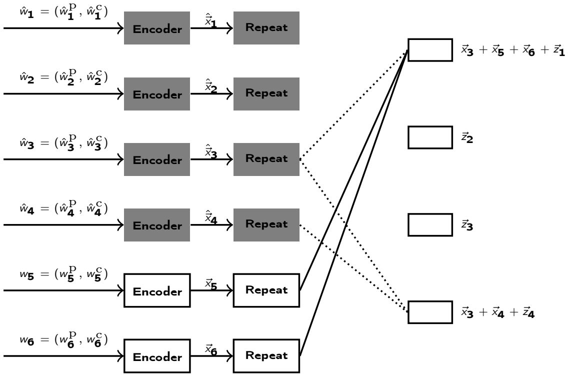
Unsourced MAC – SIC UGMAC Scheme for $T = 2$



Key Features

- Devices repeat codewords in multiple slots based on w_p
- Schedule selected based on **message bits**
- Scheme facilitates peeling decoder

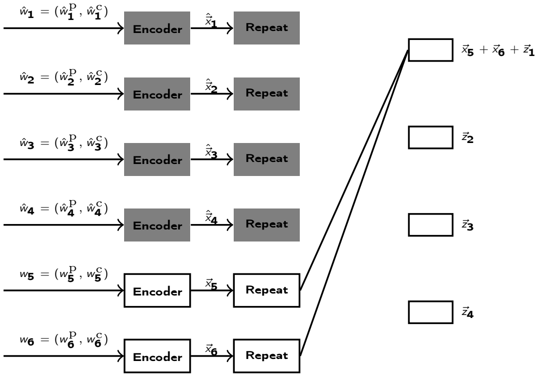
Unsourced MAC – SIC UGMAC Scheme for $T = 2$



Key Features

- Devices repeat codewords in multiple slots based on w_p
- Schedule selected based on **message bits**
- Scheme facilitates peeling decoder

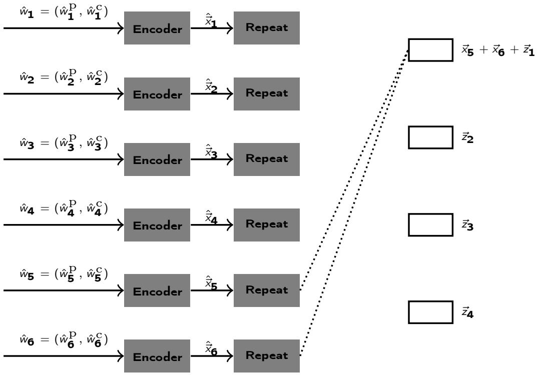
Unsourced MAC – SIC UGMAC Scheme for $T = 2$



Key Features

- Devices repeat codewords in multiple slots based on w_p
- Schedule selected based on **message bits**
- Scheme facilitates peeling decoder

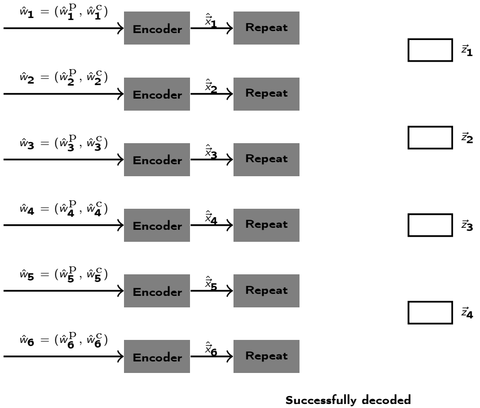
Unsourced MAC – SIC UGMAC Scheme for $T = 2$



Key Features

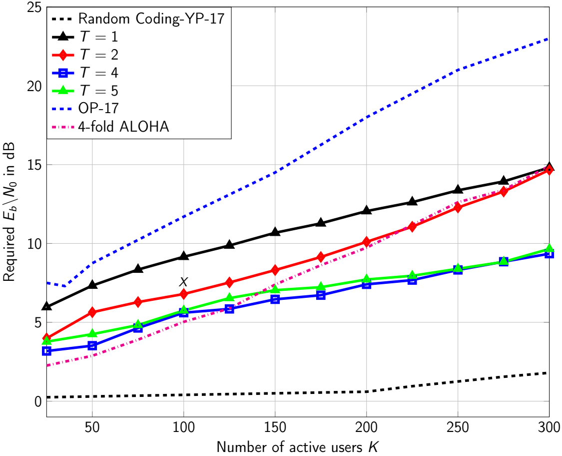
- Devices repeat codewords in multiple slots based on w_p
- Schedule selected based on **message bits**
- Scheme facilitates peeling decoder

Unsourced MAC – SIC UGMAC Scheme for $T = 2$



Key Features

- Devices repeat codewords in multiple slots based on w_p
- Schedule selected based on **message bits**
- Scheme facilitates peeling decoder

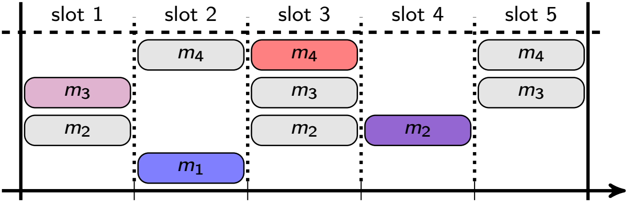


Random Coding-YP-17
$T = 1$
$T = 2$
$T = 4$
$T = 5$
OP-17
4-fold ALOHA
Required $E_b \backslash N_0$ in dB
Number of active users K
X
0
5
10
15
20
25
50
100
150
200
250
300

Limitations of Sparsifying Collisions

Drawbacks of Slots

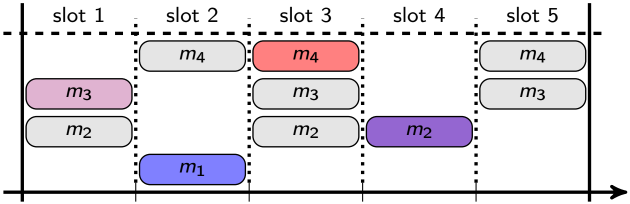
- Second order dispersion effects comes into play in FBL
- Energy expended solely to resolving collisions
- Gray slots are discarded during decoding process (60%)



Limitations of Sparsifying Collisions

Drawbacks of Slots

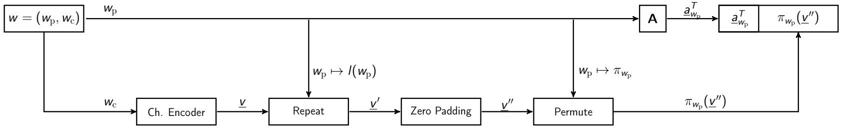
- Second order dispersion effects comes into play in FBL
- Energy expended solely to resolving collisions
- Gray slots are discarded during decoding process (60%)



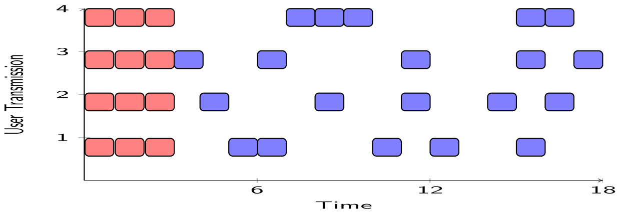
To fix this - Sparse IDMA

An IDMA like scheme which does not divide the number of channel uses into slots

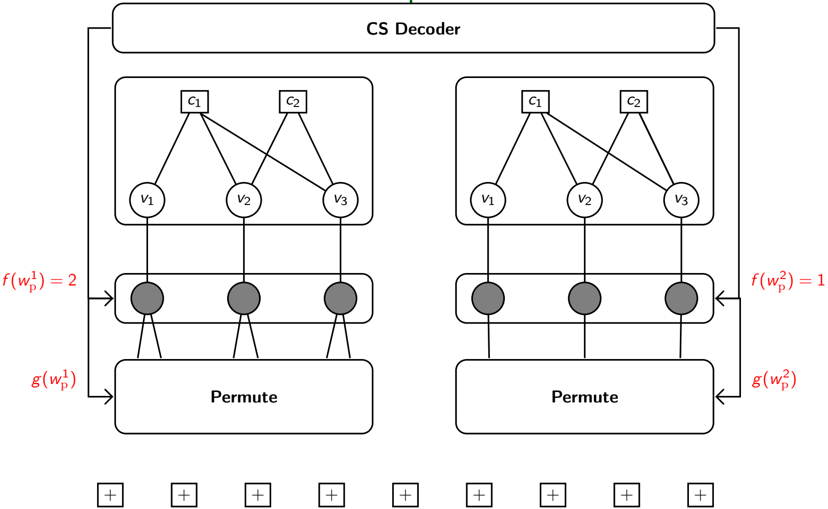
Sparse IDMA - Encoding



- Divide the message into two parts: $w_\mathrm{p}, w_\mathrm{c}$
- w_p is transmitted using compressed sensing
- w_c is transmitted using a channel code
- Based on w_p a repetition pattern and permutation pattern is chosen for the channel coding part

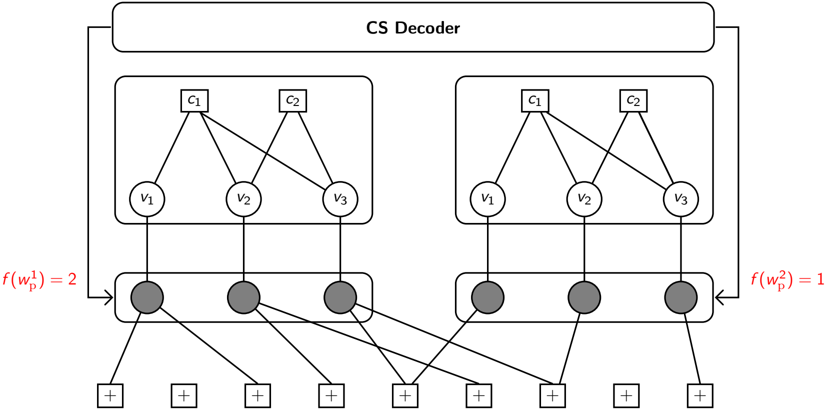


CS Decoder and the Joint Graph



- Decode the first part using non-negative least square
- Recover the permutation patterns from the first part

CS Decoder and the Joint Graph



- Decode the first part using non-negative least square
- Recover the permutation patterns from the first part
- Use the permutation patterns to decode the second part of the message by using message passing decoder

Density Evolution and Threshold

Density Evolution

Compute $I^t_{+\to v}, I^t_{v\to +}, I^t_{v\to c}(i), I^{t-1}_{c\to v}(i)$ from $I^{t-1}_{+\to v}, I^{t-1}_{v\to +} I^{t-1}_{v\to c}(i), I^{t-1}_{c\to v}(i)$ for $t = 1, 2, \cdots, \infty$

[4] R. Storn and K. Price, "Differential evolution a simple and efficient heuristic for global optimization over continuous spaces," Journal of Global Optimization.

Density Evolution and Threshold

Density Evolution

Compute $I^{t}_{+\to v}, I^{t}_{v\to +}, I^{t}_{v\to c}(i), I^{t-1}_{c\to v}(i)$ from $I^{t-1}_{+\to v}, I^{t-1}_{v\to +} I^{t-1}_{v\to c}(i), I^{t-1}_{c\to v}(i)$ for $t = 1, 2, \cdots, \infty$

Threshold

Threshold σ^*= maximum σ such that $I_{v\to c}(i) \to 1$ for each $i \in E$

[4]R. Storn and K. Price, "Differential evolution a simple and efficient heuristic for global optimization over continuous spaces," Journal of Global Optimization.

Density Evolution and Threshold

Density Evolution

Compute $I^t_{+\to v}, I^t_{v\to +}, I^t_{v\to c}(i), I^{t-1}_{c\to v}(i)$ from $I^{t-1}_{+\to v}, I^{t-1}_{v\to +} I^{t-1}_{v\to c}(i), I^{t-1}_{c\to v}(i)$ for $t = 1, 2, \cdots, \infty$

Threshold

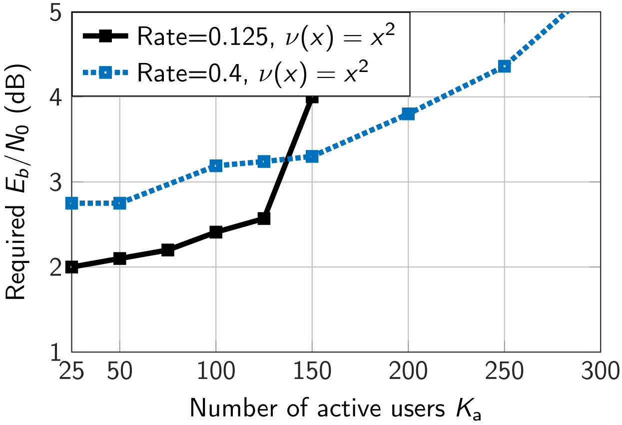
Threshold σ^*= maximum σ such that $I_{v\to c}(i) \to 1$ for each $i \in E$

Optimization

Optimize the protograph and repetition factor to maximize the threshold using differential evolution[4]

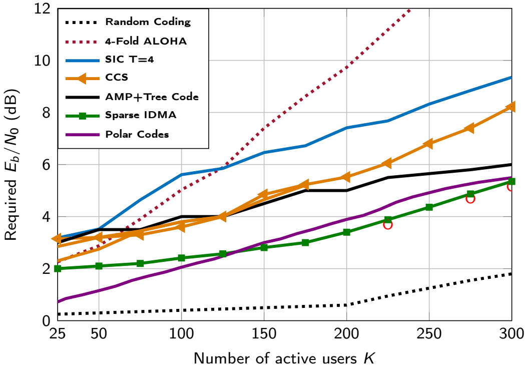
[4] R. Storn and K. Price, "Differential evolution a simple and efficient heuristic for global optimization over continuous spaces," Journal of Global Optimization.

Rate of the LDPC Code vs K



- Optimal rate changes with K

Performance Comparison



- $B = 100$, $N = 30000$
- Only 3.2 dB away from Polyanksiy's achievability result

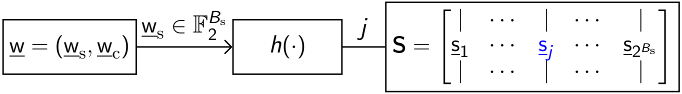
Takeaways

- Slotted ALOHA - interference cancellation for handling interference
- Proposed an IDMA like scheme for using the dimensions better
- Sparse IDMA vs. IDMA
  - Sparsity allows us to control interference
  - Makes it easier to design LDPC like codes
- Low complexity scheme for large number of users

What this part is about

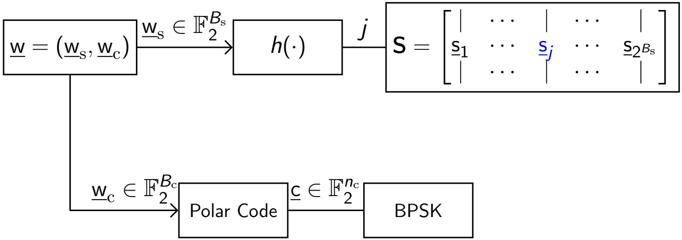
- (Non-orthogonal) spreading sequences for controlling interference
- Spreading + Polar codes + list decoding

Encoding



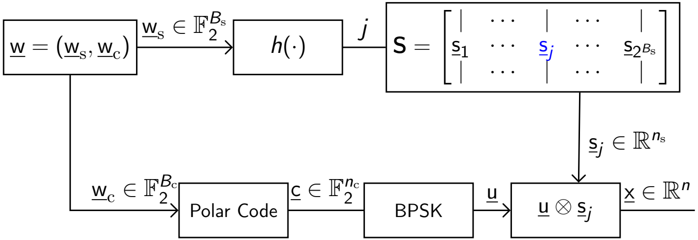
- Divide the message into two parts: $w_{\mathrm{s}}, w_{\mathrm{c}}$
- Based on w_{s} a spreading sequence is chosen from the set $\mathbf{S}$

Encoding



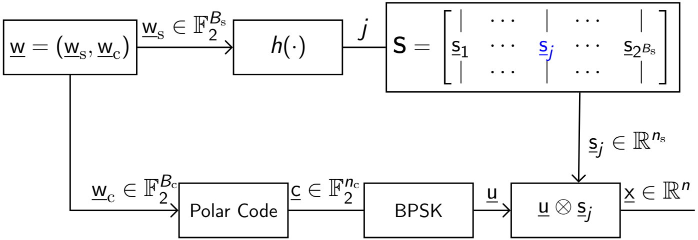
- Divide the message into two parts: w_s, w_c
- Based on w_s a spreading sequence is chosen from the set $\mathbf{S}$
- w_c is encoded using a polar code

Encoding



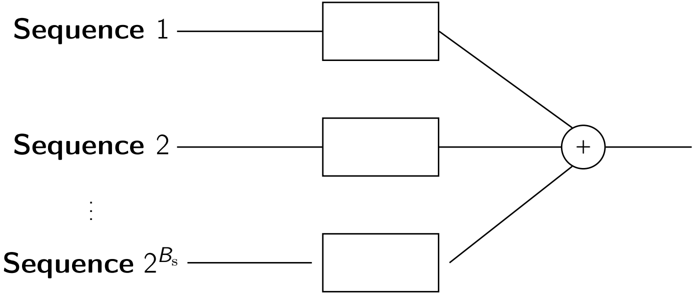
- Divide the message into two parts: w_s, w_c
- Based on w_s a spreading sequence is chosen from the set $\mathbf{S}$
- w_c is encoded using a polar code
- Coded bits are spread using the spreading sequence $\underline{s}_j$

Encoding



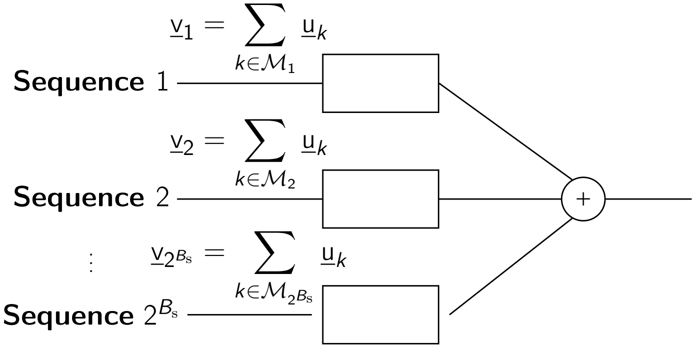
- Divide the message into two parts: w_s, w_c
- Based on w_s a spreading sequence is chosen from the set $\mathbf{S}$
- w_c is encoded using a polar code
- Coded bits are spread using the spreading sequence $\underline{s}_j$
- 2^{B_s} is not too large
- With non-trivial probability, multiple users will choose the same $\underline{s}_j$

Transmitter from the Spreading Sequence Perspective



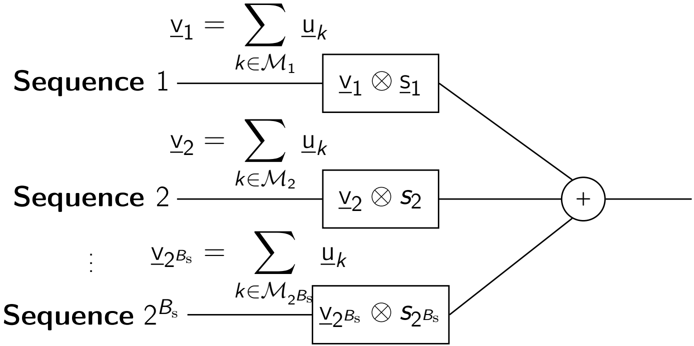
- $\mathcal{M}_j$: set of active users who choose $\underline{s}_j$

Transmitter from the Spreading Sequence Perspective



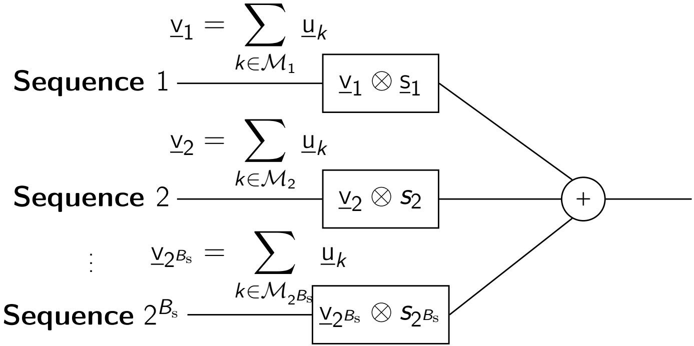
- $\mathcal{M}_j$: set of active users who choose $\underline{s}_j$
- Sum of the codewords associated with sequence $\underline{s}_j$: $\underline{v}_j = \sum_{k \in \mathcal{M}_j} \underline{u}_k$

Transmitter from the Spreading Sequence Perspective



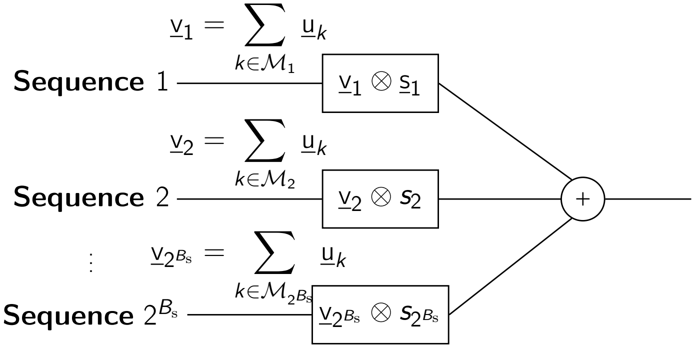
- $\mathcal{M}_j$: set of active users who choose $\underline{s}_j$
- Sum of the codewords associated with sequence $\underline{s}_j$: $\underline{v}_j = \sum_{k \in \mathcal{M}_j} \underline{u}_k$

Transmitter from the Spreading Sequence Perspective



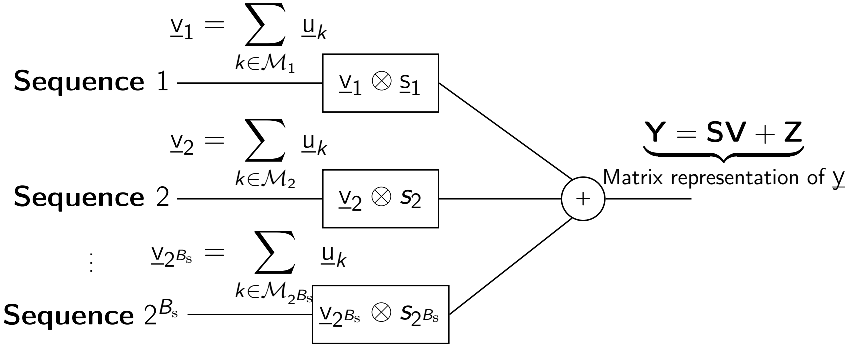
- $\mathcal{M}_j$: set of active users who choose $\underline{s}_j$
- Sum of the codewords associated with sequence $\underline{s}_j$: $\underline{v}_j = \sum_{k \in \mathcal{M}_j} \underline{u}_k$
- $\mathbf{V} := \begin{bmatrix} \underline{v}_1^{\mathsf{T}} & \underline{v}_2^{\mathsf{T}} & \cdots & \underline{v}_{2^{B_s}}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}$

Transmitter from the Spreading Sequence Perspective



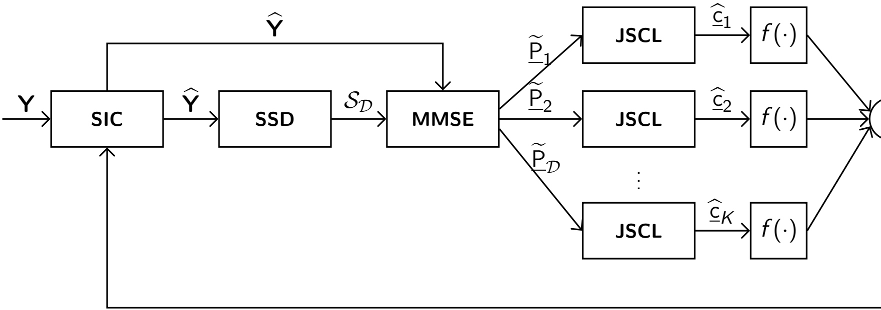
- $\mathcal{M}_j$: set of active users who choose $\underline{s}_j$
- Sum of the codewords associated with sequence $\underline{s}_j$: $\underline{v}_j = \sum_{k \in \mathcal{M}_j} \underline{u}_k$
- $\mathbf{V} := \begin{bmatrix} \underline{v}_1^{\mathsf{T}} & \underline{v}_2^{\mathsf{T}} & \cdots & \underline{v}_{2^{B_s}}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}$
- $\underline{y} = \underbrace{\underline{y}(1 : n_s)}_{\underline{y}_1^{\mathsf{T}}} \underbrace{\underline{y}(n_s + 1 : 2n_s)}_{\underline{y}_2^{\mathsf{T}}} \cdots \underbrace{\underline{y}((i-1)n_s + 1 : in_s)}_{\underline{y}_i^{\mathsf{T}}} \cdots \underbrace{\underline{y}(N - n_s + 1 : n_c)}_{\underline{y}_{n_c}^{\mathsf{T}}}$

Transmitter from the Spreading Sequence Perspective



- $\mathcal{M}_j$: set of active users who choose $\underline{s}_j$
- Sum of the codewords associated with sequence $\underline{s}_j$: $\underline{v}_j = \sum_{k\in\mathcal{M}_j} \underline{u}_k$
- $\mathbf{V} := \begin{bmatrix} \underline{v}_1^{\mathsf{T}} & \underline{v}_2^{\mathsf{T}} & \cdots & \underline{v}_{2^{B_s}}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}$
- $\underline{y} =$
  $\underbrace{\underline{y}(1:n_s)}_{\underline{y}_1^{\mathsf{T}}}\underbrace{\underline{y}(n_s+1:2n_s)}_{\underline{y}_2^{\mathsf{T}}}\cdots\underbrace{\underline{y}((i-1)n_s+1:in_s)}_{\underline{y}_i^{\mathsf{T}}}\cdots\underbrace{\underline{y}(N-n_s+1:n_c)}_{\underline{y}_{n_c}^{\mathsf{T}}}$

Main Components of the Receiver



- Blind Spreading Sequence detector (SSD)
- Soft Output MMSE Multi-user Detector
- Joint successive cancellation list (JSCL) decoder of polar codes + CRC
- Successive interference canceller (SIC)

Illustration of Decoding: $K = 3$

- User 1 picks $\underline{s}_5$, $\underline{v}_5 = \underline{u}_1$
- Users 2 and 3 pick $\underline{s}_1$, $\underline{v}_1 = \underline{u}_2 + \underline{u}_3$

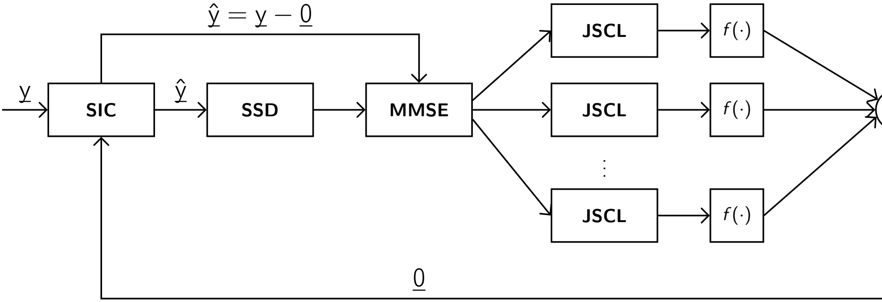
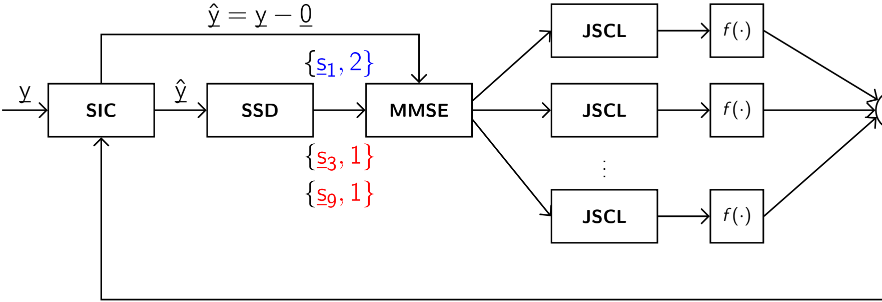


Illustration of Decoding: $K = 3$

- User 1 picks $\underline{s}_5$, $\underline{v}_5 = \underline{u}_1$
- Users 2 and 3 pick $\underline{s}_1$, $\underline{v}_1 = \underline{u}_2 + \underline{u}_3$

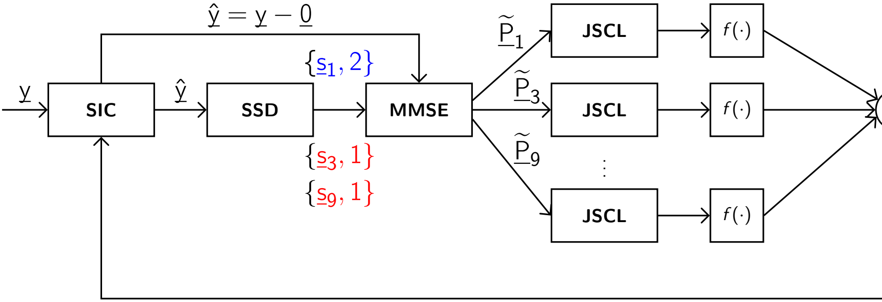


Iteration 1

- $S_{\mathcal{D}} = \{\underline{s}_1, \underline{s}_3, \underline{s}_9\}$.

Illustration of Decoding: $K = 3$

- User 1 picks $\underline{s}_5$, $\underline{v}_5 = \underline{u}_1$
- Users 2 and 3 pick $\underline{s}_1$, $\underline{v}_1 = \underline{u}_2 + \underline{u}_3$

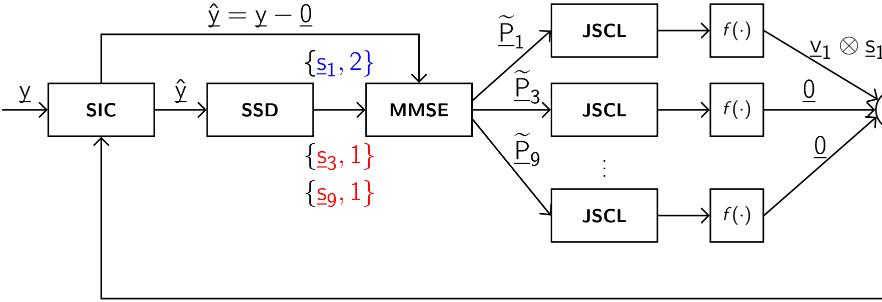


Iteration 1

- $S_{\mathcal{D}} = \{\underline{s}_1, \underline{s}_3, \underline{s}_9\}$.

Illustration of Decoding: $K = 3$

- User 1 picks $\underline{s}_5$, $\underline{v}_5 = \underline{u}_1$
- Users 2 and 3 pick $\underline{s}_1$, $\underline{v}_1 = \underline{u}_2 + \underline{u}_3$

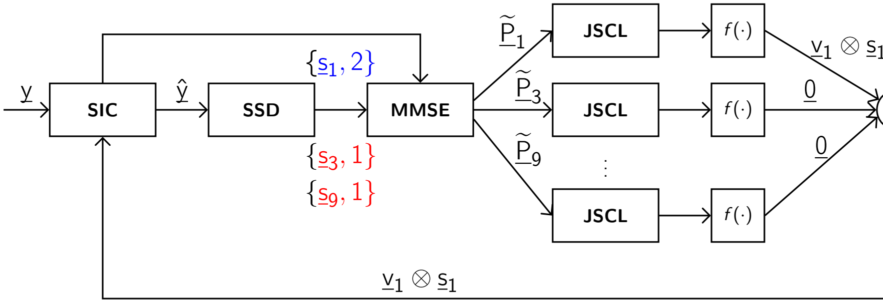


Iteration 1

- $S_{\mathcal{D}} = \{\underline{s}_1, \underline{s}_3, \underline{s}_9\}$.
- Decoded users: $2, 3$.

Illustration of Decoding: $K = 3$

- User 1 picks $\underline{s}_5$, $\underline{v}_5 = \underline{u}_1$
- Users 2 and 3 pick $\underline{s}_1$, $\underline{v}_1 = \underline{u}_2 + \underline{u}_3$

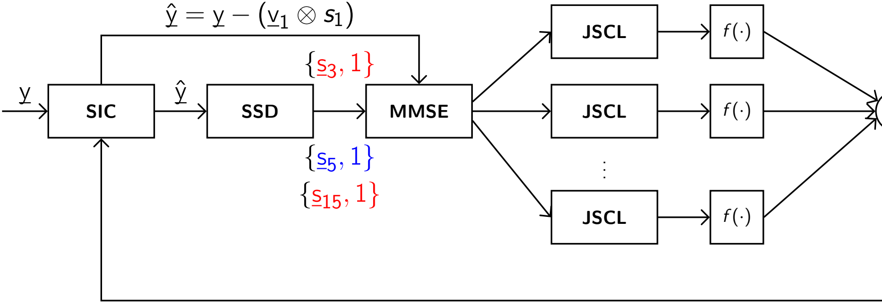


Iteration 1

- $S_{\mathcal{D}} = \{\underline{s}_1, \underline{s}_3, \underline{s}_9\}$.
- Decoded users: 2, 3.

Illustration of Decoding: $K = 3$

- User 1 picks $\underline{s}_5$, $\underline{v}_5 = \underline{u}_1$
- Users 2 and 3 pick $\underline{s}_1$, $\underline{v}_1 = \underline{u}_2 + \underline{u}_3$



Iteration 1

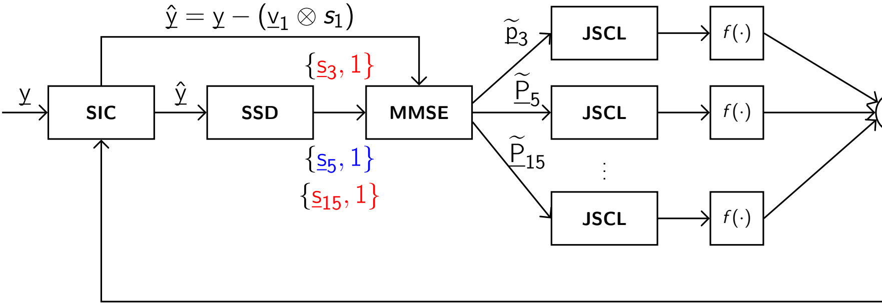
- $S_{\mathcal{D}} = \{\underline{s}_1, \underline{s}_3, \underline{s}_9\}$.
- Decoded users: 2, 3.

Iteration 2

- $S_{\mathcal{D}} = \{\underline{s}_3, \underline{s}_5, \underline{s}_{15}\}$.

Illustration of Decoding: $K = 3$

- User 1 picks $\underline{s}_5$, $\underline{v}_5 = \underline{u}_1$
- Users 2 and 3 pick $\underline{s}_1$, $\underline{v}_1 = \underline{u}_2 + \underline{u}_3$



Iteration 1

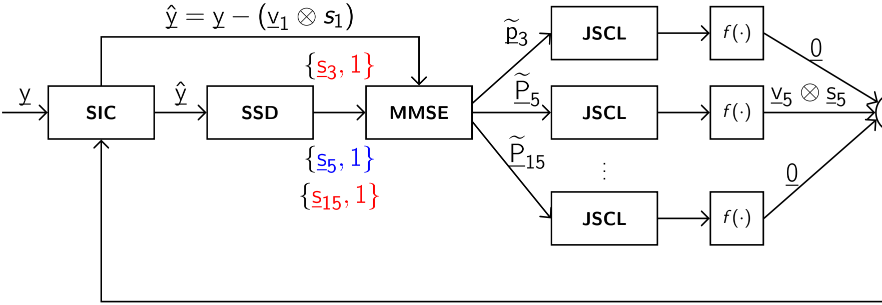
- $S_{\mathcal{D}} = \{\underline{s}_1, \underline{s}_3, \underline{s}_9\}$.
- Decoded users: 2, 3.

Iteration 2

- $S_{\mathcal{D}} = \{\underline{s}_3, \underline{s}_5, \underline{s}_{15}\}$.

Illustration of Decoding: $K = 3$

- User 1 picks $\underline{s}_5$, $\underline{v}_5 = \underline{u}_1$
- Users 2 and 3 pick $\underline{s}_1$, $\underline{v}_1 = \underline{u}_2 + \underline{u}_3$



Iteration 1

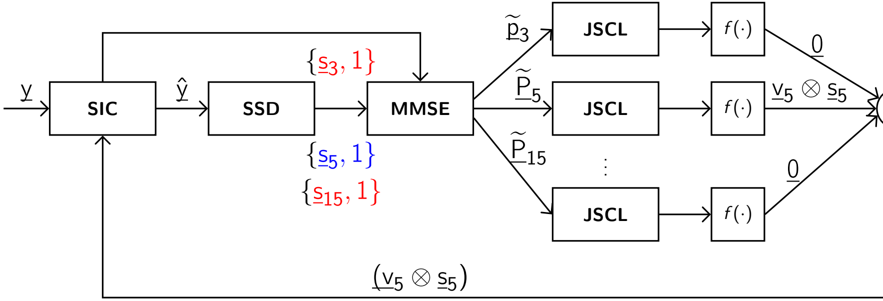
- $S_{\mathcal{D}} = \{\underline{s}_1, \underline{s}_3, \underline{s}_9\}$.
- Decoded users: $2, 3$.

Iteration 2

- $S_{\mathcal{D}} = \{\underline{s}_3, \underline{s}_5, \underline{s}_{15}\}$.
- Decoded users: 1.

Illustration of Decoding: $K = 3$

- User 1 picks $\underline{s}_5$, $\underline{v}_5 = \underline{u}_1$
- Users 2 and 3 pick $\underline{s}_1$, $\underline{v}_1 = \underline{u}_2 + \underline{u}_3$



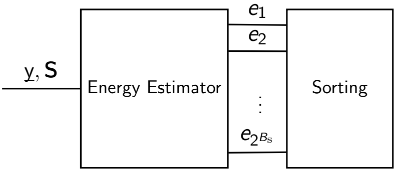
Iteration 1

- $S_{\mathcal{D}} = \{\underline{s}_1, \underline{s}_3, \underline{s}_9\}$.
- Decoded users: 2, 3.

Iteration 2

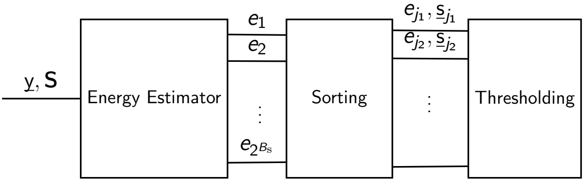
- $S_{\mathcal{D}} = \{\underline{s}_3, \underline{s}_5, \underline{s}_{15}\}$.
- Decoded users: 1.

(Blind) Spreading Sequence Detector



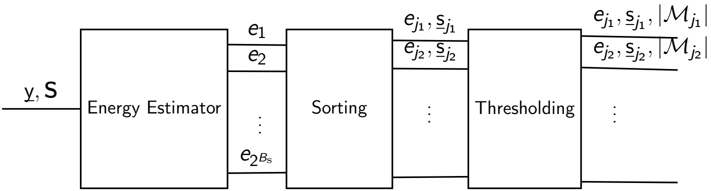
- For each $\underline{s}_j \in \mathbf{S}$ compute the statistic $e_j = \sum_{i=1}^{n_c} \left(\underline{y}_i^T \underline{s}_j\right)^2$

(Blind) Spreading Sequence Detector



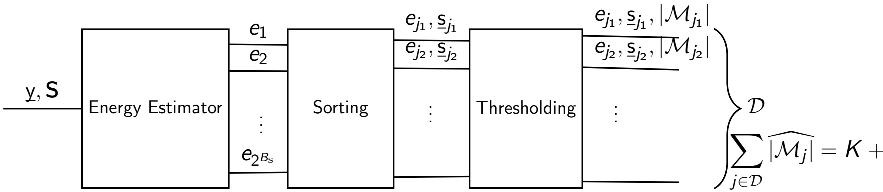
- For each $\underline{s}_j \in \mathbf{S}$ compute the statistic $e_j = \sum_{i=1}^{n_c} \left(\underline{y}_i^T \underline{s}_j\right)^2$
- Sort sequences in descending order of their statistics

(Blind) Spreading Sequence Detector



- For each $\underline{s}_j \in \mathbf{S}$ compute the statistic $e_j = \sum_{i=1}^{n_c} \left(\underline{y}_i^T \underline{s}_j\right)^2$
- Sort sequences in descending order of their statistics
- Based on e_j compute estimate $\widehat{|\mathcal{M}_j|}$ of $|\mathcal{M}_j|$

(Blind) Spreading Sequence Detector



- For each $\underline{s}_j \in \mathbf{S}$ compute the statistic $e_j = \sum_{i=1}^{n_c} \left(\underline{y}_i^T \underline{s}_j \right)^2$
- Sort sequences in descending order of their statistics
- Based on e_j compute estimate $\widehat{|\mathcal{M}_j|}$ of $|\mathcal{M}_j|$
- Output first $|\mathcal{D}|$ sequences from the sorted list
- Define $\widehat{\mathbf{M}} := \mathrm{diag}(|\widehat{\mathcal{M}}_1|, |\widehat{\mathcal{M}}_2|, \dots, |\widehat{\mathcal{M}}_{|\mathcal{D}|}|)$

MMSE Estimator

- The received signal is hypothesized as

$$\mathbf{Y} = \mathbf{S}_{\mathcal{D}}\mathbf{V}_{\mathcal{D}} + \mathbf{Z}$$

MMSE Estimator

- The received signal is hypothesized as

$$\mathbf{Y} = \mathbf{S}_{\mathcal{D}} \mathbf{V}_{\mathcal{D}} + \mathbf{Z}$$

- Pass $\mathbf{Y}$ through a MMSE filter to obtain an estimate $\widetilde{\mathbf{V}}$

$$\widetilde{\mathbf{V}} = \begin{bmatrix} \widetilde{\underline{\mathsf{v}}}_1 \\ \widetilde{\underline{\mathsf{v}}}_2 \\ \vdots \\ \widetilde{\underline{\mathsf{v}}}_{|\mathcal{D}|} \end{bmatrix} = \underbrace{\widehat{\mathbf{M}} \mathbf{S}_{\mathcal{D}}^{\mathsf{T}} (\mathbf{S}_{\mathcal{D}} \mathbf{S}_{\mathcal{D}}^{\mathsf{T}} + I_{n_{\mathrm{s}}})^{-1}}_{\text{Linear MMSE filter}} \mathbf{Y}$$

MMSE Estimator

- The received signal is hypothesized as

$$\mathbf{Y} = \mathbf{S}_{\mathcal{D}}\mathbf{V}_{\mathcal{D}} + \mathbf{Z}$$

- Pass $\mathbf{Y}$ through a MMSE filter to obtain an estimate $\widetilde{\mathbf{V}}$

$$\widetilde{\mathbf{V}} = \begin{bmatrix} \widetilde{\underline{v}}_1 \\ \widetilde{\underline{v}}_2 \\ \vdots \\ \widetilde{\underline{v}}_{|\mathcal{D}|} \end{bmatrix} = \underbrace{\widehat{\mathbf{M}}\mathbf{S}_{\mathcal{D}}^{\mathsf{T}}(\mathbf{S}_{\mathcal{D}}\mathbf{S}_{\mathcal{D}}^{\mathsf{T}} + I_{n_{\mathrm{s}}})^{-1}}_{\text{Linear MMSE filter}}\mathbf{Y}$$

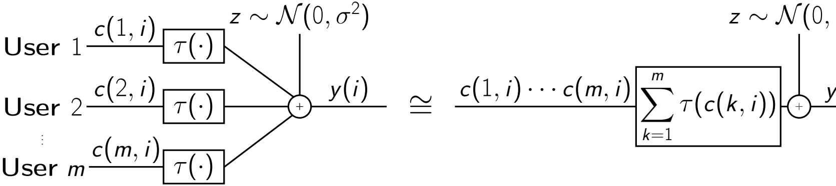
- The error covariance matrix is given by

$$\Sigma = I_{|\mathcal{D}|} - \widehat{\mathbf{M}}\mathbf{S}_{\mathcal{D}}^{\mathsf{T}}(\mathbf{S}_{\mathcal{D}}\mathbf{S}_{\mathcal{D}}^{\mathsf{T}} + I_{n_{\mathrm{s}}})^{-1}\widehat{\mathbf{M}}\mathbf{S}_{\mathcal{D}}$$

- We convert $\widetilde{v}_j$ and Σ_{jj} into LLRs to be fed to Polar decoder

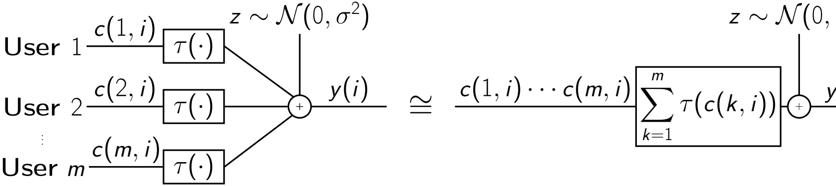
JSCL Decoding of Polar Codes

- Recall that multiple users can pick the same spreading sequence
- m-user GMAC over $\mathbb{F}_2$ is equivalent to single user AWGN over $\mathbb{F}_2^m$.



JSCL Decoding of Polar Codes

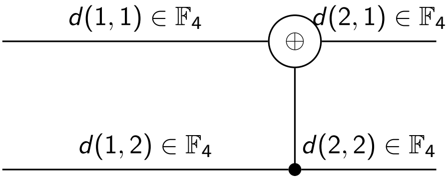
- Recall that multiple users can pick the same spreading sequence
- m-user GMAC over $\mathbb{F}_2$ is equivalent to single user AWGN over $\mathbb{F}_2^m$.



- $\underline{c}(:, i) = \begin{bmatrix} \underline{c}(1, i) & \underline{c}(2, i) & \cdots & \underline{c}(m, i) \end{bmatrix}$
- $\Pr(\underline{c}(:, i) = \mathrm{g} | y(i)) \propto \exp\left(-\frac{(y(i) - \tau(\mathrm{g}))^2}{2\sigma^2}\right)$, for $\mathrm{g} \in \mathbb{F}_2^m$

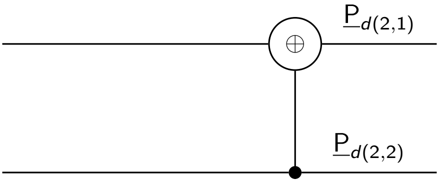
Example: JSC Decoding of Polar Codes

- $m = 2, n_{\mathrm{c}} = 2$



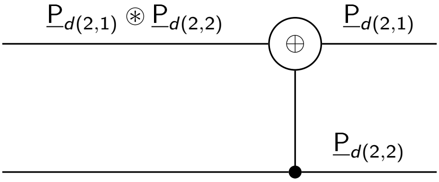
Example: JSC Decoding of Polar Codes

- $m = 2, n_c = 2$
- $$\underline{P}_{d(2,1)} = \Pr(d(2,1)|y(1)) = \{\Pr(00|y(1)), \Pr(01|y(1)), \Pr(10|y(1)), \Pr(11|y(1))$$
- $$\underline{P}_{d(2,2)} = \Pr(d(2,2)|y(2)) = \{\Pr(00|y(2)), \Pr(01|y(2)), \Pr(10|y(2)), \Pr(11|y(2))$$



Example: JSC Decoding of Polar Codes

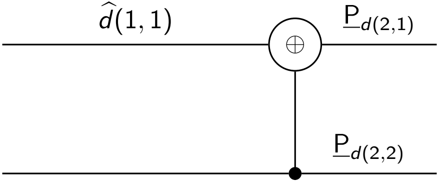
- $m = 2, n_{\mathrm{c}} = 2$
- $$\underline{P}_{d(2,1)} = \Pr\left(d(2,1)|y(1)\right) = \{\Pr\left(00|y(1)\right), \Pr\left(01|y(1)\right), \Pr\left(10|y(1)\right), \Pr\left(11|y(1)\right)\}$$
- $$\underline{P}_{d(2,2)} = \Pr\left(d(2,2)|y(2)\right) = \{\Pr\left(00|y(2)\right), \Pr\left(01|y(2)\right), \Pr\left(10|y(2)\right), \Pr\left(11|y(2)\right)\}$$



- $\underline{P}_{d(1,1)} = \underline{P}_{d(2,1)} \circledast \underline{P}_{d(2,2)}$

Example: JSC Decoding of Polar Codes

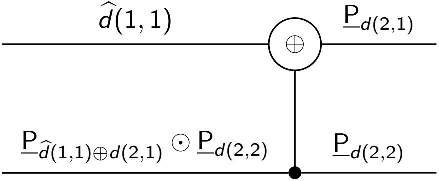
- $m = 2, n_{\mathrm{c}} = 2$
- $$\underline{\mathrm{P}}_{d(2,1)} = \Pr\left(d(2,1)|y(1)\right) = \{\Pr\left(00|y(1)\right), \Pr\left(01|y(1)\right), \Pr\left(10|y(1)\right), \Pr\left(11|y(1)\right)\}$$
- $$\underline{\mathrm{P}}_{d(2,2)} = \Pr\left(d(2,2)|y(2)\right) = \{\Pr\left(00|y(2)\right), \Pr\left(01|y(2)\right), \Pr\left(10|y(2)\right), \Pr\left(11|y(2)\right)\}$$



- $\underline{\mathrm{P}}_{d(1,1)} = \underline{\mathrm{P}}_{d(2,1)} \circledast \underline{\mathrm{P}}_{d(2,2)}$
- Based on $\underline{\mathrm{P}}_{d(1,1)}$ make a hard decision $\widehat{d}(1,1)$ on $d(1,1)$

Example: JSC Decoding of Polar Codes

- $m = 2, n_c = 2$
- $$\underline{P}_{d(2,1)} = \Pr\left(d(2,1)|y(1)\right) = \left\{\Pr\left(00|y(1)\right), \Pr\left(01|y(1)\right), \Pr\left(10|y(1)\right), \Pr\left(11|y(1)\right)\right\}$$
- $$\underline{P}_{d(2,2)} = \Pr\left(d(2,2)|y(2)\right) = \left\{\Pr\left(00|y(2)\right), \Pr\left(01|y(2)\right), \Pr\left(10|y(2)\right), \Pr\left(11|y(2)\right)\right\}$$



- $\underline{P}_{d(1,1)} = \underline{P}_{d(2,1)} \circledast \underline{P}_{d(2,2)}$
- Based on $\underline{P}_{d(1,1)}$ make a hard decision $\widehat{d}(1,1)$ on $d(1,1)$
- $\underline{P}_{d(1,2)} = \underline{P}_{\widehat{d}(1,1)+d(2,1)} \odot \underline{P}_{d(2,2)}$

Successive Interference Cancellation

- If the decoding is successful, remove $\underline{\widetilde{v}}_j$ from $\underline{y}$

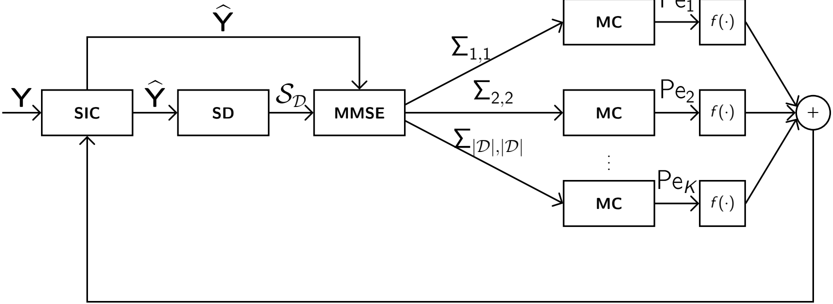
$$\underline{y} = \underline{y} - \underline{v}_j \otimes \underline{s}_j$$

Choice of Parameters

Parameters to choose

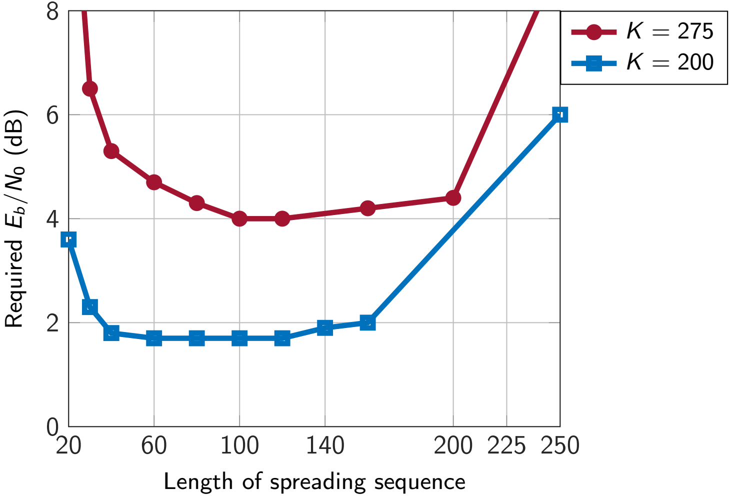
- Spreading sequence length
- Rate of the code
- Number of spreading sequences in the master list

Density Evolution Using Meta-Converse (MC) Bound

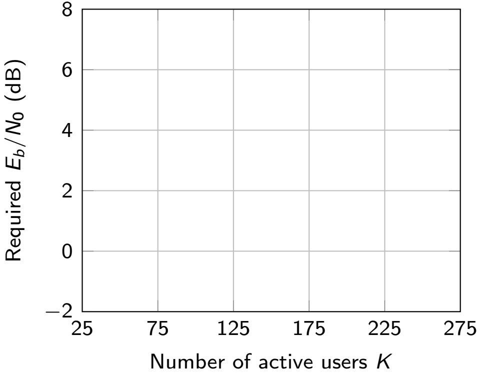


- $\Sigma = I_{|\mathcal{D}|} - \widehat{\mathbf{M}}\mathbf{S}_{\mathcal{D}}^{\mathsf{T}}(\mathbf{S}_{\mathcal{D}}\mathbf{S}_{\mathcal{D}}^{\mathsf{T}} + I_{n_{\mathrm{s}}})^{-1}\widehat{\mathbf{M}}\mathbf{S}_{\mathcal{D}}$
- $f(\mathrm{Pe}_j) = \begin{cases} \underline{\mathrm{v}}_1 \otimes s_1, \text{ with probability } 1 - \mathrm{Pe}_j \\ 0, \text{ with probability } \mathrm{Pe}_j \end{cases}$

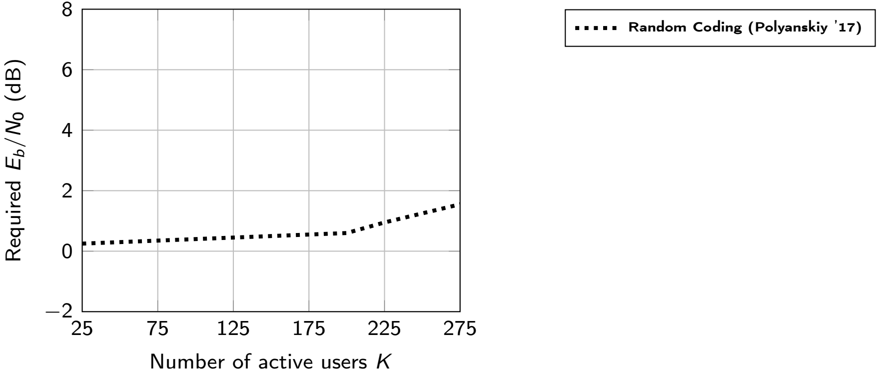
SNR versus Length of Spreading Sequences



Comparison

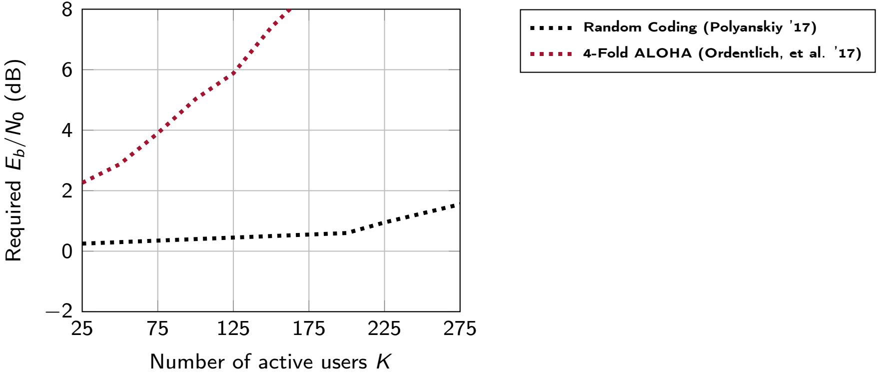

8
6
4
2
0
−2
25
75
125
175
225
275
Required E_b/N_0 (dB)
Number of active users K

Comparison



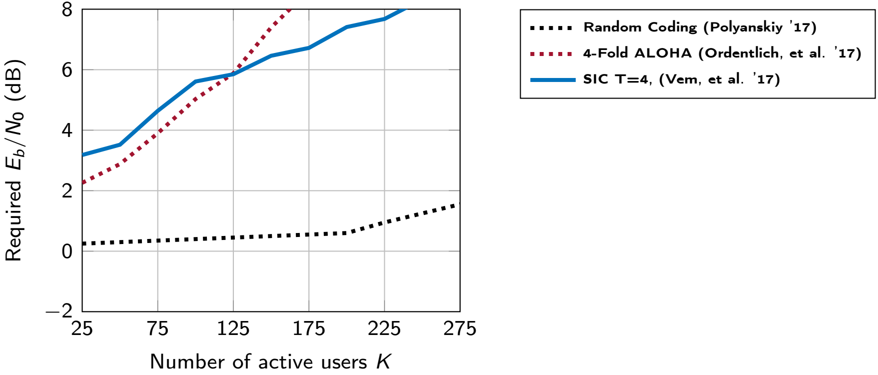
Required E_b/N_0 (dB)
8
6
4
2
0
−2
25
75
125
175
225
275
Number of active users K
Random Coding (Polyanskiy '17)

Comparison



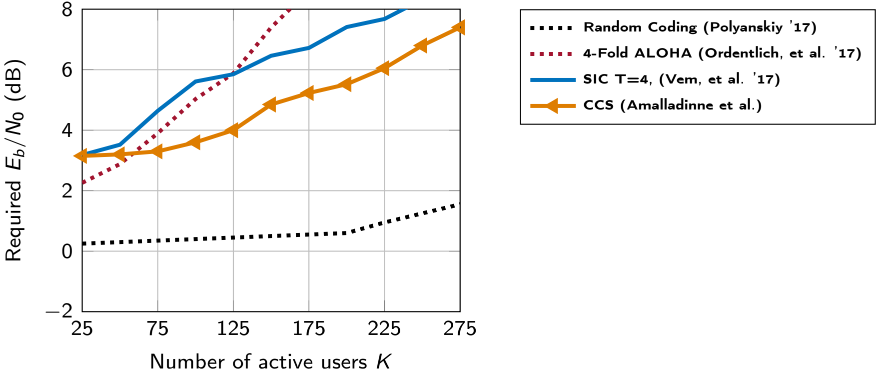
Required E_b/N_0 (dB)
8
6
4
2
0
−2
25
75
125
175
225
275
Number of active users K
Random Coding (Polyanskiy '17)
4-Fold ALOHA (Ordentlich, et al. '17)

Comparison



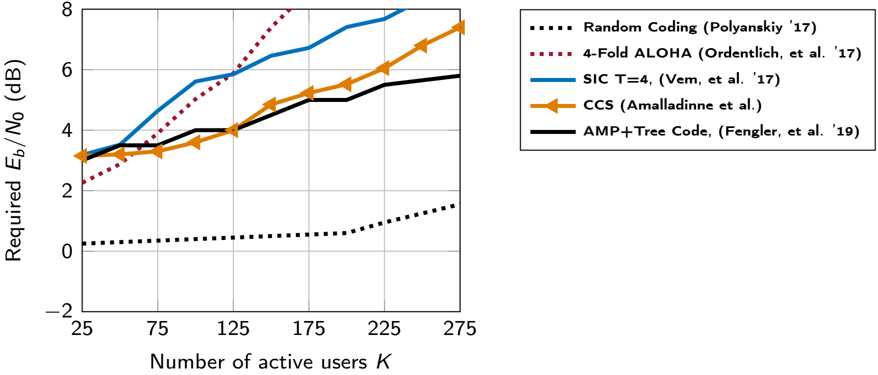
Required E_b/N_0 (dB)
8
6
4
2
0
−2
25
75
125
175
225
275
Number of active users K
Random Coding (Polyanskiy '17)
4-Fold ALOHA (Ordentlich, et al. '17)
SIC T=4, (Vem, et al. '17)

Comparison



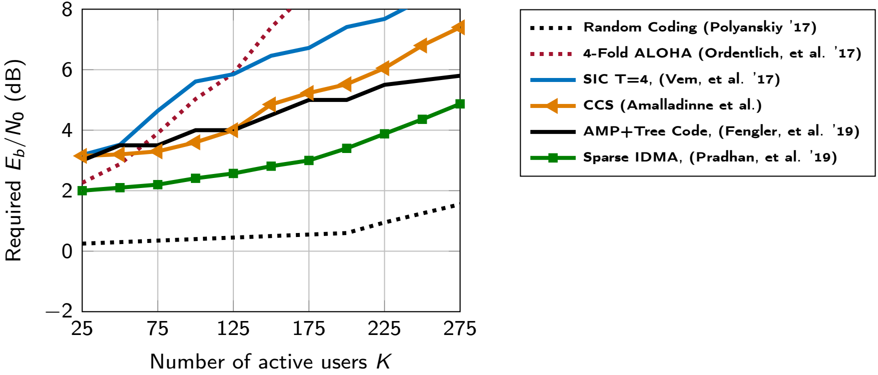
8
6
4
2
0
−2
25
75
125
175
225
275
Required E_b/N_0 (dB)
Number of active users K
Random Coding (Polyanskiy '17)
4-Fold ALOHA (Ordentlich, et al. '17)
SIC T=4, (Vem, et al. '17)
CCS (Amalladinne et al.)

Comparison



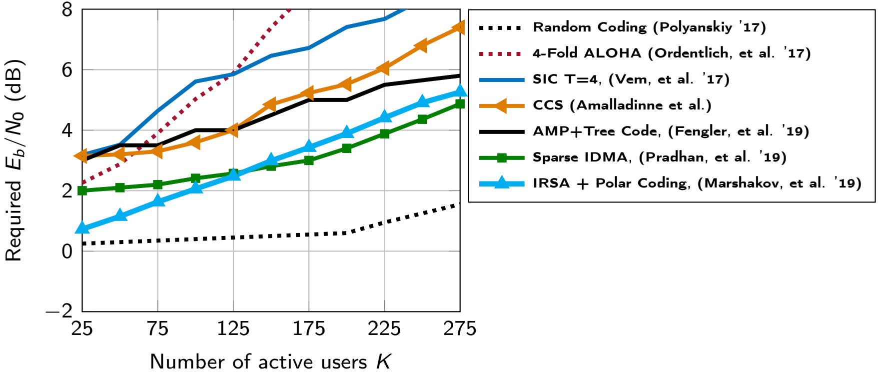
8
6
4
2
0
−2
25
75
125
175
225
275
Required E_b/N_0 (dB)
Number of active users K
Random Coding (Polyanskiy '17)
4-Fold ALOHA (Ordentlich, et al. '17)
SIC T=4, (Vem, et al. '17)
CCS (Amalladinne et al.)
AMP+Tree Code, (Fengler, et al. '19)

Comparison



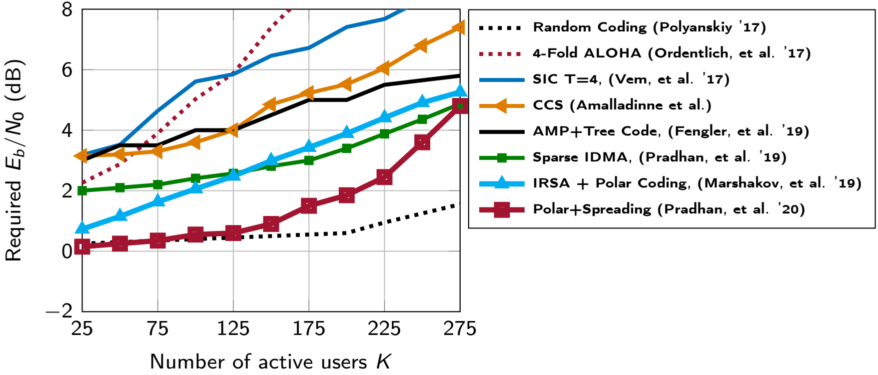
Required E_b/N_0 (dB)
8
6
4
2
0
−2
25
75
125
175
225
275
Number of active users K
Random Coding (Polyanskiy '17)
4-Fold ALOHA (Ordentlich, et al. '17)
SIC T=4, (Vem, et al. '17)
CCS (Amalladinne et al.)
AMP+Tree Code, (Fengler, et al. '19)
Sparse IDMA, (Pradhan, et al. '19)

Comparison



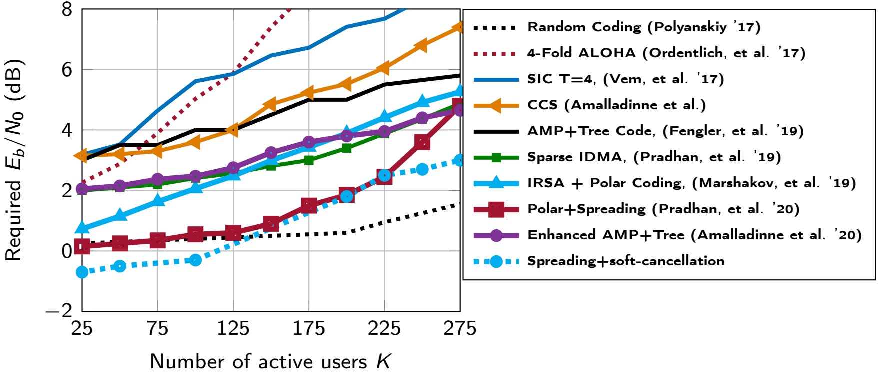
Required E_b/N_0 (dB)
8
6
4
2
0
−2
25
75
125
175
225
275
Number of active users K
Random Coding (Polyanskiy '17)
4-Fold ALOHA (Ordentlich, et al. '17)
SIC T=4, (Vem, et al. '17)
CCS (Amalladinne et al.)
AMP+Tree Code, (Fengler, et al. '19)
Sparse IDMA, (Pradhan, et al. '19)
IRSA + Polar Coding, (Marshakov, et al. '19)

Comparison



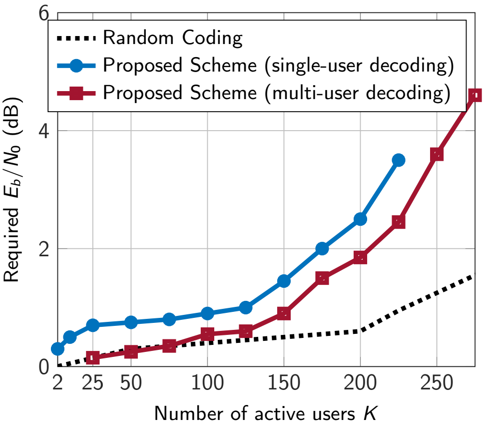
Required E_b/N_0 (dB)
8
6
4
2
0
−2
25
75
125
175
225
275
Number of active users K
Random Coding (Polyanskiy '17)
4-Fold ALOHA (Ordentlich, et al. '17)
SIC T=4, (Vem, et al. '17)
CCS (Amalladinne et al.)
AMP+Tree Code, (Fengler, et al. '19)
Sparse IDMA, (Pradhan, et al. '19)
IRSA + Polar Coding, (Marshakov, et al. '19)
Polar+Spreading (Pradhan, et al. '20)

Comparison



Required E_b/N_0 (dB)
8
6
4
2
0
−2
25 75 125 175 225 275
Number of active users K
Random Coding (Polyanskiy '17)
4-Fold ALOHA (Ordentlich, et al. '17)
SIC T=4, (Vem, et al. '17)
CCS (Amalladinne et al.)
AMP+Tree Code, (Fengler, et al. '19)
Sparse IDMA, (Pradhan, et al. '19)
IRSA + Polar Coding, (Marshakov, et al. '19)
Polar+Spreading (Pradhan, et al. '20)
Enhanced AMP+Tree (Amalladinne et al. '20)
Spreading+soft-cancellation

Simulation Results



- List size - 32
- m - 4
- CRC length - 16 bits

Take Aways

- Proposed a receiver with complexity $O(K^3)$ (can be reduced)
- Blind sequence detection + classical SIC+MMSE receivers
- Near finite length bound achieving codes are required (CRC+Polar+List)
- All these are standard components of a 5G system
- Scaling with the number of users should be improved